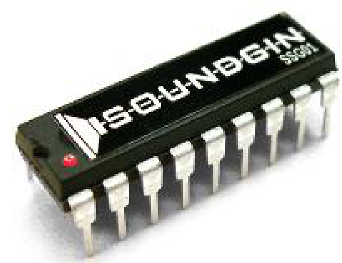
SOUNDGIN

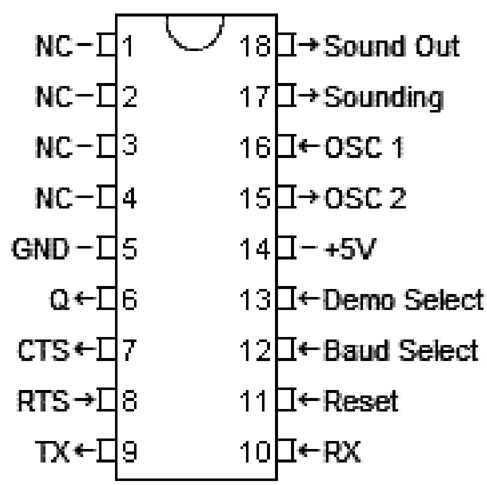
Voice and Sound Effects Engine

The Soundgin™ Voice and Sound Effects Engine a single chip 6-voice electronic music synthesizer / sound effects / voice synthesizer. It produces complex sound effects, synthesizer style music and English speech with an unlimited vocabulary. The chip provides wide-range, high resolution dynamic control of pitch (frequency), tone color (harmonic content) and dynamics (volume).



- Available in an 18-Pin DIP or SOIC package.
- 2.0 to 5.0 Volts Operation
- 16KHz Sample Output Rate
- 2400 or 9600 Baud Serial Connection
- 6 Independently Controlled Voices
- Amplitude Modulation
- Frequency Modulation
- Ring Modulation
- Hard Sync.
- ADSR Envelopes
- Musical Notes
- English Phonemes
- Sound Morphing
- In-Chip Presets

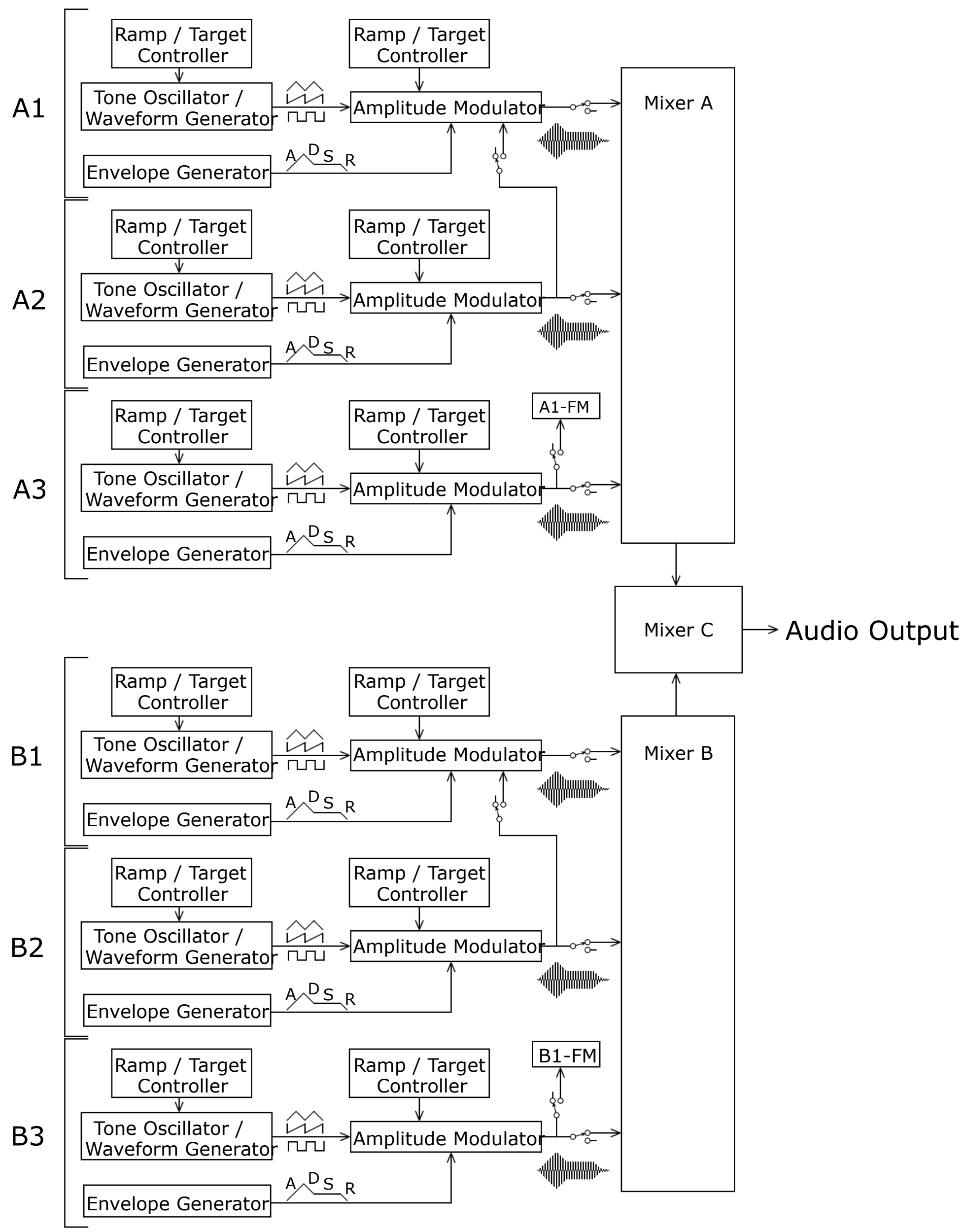
Pin Configuration



Description

The Soundgin consists of six synthesizer "voices" which can be used independently or in conjunction with each other to create complex sounds. Each voice consists of a Tone Oscillator / Wave Generator, an Envelope Generator, an Amplitude Modulator and Ramping / Target controls. The Tone Oscillator controls the pitch of the voice over a wide range. The Oscillator produces one of eight waveforms at the selected frequency, with the unique harmonic content of each waveform providing simple control of tone color. The volume dynamics of the oscillator are controlled by an Amplitude Modulator which can control by the Envelope Generator, the Ramping / Target control or by another oscillator. When triggered , the Envelope Generator creates an amplitude envelope with programmable rates of increasing and decreasing volume. In addition to the six voices, two mixers are provided to combine the waveforms outputs of the oscillators.

Functional Block Diagram



The oscillators and mixers are arranged so that there are two identical complex sound generators; A and B. Mixer C combines the output from mixer A and B into a final composite sound. The final sound is then sent out of the chip as Audio Out.

Operation

The Soundgin is a register based system where all the functions are controlled by the values that reside in the registers. There is a total of 144 eight-bit registers in Soundgin which control the generation of sound. These registers are listed in the table below. Soundgin **Register Map**

The 144 registers are arranged into 9 groups of 16 bytes each. Each group of registers controls an associated group of functions. Several of the groups are similar in function. Functionally similar groups are: Mixer A and B, Oscillators A1, A2, A3, B1, B2, and B3. The correlation between which bytes of memory control which functions is identical across the functionally similar groups. For instance, Byte 1 of Oscillator A1 is functionally identical to Byte 1 of Oscillator A2, A3, B1, etc. the only difference being which oscillator it controls.

Soundgin Register Map

Sound Generator A h00-h3F (000-063)

Mixer A	**h00-h0F**	**000-015**	**Oscillator A1**	**h10-h1F**	**016-031**
Mix Control 0	h00	000	Osc Control	h10	016
Mix Control 1	h01	001	Frequency Fine	h11	017
Osc A1 Distortion	h02	002	Frequency Low	h12	018
Osc A2 Distortion	h03	003	Frequency High	h13	019
Osc A3 Distortion	h04	004	Frequency Target Low	h14	020
Osc A1 PWM	h05	005	Frequency Target High	h15	021
Osc A2 PWM	h06	006	Frequency Transition Low	h16	022
Osc A3 PWM	h07	007	Frequency Transition High	h17	023
Amplitude	h08	008	Amplitude	h18	024
Amplitude Target	h09	009	Amplitude Target	h19	025
Amplitude Transition Low	h0A	010	Amplitude Transition Low	h1A	026
Amplitude Transition High	h0B	011	Amplitude Transition High	h1B	027
Envelope Control	h0C	012	Envelope Control	h1C	028
Envelope Attack	h0D	013	Envelope Attack	h1D	029
Envelope Decay	h0E	014	Envelope Decay	h1E	030
Envelope Release	h0F	015	Envelope Release	h1F	031

Oscillator A2	**h20-h2F**	**032-047**	**Oscillator A3**	**h30-h3F**	**048-063**
Osc Control	h20	032	Osc Control	h30	048
Frequency Fine	h21	033	Frequency Fine	h31	049
Frequency Low	h22	034	Frequency Low	h32	050
Frequency High	h23	035	Frequency High	h33	051
Frequency Target Low	h24	036	Frequency Target Low	h34	052
Frequency Target High	h25	037	Frequency Target High	h35	053
Frequency Transition Low	h26	038	Frequency Transition Low	h36	054
Frequency Transition High	h27	039	Frequency Transition High	h37	055
Amplitude	h28	040	Amplitude	h38	056
Amplitude Target	h29	041	Amplitude Target	h39	057
Amplitude Transition Low	h2A	042	Amplitude Transition Low	h3A	058
Amplitude Transition High	h2B	043	Amplitude Transition High	h3B	059
Envelope Control	h2C	044	Envelope Control	h3C	060
Envelope Attack	h2D	045	Envelope Attack	h3D	061
Envelope Decay	h2E	046	Envelope Decay	h3E	062
Envelope Release	h2F	047	Envelope Release	h3F	063

Soundgin Register Map (Cont.)

Sound Generator B h40-h7F (064-127)

Mixer B h40-h4F 064-079

Mix Control 0	h40	064
Mix Control 1	h41	065
Osc B1 Distortion	h42	066
Osc B2 Distortion	h43	067
Osc B3 Distortion	h44	068
Osc B1 PWM	h45	069
Osc B2 PWM	h46	070
Osc B3 PWM	h47	071
Amplitude	h48	072
Amplitude Target	h49	073
Amplitude Transition Low	h4A	074
Amplitude Transition High	h4B	075
Envelope Control	h4C	076
Envelope Attack	h4D	077
Envelope Decay	h4E	078
Envelope Release	h4F	079

Oscillator B1 h50-h5F 080-095

Osc Control	h50	080
Frequency Fine	h51	081
Frequency Low	h52	082
Frequency High	h53	083
Frequency Target Low	h54	084
Frequency Target High	h55	085
Frequency Transition Low	h56	086
Frequency Transition High	h57	087
Amplitude	h58	088
Amplitude Target	h59	089
Amplitude Transition Low	h5A	090
Amplitude Transition High	h5B	091
Envelope Control	h5C	092
Envelope Attack	h5D	093
Envelope Decay	h5E	094
Envelope Release	h5F	095

Oscillator B2 h60-h6F 096-111

Osc Control	h60	096
Frequency Fine	h61	097
Frequency Low	h62	098
Frequency High	h63	099
Frequency Target Low	h64	100
Frequency Target High	h65	101
Frequency Transition Low	h66	102
Frequency Transition High	h67	103
Amplitude	h68	104
Amplitude Target	h69	105
Amplitude Transition Low	h6A	106
Amplitude Transition High	h6B	107
Envelope Control	h6C	108
Envelope Attack	h6D	109
Envelope Decay	h6E	110
Envelope Release	h6F	111

Oscillator B3 h70-h7F 112-127

Osc Control	h70	112
Frequency Fine	h71	113
Frequency Low	h72	114
Frequency High	h73	115
Frequency Target Low	h74	116
Frequency Target High	h75	117
Frequency Transition Low	h76	118
Frequency Transition High	h77	119
Amplitude	h78	120
Amplitude Target	h79	121
Amplitude Transition Low	h7A	122
Amplitude Transition High	h7B	123
Envelope Control	h7C	124
Envelope Attack	h7D	125
Envelope Decay	h7E	126
Envelope Release	h7F	127

Chip Control h80-h8F (112-127)

Misc Control	h80	128
Speech Control	h81	129
Speech Frequency Low	h82	130
Speech Frequency High	h83	131
Transition Speed	h84	132
Internal Clock	h85	133
Pitch Bend A	h86	134
Pitch Bend B	h87	135
Amplitude	h88	136
Port	h89	137
Osc A1 Output	h8A	138
Osc A2 Output	h8B	139
Osc A3 Output	h8C	140
Osc B1 Output	h8D	141
Osc B2 Output	h8E	142
Osc B3 Output	h8F	143

Register Functions

Amplitude	signed 8-Bit value. Specifies the overall amplitude levels for the final composite oscillator output. The amplitude levels range from no output (0) to maximum amplitude (127 or $7F) in 128 linear steps. Some amplitude level other than zero must be selected in order for any output to occur. A negative value has the same effect.
Amplitude Target	signed 8-Bit value. Together these registers specify a target amplitude. When the oscillator's Amplitude Target bit is set, the oscillator's amplitude will move towards the target at the rate determined by the Amplitude Transition value until the oscillator's amplitude has reached the target.
Amplitude Transition Low Amplitude Transition High	signed 16-Bit value. Together these registers specify a amplitude transition. When the oscillator's Amplitude Ramp bit is set, this value will be added to the oscillator's amplitude. If the oscillator's Amplitude Targeting bit is set, then this value's sign is adjusted by the relation between the target amplitude and the current amplitude.
Distortion	unsigned 8-Bit value. This register specifies the level of distortion to apply to the oscillator's frequency.
Envelope Control	An 8-Bit value who's bits control functions of the oscillator's Envelope, Ramp and Target.

	Current Phase	
Bit 0-1	00	Attack
	01	Release
	10	None
	11	Decay
Bit 2	Frequency Target	
Bit 3	Amplitude Target	
Bit 4	Frequency Ramp	
Bit 5	Amplitude Ramp	
Bit 6	Envelope Running	
Bit 7	Envelope Active	

Envelope Attack Envelope Decay Envelope Release	An 8-Bit value that specifies the envelope's Attack, Decay and Release levels and transitions.

Bits 0-3	Duration
Bits 4-7	Target Amplitude

Frequency Fine Frequency Low Frequency High	signed 24-Bit number. Together these registers linearly control the frequency of the oscillator. The frequency is determined by the following equation: **Frequency = (FreqReg24 * 0.0009313226) Hz** The frequency register's resolution is sufficient for any tuning scale and allows sweeping from note to note (portamento) with no discernable frequency steps. The Soundgin has a built -in lookup table of the a440 temperament frequencies. Each oscillator can be set to a specific note in this table by issuing either the Play Note or Set Note command.
Frequency Target Low Frequency Target High	unsigned 16-Bit value. Together these registers specifies a target frequency. When the oscillator's Frequency Target bit is set, the oscillator's frequency will move towards the target at the rate determined by the Frequency Transition value until the oscillator's frequency has reached the target.
Frequency Transition Low Frequency Transition High	signed 16-Bit value. Together these registers specify a frequency transition. When the oscillator's Frequency Ramp bit is set, this value will be added to the two high bytes of the oscillator's frequency. If the oscillator's Frequency's Targeting bit is set, then this value's sign is adjusted by the relation between the Target Frequency and the current Frequency.
Internal Clock	An 8-Bit value that specifies a divider which is used to internally calculate 1ms.
Misc Control	An 8-Bit signed value who's bits control miscellaneous functions.

Bit 0	No Function
Bit 1	No Function
Bit 2	No Function
Bit 3	X-on / X-off
Bit 4	Q
Bit 5	Speed Low
Bit 6	No Function
Bit 7	No Function

Mix Control 0

An 8-Bit signed value who's bits control mixer functions.

Bit 0	Mix Oscillator
Bit 1	Mix Oscillator
Bit 2	Mix Oscillator
Bit 3	Modulate Mixer With Oscillator B1
Bit 4	Modulate Mixer With Oscillator B1 at 1/2
Bit 5	Synchronize Oscillators Oscillator B1
Bit 6	No Function
Bit 7	No Function

Mix Control 1

An 8-Bit signed value who's bits control mixer functions.

Bit 0	Modulate Oscillator 0 PWM With Oscillator B1
Bit 1	Modulate Oscillator 1 PWM With Oscillator B2
Bit 2	Modulate Oscillator 2 PWM With Oscillator B3
Bit 3	Modulate Oscillator 1 Amplitude With Oscillator 2
Bit 4	Modulate Oscillator 1 Amplitude With Oscillator 2 at 1/2
Bit 5	Synchronize Oscillator 1 with Oscillator 2
Bit 6	Modulate Oscillator 1 Frequency With Oscillator 3
Bit 7	Modulate Oscillator 1 Frequency With Oscillator 3 Flip

Osc Control

An 8-Bit value who's bits control individual functions of the Oscillator.

	Waveform	
Bit 0-2	000	Sine
	001	Triangle
	010	Saw
	011	Ramp
	100	Pulse
	101	Noise
	110	Level
	111	Vocal
Bit 3		ABS / Full
Bit 4		Ovf / Reset
Bit 5		Apply Bend
Bit 6		No Function
Bit 7		On / Off

Output	signed 8-Bit value. This register contains the final composite oscillator output.
Pitch Bend	signed 8-Bit value that specifies a bend frequency.
Port	An 8-Bit value that specifies I/O bits. (For custom versions)
PWM	signed 8-Bit value. This register linearly controls the pulse width (duty cycle) of the pulse and vocal waveforms. The pulse width resolution allows the width to be smoothly swept with no discernable stepping. Note that the pulse waveform or the vocal waveform must be selected in order for the pulse width registers to have any audible effect. A value of 0 in the pulse width registers will produce a square wave, while a value of -128 (h80) will produce a constant DC output at the level of the oscillator's amplitude.
Speech Control	signed 8-Bit value. This register controls modifications made to the next played Allophone. It is cleared to 0 after each Allophone is played.

Bit 0	Speed Up
Bit 1	Speed Down
Bit 2	Volume Up
Bit 3	Volume Down
Bit 4	Pitch Up
Bit 5	Pitch Down
Bit 6	Bend Up
Bit 7	Bend Down

Speech Frequency Low Speech Frequency High	unsigned 16-Bit value. Together, these registers specify the frequency target for the next phoneme. When a phoneme command has been received, the oscillator presets specified by the phoneme's data are loaded into the appropriate frequency and amplitude targets. In addition, the Speech Frequency is loaded into oscillator B1's frequency target. After all targets are loaded, both mixer A and mixer B sounds are morphed into the new sound resulting in the selected phonetic sound being produced at the specified voicing frequency.
Transition Speed	unsigned 8-Bit value. This register specifies the speed to morph from one sound to the next.

Serial Commands

Serial Commands are the method in which the Sounding is instructed to perform its various functions. When operating, the Soundgin monitors the Serial Input Pin (RX) for incoming serial data. As serial data is received, it is compared against a list of commands and if a match is found, then that command is performed. Commands always begin with the Escape character (Deciamal 27 / Hex 1B) and have at least one command byte. Some of the commands take arguments such as the Write One Byte command. In these cases, the command and all of its arguments must be sent before the command is read out of the input buffer. The following table lists the recognized commands.

Serial Commands Table

h1B, h00	Read One Byte		
h1B, h01	Write One Byte		
h1B, h02	Write Two Bytes		
h1B, h03	Write Three Bytes		
h1B, h04	Write One Byte With Mask		
h1B, h06	Clear Mixers A, B		
h1B, h07	Ramp To Targets A, B		
h1B, h08	Set Voice Note		
h1B, h09	Set Voice Frequency		
h1B, h0A	Set Voice Delay		
h1B, h0B	Set Voice Default Parameters		
h1B, h0E	Turn Q On		
h1B, h0F	Turn Q Off		
h1B, h46	Clear Mixer & Oscillators A	h1B, h66	Clear Mixer & Oscillators B
h1B, h47	Ramp to Targets A	h1B, h67	Ramp to Targets B
h1B, h48	Loads a Musical Note into Oscillator A1	h1B, h68	Loads a Musical Note into Oscillator B1
h1B, h49	Loads a Musical Note into Oscillator A2	h1B, h69	Loads a Musical Note into Oscillator B2
h1B, h4A	Loads a Musical Note into Oscillator A3	h1B, h6A	Loads a Musical Note into Oscillator B3
h1B, h4B	Loads a Sound into Mixer A	h1B, h6B	Loads a Sound into Mixer B
h1B, h4C	Loads a Frequency into Oscillator A1	h1B, h6C	Loads a Frequency into Oscillator B1
h1B, h4D	Loads a Frequency into Oscillator A2	h1B, h6D	Loads a Frequency into Oscillator B2
h1B, h4E	Loads a Frequency into Oscillator A3	h1B, h6E	Loads a Frequency into Oscillator B3
h1B, h50	Plays the Current Oscillator A1 settings	h1B, h70	Plays the Current Oscillator B1 settings
h1B, h51	Plays the Current Oscillator A2 settings	h1B, h71	Plays the Current Oscillator B2 settings
h1B, h52	Plays the Current Oscillator A3 settings	h1B, h72	Plays the Current Oscillator B3 settings
h1B, h53	Plays the Current Mixer A settings	h1B, h73	Plays the Current Mixer B settings
h1B, h54	Releases Oscillator A1	h1B, h74	Releases Oscillator B1
h1B, h55	Releases Oscillator A2	h1B, h75	Releases Oscillator B2
h1B, h56	Releases Oscillator A3	h1B, h76	Releases Oscillator B3
h1B, h57	Releases Mixer A	h1B, h77	Releases Mixer B
h1B, h58	Loads & Plays a Musical Note in Oscillator A1	h1B, h78	Loads & Plays a Musical Note in Oscillator B1
h1B, h59	Loads & Plays a Musical Note in Oscillator A2	h1B, h79	Loads & Plays a Musical Note in Oscillator B2
h1B, h5A	Loads & Plays a Musical Note in Oscillator A3	h1B, h7A	Loads & Plays a Musical Note in Oscillator B3
h1B, h5B	Reset to Voice Parameters	h1B, h7B	Reset to Voice Parameters
h1B, h5C	Loads & Plays a Frequency in Oscillator A1	h1B, h7C	Loads & Plays a Frequency in Oscillator B1
h1B, h5D	Loads & Plays a Frequency in Oscillator A2	h1B, h7D	Loads & Plays a Frequency in Oscillator B2
h1B, h5E	Loads & Plays a Frequency in Oscillator A3	h1B, h7E	Loads & Plays a Frequency in Oscillator B3

Serial Commands Detail

Command	Args	Description
Read One Byte	1	Reads the memory at the location specified by Argument-1. *27,0,24* - Causes Oscillator A1's Amplitude to be sent out TX.
Write One Byte	2	Writes Argument-2 at memory at the location specified by Argument-1. *27,1,24,127* - Sets Oscillator A1's Amplitude to 127.
Write Two Bytes	3	Writes Argument-2,3 at memory at the location specified by Argument-1. *27,1,20,163,1* - Sets Oscillator A1's Frequency Target to 100Hz
Write Three Bytes	4	Writes Argument-2,3,4 at memory at the location specified by Argument-1. *27,1,17,110,163,1* - Sets Oscillator A1's Frequency to 100.00Hz
Write One Byte With Mask	3	Writes Arg-2 masked with arg-3 at memory at the location specified by Arg-1. *27,4,0,2,255* - Causes Oscillator A2 to be included in Mixer A's output.
Clear Mixers A, B	0	Clears memory locations 0-127.
Ramp To Targets	0	Causes the oscillators to calculate a ramp value and then turn on the ramp and target options.
Set Voice Note	1	Sets the Voice frequency to the specified musical note frequency.
Set Voice Frequency	2	Sets the Voice frequency to the specified 16-Bit frequency.
Set Voice Delay	1	Sets the Voice delay to the specified 8-Bit amount.
Set Voice Defaults	0	Sets the Voice options to the default values.
Turn Q On	0	Turns the Q line on.
Turn Q Off	0	Turns the Q line off.
Clear Mixer & Oscillators	0	Clears memory locations for just one mixer and its associated oscillators.
Load a Musical Note	1	Sets the oscillator's frequency to the specified musical note frequency.
Load a Sound	1	Sets the mixer and its associated oscillators to a predefined sound. *27,75,3* - Set the Oscillators in Mixer A to sound 3.
Load a Frequency	2	Sets the oscillator's frequency to the specified 16-bit frequency.
Play Current settings	0	Start the envelope.
Release	0	Release the envelope. *27,84* - Releases Oscillator A1's envelope.
Load & Play a Musical Note	1	Sets the oscillator's frequency to the specified musical note frequency and start the envelope. *27,88,64* - Sets Oscillator A1's frequency to C4 (Middle-C) and starts the envelope.
Load & Play a Frequency	2	Sets the oscillator's frequency to the specified 16-bit frequency and starts the envelope.

Concepts of Sound

The Soundgin is a complex sound generator. It contains six sound generators called *oscillators* which can interact in various ways to produce a wide variety of sounds. In this section, the concepts that the Soundgin incorporates into its oscillators are discussed and defined.

Sound is vibration of the air. It can be compared to throwing a stone into a pool and seeing the waves (vibrations) radiate outward. When similar waves are created in air, we hear it. The speed of vibration is called *frequency*. Frequency is measured in Hertz (Hz). One Hz is equal to one vibration per second. Typically, humans can hear vibrations from 20Hz to 20,000Hz. Sounds generally have many frequency components called *harmonics*. Harmonics contribute to a sound's characteristic tone or timbre.

Two ways to graphically represent sound are as a *waveform* and as a *spectrum*. The waveform is a time graph of the actual shape of the vibration. The waveform of a single frequency is called a sine wave (figure 2-1). Notice that the waveform looks like a cross section of the waves in the pool with the peaks and valleys represented as high and low positions on the line. The spectrum on the other hand shows how strong each of the sound's frequencies and harmonics are. The spectrum of a single frequency is just a single line (figure 2-1).

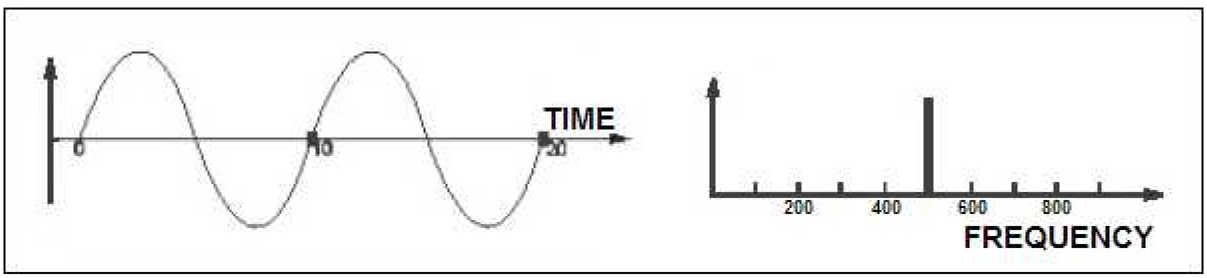


Figure 2-1 - A musical sound can be represented as a waveform (left) or as a spectrum (right).

Notice that the sound in figure 2-1 only has one harmonic in its spectrum. If you listen to a 500Hz sine wave, you hear a whistle sound like a flute approximately an octave above middle C.

An acoustic instrument, like a guitar or a violin, has a very complicated harmonic structure. In fact, the harmonic structure may vary as a single note is played. Notice that the complex sound in figure 2-2 has several oddly placed peaks and valleys on the waveform and has several harmonics of different levels on the spectrum.

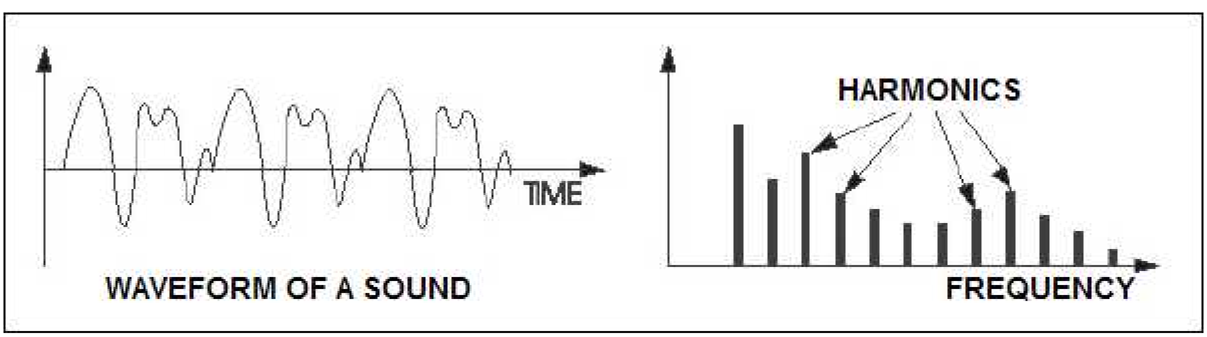


Figure 2-2: The waveform and spectrum of a complex sound.

Each of the Soundgin's six oscillators is capable of producing a single sound wave. These sound waves can then be mixed together in various ways to produce complex sounds.

Spectrograph

A graphical representation of the components of a sound is a Spectrograph. A bar on the graph represents each component of a complex sound. The frequency of a component is indicated by its position to the right or left, and its amplitude (volume) is represented by the height of the bar. The frequencies are marked out in a manner that gives equal space to each octave of the audible spectrum.

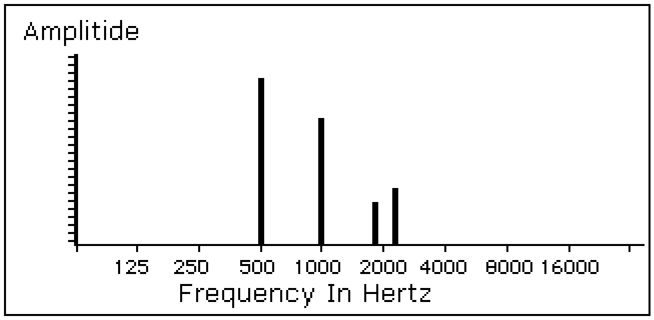


Figure 2-3: A Spectrograph.

Frequency

The frequency of a wave is simply how often it repeats. The perceived highness or lowness of a sound (pitch) is determined by the frequency of the sound waves. For example, a waveform that repeats 261.626 times a second (261.626 cycles per second) has a frequency of 261.626Hz, and is the traditional frequency for the middle C on a piano keyboard.

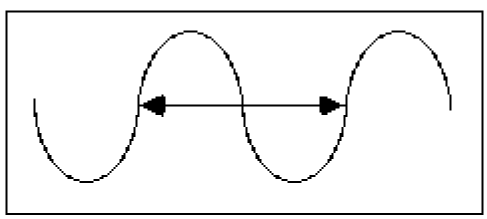


Figure 2-4: The beginning and end of one wave.

Each of the Soundgin's six oscillators can produce a sound wave with a frequency ranging from 0.001hz to 7,812.499hz. The frequency of each oscillator is set by writing to the oscillator's three frequency control registers; Frequency High, Frequency Low and Frequency Fine. The frequency control registers control the speed that the oscillator scrolls through a wave table which produces the desired output frequency.

Hertz

The unit of frequency for sound is Hertz, (abbr. Hz). Hertz is the number of times some event occurs within a second and thusly is also referred to as cycles per second. (c.p.s.) Since frequencies are often in the thousands of Hertz, the unit kilohertz (1000hz=1khz) is used.

Harmonic series

A harmonic series is a group of frequencies constructed by multiplying a fundamental frequency by each integer in turn. The frequency used is the fundamental, and is the first number in the series. Other frequencies are named after their order in the series, so you would say that 1500hz is the third harmonic of 500hz. Octaves are based on Harmonics.

Octave

An octave is an interval between one musical note and another with a frequency ratio of 2:1. For example, the note A4 is at 440Hz. The note one octave above it (A5) is at 880Hz and the note one octave below it (A3) is at 220Hz.

Temperament

Temperament is the set of frequencies commonly used to produce traditional, tonal music. While the concept of the octave is almost universal, the number of other notes that may be found in between is highly variable from one culture to another. For the past hundred and fifty years in Western culture, the tunings of these notes have been standardized as dividing the octave into twelve equal steps (called semi-tones). Equal temperament has a variety of advantages over the alternatives, the most notable one being the ability of simple keyboard instruments to play in any key.

The Soundgin has a built -in lookup table of the A440 temperament frequencies . Each oscillator can be set to a specific note in this table by issuing either the Play Note or Set Note command.

Beats

If two sounds have nearly the same frequency and they are mixed together, their combined amplitude will fluctuate up and down at a rate equal to the difference frequency between them. This phenomenon is called "beating ". A beating waveform looks very much like amplitude modulation. It differs in that only the two frequency components that are combined. There are no new frequencies such as sidebands created.

Synchronization

Synchronization is an effect where one wave starts over at the beginning of its wave at the frequency of a second wave thereby causing it to have two composite frequencies.

Oscillators A2 and B2 can be used to synchronize Oscillators A1 and B1 respectively. In addition, B1 can be used to synchronize oscillators A1, A2 and A3 collectively. When the synchronizer oscillator's value crosses zero (0), the synchronized oscillator restarts from the beginning.

Synchronization is selected by setting the appropriate bits in the Mixer Control Registers.

LFO

LFO stands for Low Frequency Oscillator. LFO's are generally used to create slow modulations such as vibrato and tremolo.

All six of the Soundgin's Oscillators are capable of producing very low frequencies. Oscillators A2 and B2 can be used for tremolo and Oscillators A3 and B3 can be used for vibrato.

Amplitude

The peak amount of change in pressure (or the electrical equivalent, voltage) is the amplitude of the wave (Figure 2-5). Amplitude includes both positive and negative swings around the average pressure (Zero-Crossing). The amplitude of sound is heard as the loudness, or volume.

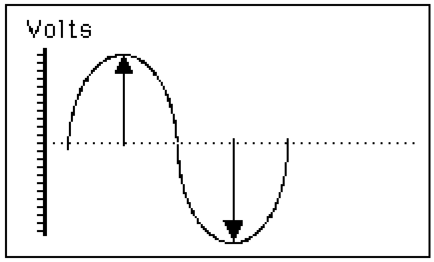


Figure 2-5: The top and bottom of a wave.

Each of the Soundgin's six oscillators applies its amplitude value by multiplying the specified amplitude level by the current wave table sample. In the cases where the amplitude is being modulated, the sample rate is further multiplied by the modulating signal. If the modulating signal cycles between positive amplitude and negative amplitude, the resulting output signal will be ring modulated. This concept is covered further in the *ring modulation* section. The final output amplitudes of the oscillators are then mixed together by mixers which have their own amplitude controls.

The amplitude of each oscillator is set by writing to the oscillator's amplitude control register: Amplitude. Note that writing a negative value to the amplitude register will invert the output amplitude. This means that the first amplitude peak will occur below the zero-crossing. Acoustically, this makes no difference in the way the wave sounds. It does however make a significant difference when using the Level Waveform.

Waveform

The repeating change in pressure (or the electrical equivalent, voltage) over a single cycle of the sound is the waveform. The waveform is visually represented by the shape of the wave.

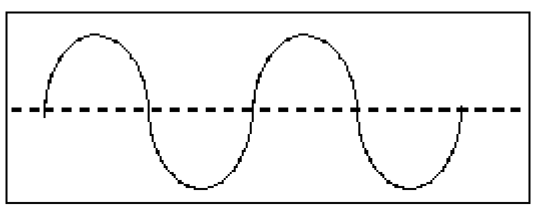


Figure 2-6: The waveform of a wave.

The curvy line shown in Figure 2-6 represents the changes in pressure that occur in a sine wave. Time is along the horizontal axis, but units are usually not indicated, as the waveform of a sound is independent of its frequency. A waveform graph is always at least one complete period.

The dotted line is the Zero-Crossing of the signal. The portion of waveform above the Zero-Crossing represents the positive energy and the portion below the Zero-Crossing represents the negative energy. Typically, as the amplitude increases, both the positive and negative energies correspond equally. It is also possible to force the entire waveform to be above or below the Zero-Crossing with the ABS control

Each of the Soundgin's six oscillators can produce any one of 7 different waveforms; Sine, Triangle, Sawtooth, Ramp, Pulse (square), Noise, and Level. One of the Soundgin's oscillators can produce an eighth shape, Vocal . The waveform of each oscillator is set by writing to the oscillator's control register, Osc Control. The Osc Control register controls which wave table is used when generating the oscillator's final output.

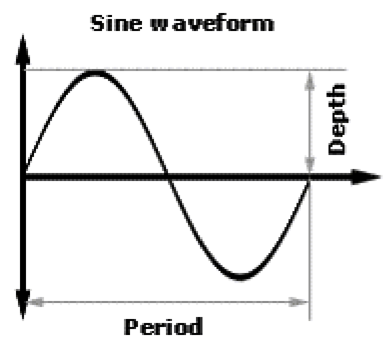
Timbre

The tonal quality of a sound is called the timbre. The timbre of a sound is determined primarily by its waveform, and also by its harmonic structure. Timbre is why the sound of a trumpet playing a tone is distinguishable from a flute playing the same tone.

Sine Wave

The sinusoidal wave, or sine wave for short, is a very smooth waveform typical of flute and whistle sounds.

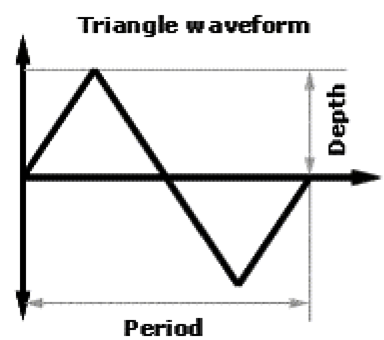
The sine wave is the wave produced by a single frequency and therefore has a single harmonic.



Triangle Wave

A very smooth waveform typical of flute and whistle sounds. A triangle wave looks a bit like a sine wave with straight lines. It slowly falls then slowly rises, making it look like a zigzag shape.

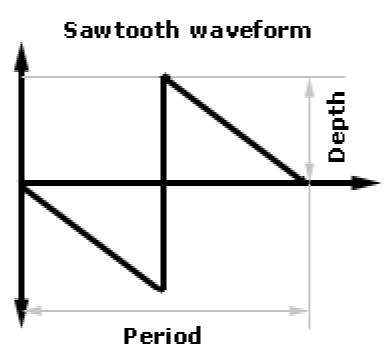
Only odd harmonics are present in the triangle wave. The amplitude of each harmonic, however, is 1/n^2. This causes the higher harmonics to be much quieter than they are in a square wave. For this reason, the triangle wave can be a good starting point for emulating bass instruments.



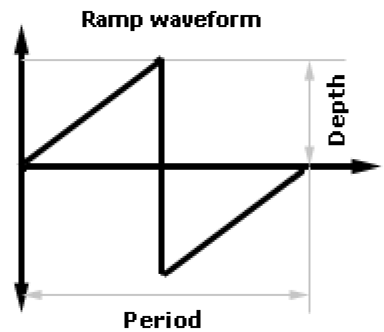
Sawtooth Wave

A very sharp waveform typical of trumpet-like sounds. As its name suggests, the sawtooth wave resembles the teeth of a saw. It rises quickly and slowly falls.

It includes all harmonics. This means it includes the fundamental harmonic, and another sine wave at twice its speed, and another at three times its speed, and so on. The second sine wave is half as loud, as well as twice as fast. The third one is a third as loud, as well as three times the speed.



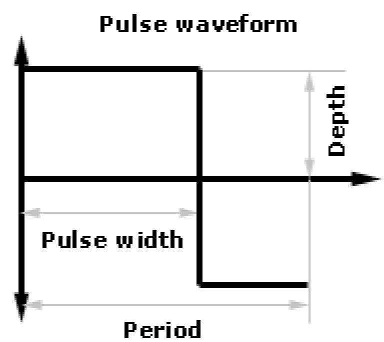
Ramp Wave



A modification of the sawtooth wave, the ramp wave slowly rises then quickly falls.

The Ramp Wave and the Saw Wave sound exactly the same, so the difference is mainly used when modulating other sounds.

Pulse Wave

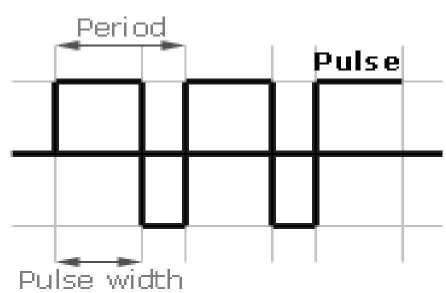


A pulse wave is a type of waveform that only has two possible states: "on" or "off".

The width of the "on" pulse can vary. The amount of time a pulse wave spends in the "on" position is called its duty cycle. A pulse wave, which spends a quarter of the time on, then three quarters off before repeating, is said to have a 25% duty cycle.

The duty cycle of the pulse width is controlled by the PWM setting.

PWM



Pulse width modulation (PWM) is the variation of the pulse width on a pulse wave. The Pulse width (duty cycle) is another component of the timbre of the sound.

A pulse wave with a 1% duty cycle sounds different from a pulse wave with a 50% duty cycle. This is because they are different shapes, and therefore have different harmonic contents. Once the halfway mark is reached, though, the sounds repeat - a 55% duty cycle sounds the same as a 45% one, and a 66% duty cycle sounds the same as a 33% one. A duty cycle of exactly 0% or 100% would be completely quiet, as the waveform would just be a straight line.

Changing the duty cycle over time creates an evolving timbre. This is called pulse width modulation. By changing the pulse width, the timbre of the sound produced by the Pulse wave can be varied tremendously.

Each of the Soundgin's six oscillators have a PWM Register that can be set to control the pulse width. This register affects the Pulse Waveform and the Vocal Waveform. It has no affect if any other Waveform is selected. The PWM Register is a singed value with a range of +/-127 where a value of 0 gives a duty cycle of 50%. For more information, see the PWM Register description.

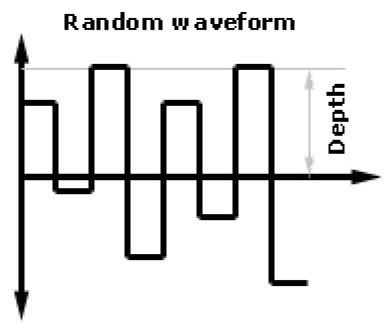
The Pulse Width can also be modulated by the output of another oscillator by turning on the appropriate bits in the Mixer Control Registers.

Square Wave

A square wave is one example of a pulse wave. To be exact, a square wave is a pulse wave which spends exactly the same amount of time in the off position as it does in the on position. Setting the pulse width to its center position results in a clear 50% square waveform. The square wave is a pulse wave with a duty cycle of 50%. It only has odd harmonics (the first sine wave, the third one, the fifth and so on), but they are still 1/n where n is the number of the harmonic. So a square wave is like a ramp wave, only with every even numbered harmonic missing. This makes it sound hollow in comparison.

Setting the oscillator's PWM Register to 0 causes the pulse waveform to be square. When the Soundgin is reset or it receives the Reset Command the PWM Registers are all set to 0.

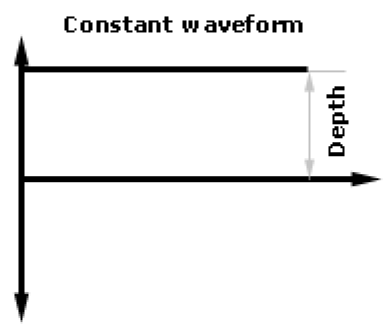
Noise Wave



The noise waveform is a modified square wave where, each half cycle of the wave has its amplitude level set to a random value between 0 and full amplitude. The first half of the wave will always be a positive random value and the last half of the have will always be a negative random value.

The random waveform is widely used for special effects, like drums, wind, and other interesting noises. It has the retro Atari noise sound.

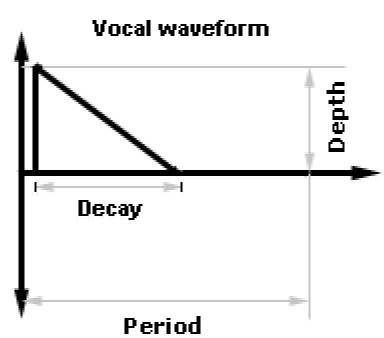
Level Wave



A waveform that maintains the amplitude level with no change.

If the amplitude level is a positive value then the waveform is on the positive energy side of Zero-Center and if the amplitude level is a negative value, then the wave is on the negative energy side of Zero-Center.

Vocal Wave



A waveform that emulates the waveform produced by the human vocal cords.

The Vocal waveform is similar to the Ramp Waveform except that it has a specific decay time that is independent of the period. It sharply rises and then slowly falls at a decay rate. Once it has reached zero, it remains there until the period of the wave is reached at which time the wave repeats. The Vocal waveform is a positive only wave which will never have any negative energy.

The decay rate is controlled by oscillator B1's PWM Register.

ABS

ABS is a waveform option that controls where the energy of the wave resides. Typically a waveform will oscillate between positive energy and negative energy. This is called a *full wave* and is shown on the left side of figure 2-17.

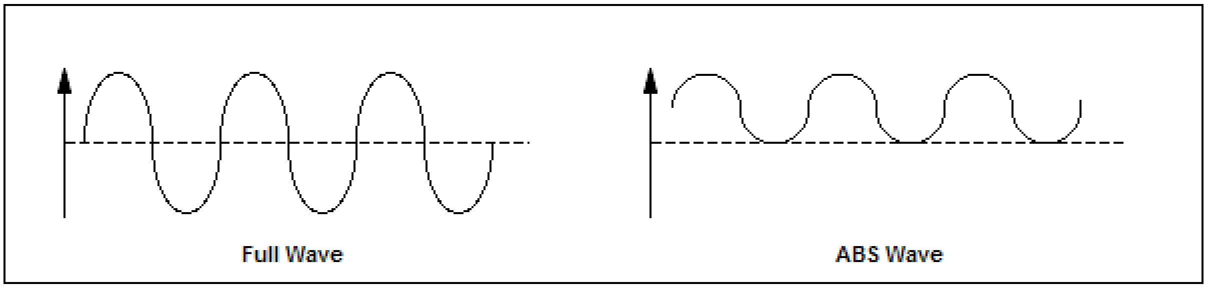


Figure 2-17. Two waves. One at full amplitude. One at ABS Amplitude.

Each oscillator has an ABS option that can be turned on which shifts the wave so that it falls entirely within one side. If the amplitude value is positive, the wave will be generated with only positive energy and if the amplitude value is a negative value, the wave will be generated with only negative energy. Regardless of the side the wave falls on, one edge of the wave will always be touching the Zero-Crossing. Absolute waves are typically used for modulating other waves.

A wave that has the ABS option turned on is shown on the right side of Figure 2-17.

Absolute Value is controlled by the ABS bit in the Osc Control Register.

Envelope

The amplitude's rise and fall during the playing of a note is called an *Envelope*. When a note is played on an instrument, the volume of the note changes from the moment you first hear it, all the way through until it dies out and you can't hear it anymore.

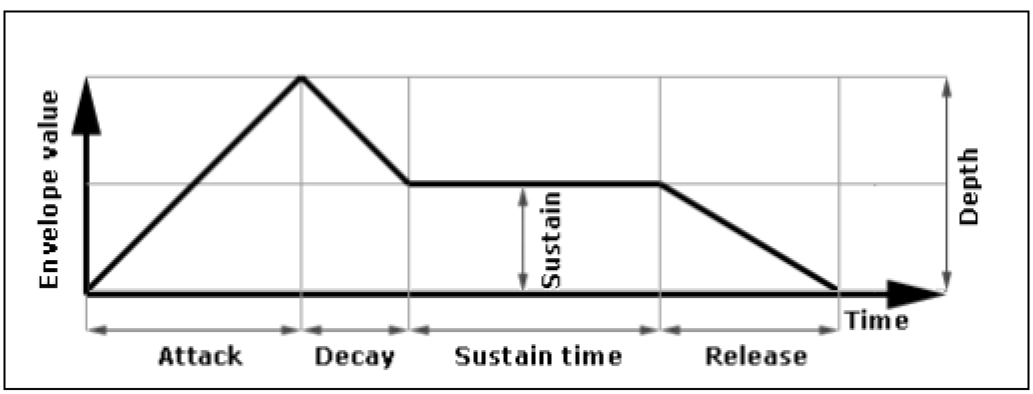


Figure 2-18: A visual representation of a sound envelope.

When the note is first struck, it rises from zero volume to its peak volume. The rate at which this happens is called the ATTACK. Then, it falls from the peak to some middle-ranged volume. The rate at which the fall of the note occurs is called the DECAY. The mid-ranged volume itself is called the SUSTAIN level. And finally, when the note stops playing, it fails from the SUSTAIN level to zero volume. The rate at which it fails is called the RELEASE.

In Figure 2-18, a typical ADSR (attach/Decay/Sustain/Release) envelope is shown that emulates a note being played. The amplitude level starts from a zero amplitude, builds quickly to a peak, falls to an intermediate value and stays near that value until the note is released, then falls back to zero.

Each of the Soundgin's six oscillators as well as two mixers have independent envelopes. The envelopes are controlled by writing to the Envelope control registers: Envelope Control , Envelope Attack, Envelope Decay and Envelope Release.

The envelopes for the oscillators and the mixers work identically ecept that the envelopes for the mixers will envelope the final output of mixing all three oscillators.

Attack

The envelope enters the attack phase when the envelope is instructed to begin by the Play Note command. In the attack phase, the envelope's corresponding amplitude level is adjusted towards the attack level at the rate specified by the attack rate. (Note that the adjustment can be either positive or negative depending on the attack level specified.) When the corresponding amplitude level matches the attack level, the envelope control shifts to the decay phase.

The Envelope Attack registers specifies the attack Level and how fast/slow the amplitude will reach the specified Attack Level.

Decay

The envelope enters the decay phase immediately after the attack phase. In the decay phase, the envelope's corresponding amplitude level is adjusted towards the decay level at the rate specified by the decay rate. (Note that the adjustment can be either positive or negative depending on the decay level specified.) When the corresponding amplitude level matches the decay level, the envelope control shifts to the Sustain phase.

The Envelope Decay register specifies the decay Level and how fast/slow the amplitude will move towards the decay level.

Sustain

The envelope enters the sustain phase immediately after the decay phase. In the sustain phase, the envelope's corresponding amplitude level sustains its current level. The envelope remains in the sustain phase until the Release Note Command is received at which time the envelope control shifts to the release phase.

Release

The envelope enters the release phase immediately upon receiving a Release Note Command . In the release phase, the envelope's corresponding amplitude level is adjusted towards the release level at the rate specified by the release rate. (Note that the adjustment can be either positive or negative depending on the release level specified.) When the corresponding amplitude level matches the release level, the envelope control shifts to an off state.

The Envelope Release register specifies the release level and how fast/slow the amplitude will move towards the release level.

Modulation

The process in which one waveform is changed in response to the contour of another waveform is Modulation. For instance, amplitude modulation is the shaping of the amplitude of one waveform by the contour of another waveform.

The Soundgin is capable of Amplitude , Ring, and Frequency Modulation.

Amplitude Modulation

Amplitude modulation is defined as the multiplication of one time-domain signal by another time-domain signal. Any analog signal can amplitude-modulate any other analog signal. A common example of amplitude modulation in electronic music is tremolo. Tremolo is the amplitude modulation of an instrument signal by a sine wave of a few Hz or so.

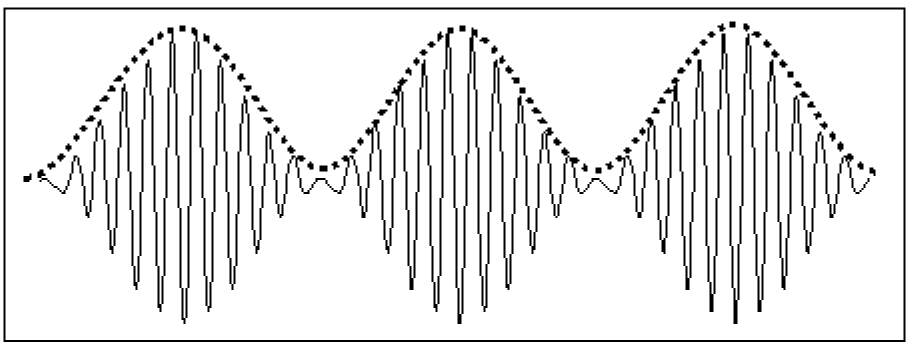


Figure 2-19: One wave being amplitude modulated by another.

Figure 2-19 shows two frequencies. The higher frequency is being amplitude modulated by the second lower frequency. Notice that the amplitude of the higher frequency (the modulated wave) is following the waveform of the lower frequency (the modulating wave). Also notice that the waveform of the modulating wave is entirely positive energy. This waveform option is controlled by the oscillator's ABS Control which causes the wave to output an Absolute Value. This causes the modulated wave's amplitude to go to zero when the the modulating wave is at its lowest point. If the modulating wave was not outputting an Absolute Value, then the waves would be ring modulating.

Amplitude Modulation is inherently a non-linear process, and always gives rise to frequency components that did not exist in either of the two original signals.

Oscillators A2 and B2 can be used to modulate the amplitude of Oscillators A1 and B1 respectively creating a tremolo effect.

Tremolo

Tremolo is a musical effect where the the amplitude or volume of a note or sound is quickly and repeatedly raised and lowered over a small distance for the duration of that note or sound. True tremolo is a low frequency periodic fluctuation in the amplitude of a sound. It is commonly confused with vibrato, which is a periodic fluctuation in the frequency.

Ring Modulation

Ring modulation is a special type of amplitude modulation in which two input signals are processed in such a way that the sums and differences of the input frequency components are generated while the input signals themselves are suppressed. For instance, if you were to put in two sine wave tones at 500Hz and 100Hz, the output would comprise tones at 400Hz and 600Hz.

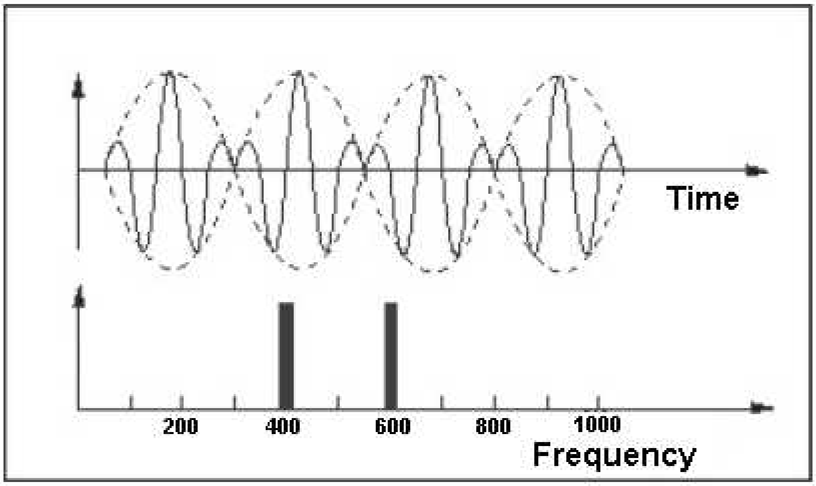


Figure 2-20: One wave being ring modulated by another.

Ring modulation generally produces atonal, non-musical sounds, which makes it popular for science-fiction special effects. Detuning the two inputs by a very small amount can produce unusual low-frequency beating effects. When harmonically rich sounds are used, all those harmonics contribute to the sum-and-difference process, resulting in a harmonically complex output.

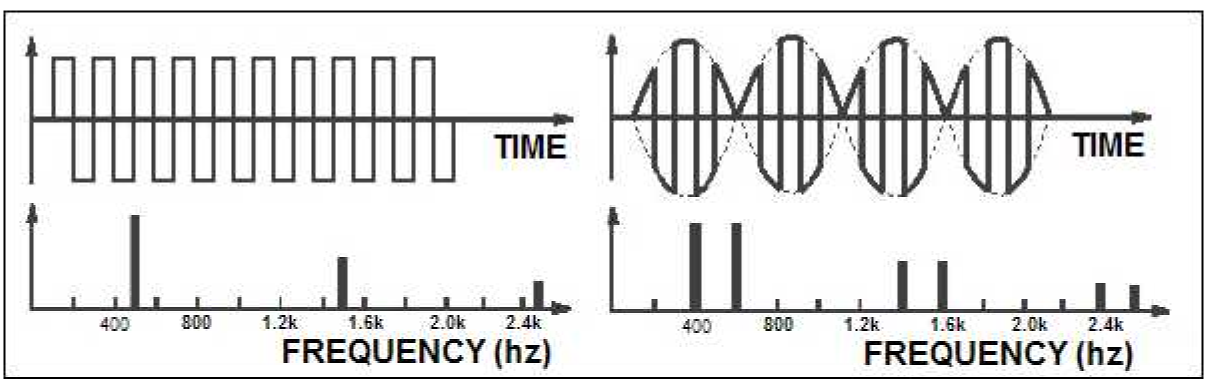


Figure 2-21: A square wave being ring modulated by a sine wave.

Figure 2-21 shows what happens when the first input is a 500Hz square wave and the second input is a 100Hz sine wave. Figure 2-21-Left shows the waveform and spectrum of a 500Hz square wave. This waveform has only odd harmonics. The first harmonic (usually called the fundamental) is 500Hz, the third harmonic is 1,500 Hz, the fifth harmonic is 2,500 Hz, and so on. This set of harmonics produces a tone that sounds bright and hollow. When this waveform is ring-modulated by a 100Hz sine wave, the resultant spectrum contains components whose frequencies are 400Hz, 600Hz, 1,400Hz, 1,600Hz, 2,400Hz, 2,600Hz, and so on. Figure 2-21-Right shows the waveform and spectrum of the ring modulation of a 500 Hz square wave by a 100 Hz sine wave.

Oscillator Pairs A1 -A2 and B1-B2 can be ring modulated. In addition, the output of Mixer A can be ring modulated with the output of Oscillator B1. To select Ring modulation, both of the input sources must have the ABS and the sync options off. If the ABS option of either input is on, the the pair will be Amplitude modulated instead.

An additional effect results when one of the input sources to the ring modulation is also frequency modulated. This causes the sum and difference frequencies to modulate toward and away from each other at the rate of the frequency modulation. For example using the Oscillator Pair A1-A2 in ring modulation while A1 is frequency modulated by A3.

Frequency Modulation

Frequency modulation is a variation of one wave's frequency caused by the amplitude of a second wave. A common example of Frequency Modulation in both electronic and acoustic music is Vibrato.

Oscillators A3 and B3 can modulate the frequency of Oscillators A1 and B1 respectively creating a Vibrato effect.

Vibrato

Vibrato is a musical effect where the pitch or frequency of a note or sound is quickly and repeatedly raised and lowered over a small distance for the duration of that note or sound. Vibrato is typically used to add expression and vocal -like qualities to instrumental notes.

Sample Rate

The rate at which the oscillator's amplitude levels are generated is called the *Sample Rate*. 15,625 times each second, each of the Soundgin's six oscillators calculate a new amplitude level. The levels are mixed in accordance with the mixer settings and then the final output is sent out the Soundgin's Audio output pin.

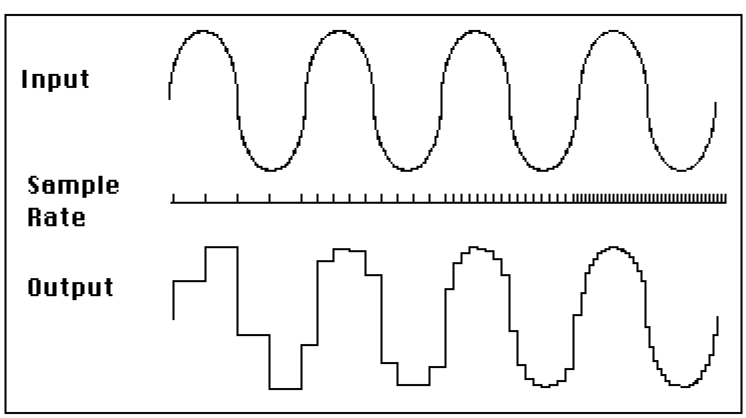


Figure 2-22. Output waveforms as the sampling rate is increased.

The desired output frequency for each oscillator can be at most 1/2 the sample rate for accurate results. If the frequency specified by the frequency registers is more than 1/2 the sample rate, the output of the oscillator it will not resemble the proper waveform at all. This kind of mistake is called aliasing.

The Soundgin's Sample Rate is 15,625Hz, therefore, the maximum output frequency is 7,812.499hz

Wavetable

A list of numbers that contain all values corresponding to the desired waveform is called a *Wave Table*. In order for the Soundgin to create an output sound wave, it reads the numbers from a wave table at a steady rate (the sampling rate), repeating the table when the end is reached.

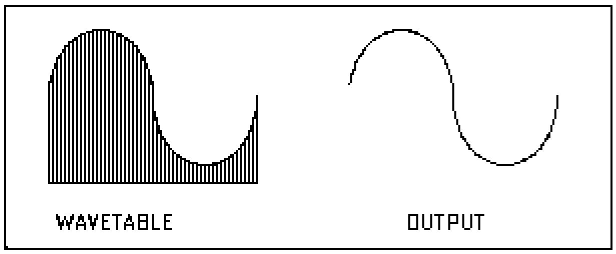


Figure 2-23: The values in the Wavetable create the desired waveform.

The oscillator's three frequency control registers, Frequency High, Frequency Low and Frequency Fine, control the speed at which the oscillator scrolls through the wave table, producing the desired output frequency. To produce higher pitches, the Soundgin skips some values each time. The number of values skipped is the sampling increment. A sampling increment of 4 (reading every fourth value) gives an output two octaves higher than the original.

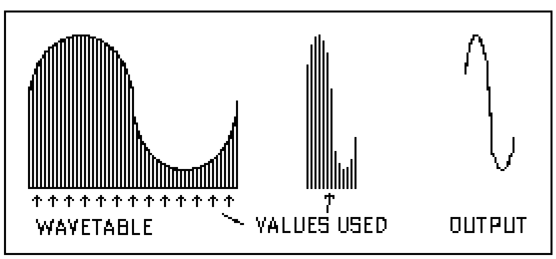


Figure 2-24: Effect of increases sampling increment.

To produce lower pitches, the Soundgin reads a number twice (or more).

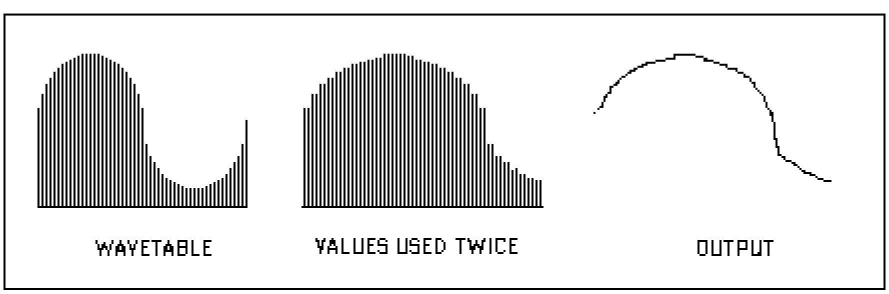


Figure 2-25: A sampling increment of .5.

The Soundgin uses a 24-Bit number to keep track of the current position in the wave table. For each sample, the 24-Bit Frequency register is added to this 24 -bit number, and an amplitude is read from the wave table. As the 24-Bit number overflows, it truncates any bits above the 24th bit. The resulting number is an acoustically accurate position in the wave table.

For waveforms other than the sine wave, a cleaner sound can be achieved by using the OVF method. The OVF (Overflow) method resets the 24-bit number each time an overflow occurs. The OVF is controlled with the OSC Control register.

In order for oscillator synchronization to occur, the oscillator that is being used as the synchronizer must have its OVF control turned on.

Mixer

Mixers add waveforms together so that all waveforms are represented in a single new complex waveform.

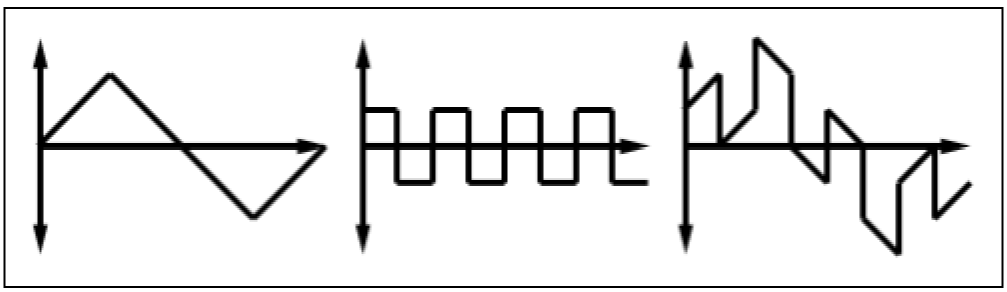
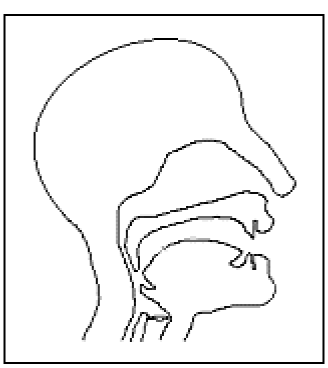


Figure 2-26. Two waves being mixed.

The Soundgin contains three mixers: A, B and C. The first two mixers mix the outputs of three oscillators and then apply an amplitude to its output. Each oscillator can be included or excluded from the mix by turning on the MIX control. The third mixer mixes the outputs of the fist two mixers and then applies an amplitude to its output which is then sent out the sound-output pin.

Human Speech:



The sounds we make when we speak are created in primarily two different ways.

The first method of making sounds uses your vocal cords and vocal tract. The vocal cords vibrate at a frequency determined by the tension in the muscles that control them, which produces a Voicing Source sound in the form of pulses of air. The Voicing Source is in turn released into the vocal tract. The vocal tract behaves like a variable filter. It is variable because by changing the position of your tongue, jaw, etc., you can adjust the frequency response. The vocal cords and the vocal tract produce a sound output with resonance at peaks called the formants (F1, F2 and F3). Sounds such as e, a and u are produced by this method. The frequencies of the formants determine which sound is made. These sounds are called *resonate* sounds.

The second primary method of making sounds is a wide range of hissing or wind noises created by passing air through a small aperture between the lips, teeth, etc. These sounds are all caused by the turbulent flow of the air, and they contain a wide range of different frequencies. Sounds such as f, sh and s are produced by this method. Turbulence at a small constriction produces broadband sound with characteristic frequencies. These sounds are called *broadband* sounds.

Resonate and broadband sounds are also often combined as in the case of the v, z, and j sounds. Some sounds such as b, d and k transition between the two sounds. when they beginning a word, they have a short burst of broadband sound then a characteristic transient (as the constriction opens) to the following resonate sound. When the same sounds are used at the end of a word they have a transient (as the constriction shuts) followed by short silence and then the broadband sound.

Phonemes of English Language

The sounds we make when we speak are called Phonemes. In American English there are 40 discernable Phonemes.

Vowels are all created by resonant sounds. They are classified according to whether the front or back of the tongue is high or low (See Figure 2-27), whether they are long or short, and whether the lips are rounded or un-rounded. In English all rounded vowels are produced in or near the back of the mouth (UE, OO, AA, AU).

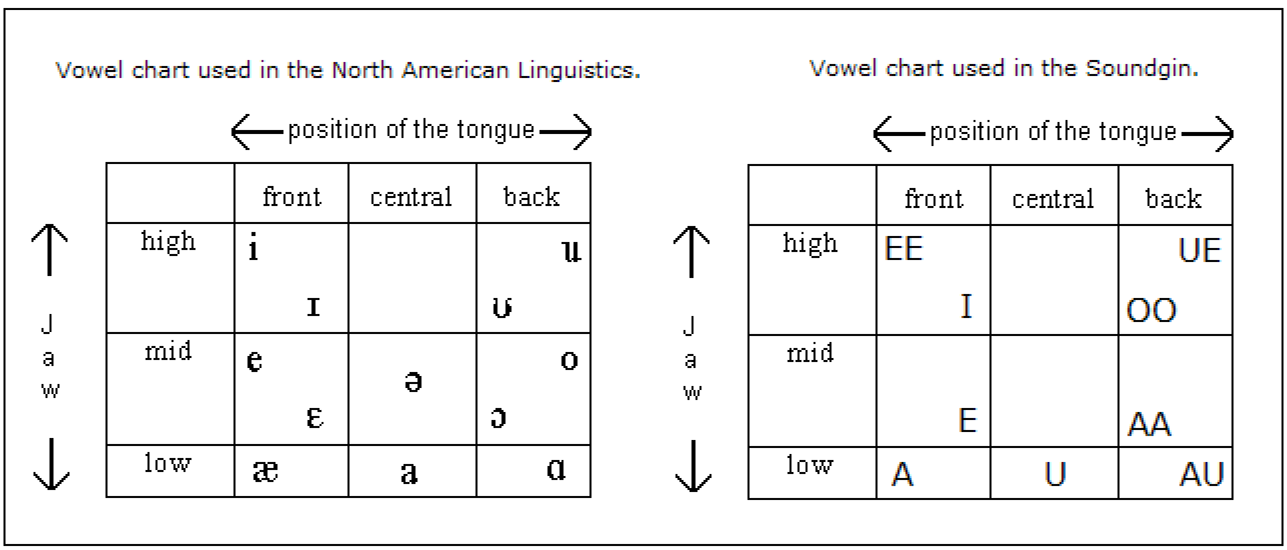


Figure 2-27:   Vowel Chart.

Consonants are created by the broadband sounds either alone or in combination with resonate sounds.

During the articulation of words, variations of the phonemes are vocalized one after another producing the spoken words and phrases. Because variations of these phonemes are also recognized as parts of the words and phrases, more sounds than the 40 Phonemes are needed to synthesize intelligible speech. The set of phonetic variations that the Soundgin uses is called *Allophones*.

Allophone Speech Synthesis

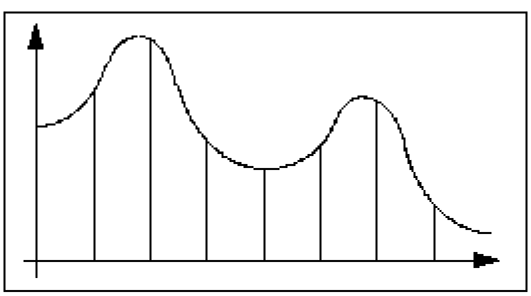
Allophone speech synthesis is a method of synthesizing a reproduction of a language by providing a technique of producing phonemic like sounds, one after another in much the same way that the human mouth does.

The Soundgin's implementation of Allophone Speech Synthesis is done entirely by emulating the phonemic sounds using the oscillators. For each Allophone, the Soundgin's oscillators are set to values that mathematically model the sounds that the human mouth produces for that phonemic sound. As the Soundgin receives each new Allophone, the oscillator's values are smoothly ramped from the previous Allophone settings to the new Allophone settings using the Soundgin's sound morphing function. This emulates the human mouth, which moves from one position to the next. The oscillator settings for each of the 58 Allophones are stored in the Soundgin's Allophone library

To produce speech, a list of selected allophones is sent to the Soundgin. As the Soundgin is vocalizing this list of allophones, it actively and continuously calculates all the sound components of the allophones including the transitional sounds made between the allophones, producing the same sounds that the human mouth does as it moves from one position to another position. Selecting the appropriate combination of allophones and pauses can thusly create any English word or phrase. Further tuning with the Rate, Pitch and Volume parameters adds to the delivery of the phrase and can change the emotion in which the phrase is perceived.

Changing the Rate, Pitch, Bend, Density, and Volume along with the other parameters of the oscillators to settings beyond the capabilities of the human mouth can result in some interesting sound effects that can still be understood as speech. The result is a system that gives the user the ability to not only produce an unlimited vocabulary, but also to produce slang, gibberish, moans, groans, yodels and other weird vocalized sounds not normally included in a canned TTS system.

Emulating Phonemic Sounds



To produce speech, the Soundgin sets the oscillators in such a way that they emulate the functions of the vocal cord and vocal tract which then produce the phonemic sounds of human speech. This emulation is accomplished in three parts:

1. The Voicing Source is emulated by using Oscillator B1 with a Voice Waveform to produce a vocal cord sound. 2. The Resonate Phonemes are emulated by using Oscillators A1, A2 and A3 to produce the resonate peaks of the formants F1, F2 and F3. Mixer A is then amplitude modulated by the Voicing Source (Oscillator B1) to create the effect that an active filter is filtering the Voicing Source sound and thus producing the formants. 3. The Fricative and Plosive Phonemes are emulated by using Oscillators B2 and B3 with different frequencies and distortion combinations.

Presets for all of the phonemic sounds listed in the Allophone Table are stored in the Soundgin and can be loaded into the Soundgin's oscillators by sending the appropriated commands. Loading the allophone presets in series produces speech output with an unlimited vocabulary. Note that when the allophone presets are loaded, only the minimum information required to produce the phoneme is loaded into the oscillators. This allows the voice to be modified in various ways to create a wide variety of voiced sound effect.

Producing English Speech

Five basic linguistic concepts will help you to create highly intelligible sounding speech with the Soundgin:

1. There is no one-to-one correspondence between written letters and speech sounds. More than one letter may represent each sound in a language and, conversely each letter may represent more than one sound. (See the sample words in the Phoneme Table .) Because of these spelling irregularities, it is necessary to think in terms of sounds, not letters, when creating phrases.
2. Speech sounds are acoustically different depending upon their position within a word. For example the initial B sound in "Beep" is acoustically different from the B sound in "Box" The B sounds differ due to the influence of the vowels, which follow them.
3. The human ear may perceive the same acoustic signal differently in the context of different sounds. Therefore, an allophone may sound slightly different when used in various phrases.
4. Some sounds in words are not actually pronounced and/or others are added when followed or preceded by certain other words. For example the two words “Night” and “Time”. When played separately, they both will articulate the “T” sound. However, played together they need to be articulated as “Ni-Time".
5. The Vocalization Pitch, Play Rate and Frequency Bend parameters are just as important as the selection of the phonemes used. For instance, playing a phrase that does not change the pitch at any point in time sounds very monotonic and robotic, while increasing the Pitch at the end of a sentence produces a questioning tone.

How to Use the Allophone Set

Phonemes are acoustically different depending upon their position within a word. Each of these positional variants is an allophone of the same phoneme. An allophone, therefore, is the manifestation of a phoneme in true speech signal. The following is a brief tutorial on creating speech using the Soundgin allophone set. Note that these are suggestions, not rules.

The Phonetic Usage Chart found in the Allophone Table references the variations of each phoneme of the English language and gives suggestions for the appropriate times that they are used. For example, there are 4 B sounds. “BE” and “BO” sound good in initial position, as in the words "Beep" and "Box" while “EB” and “OB” sound good in final position, as in "Rib" and "Club". A vowel modifier is also associated with the “BE”, “BO”, “EB” and “OB” sounds. The BE is used when followed by a front vowel sound and the BO is used when followed by a back vowel sound. Note that either can be used when followed by a central vowel sound. Also note that a B sound in the middle of a word can be either an initial position or a final position sound. For example, the word rabbit; is it RAB-IT or RA-BIT? Local dialect can be the deciding factor in cases like this.

Typically the first allophone in a phrase is longer than the rest of the allophones used with the phrase. Therefore, to create an initial SS, you can use the SLOW function before the SS allophone, which will cause the SS sound to play for 1 and 1/2 the normal speed. If an even longer sound is needed then the SS allophone can be played twice. (Note that this cannot be done with all allophones.) Repeating Diphthongs for example will cause the diphthong slide to play twice which will not result in just a longer sound. Diphthongs are perceived as single vowel sounds, but in reality they are sliding sounds that transition from one vowel sound into another vowel sound. For example the “A” sound in "Make" isn't a single vowel sound. It is a diphthong that transitions from one sound to another. This sliding sound is repeated when diphthongs are repeated. For example: "AE, AE" does not have the same affect as "EE, EE". Two "EE" s played back to back will produce

Three of the "R" sounds are diphthongs that begin with a vowel and end with the "R" sound. For example, the "AIR" in "Hair" and the "OR" in "score". Determining when to use a diphthong and when to use the independent sound again largely relies on personal preference and local dialect.

Some sounds (P, B, T, D, K, G, and CH) require a brief duration of silence before them. For these, the silence has already been added but you may decide you want to add more. To increase the duration of silence the "SLOW" command can be used and optionally a pause can be inserted before it. There are several different types and lengths of pauses that can be inserted in front of an allophone but be aware that different pauses will affect the way the Soundgin calculates the transitions from the last allophone, through the silence and into the next allophone which may not produce the desired flow of sounds. You may want to add a short pause between words and a long pause between clauses.

Stress can be accomplished in two ways. One is to cause vowels to play for a longer period of time. For example, in the word "extent" use the "Fast" command in front of the "EH" in the first syllable, which is unstressed and a "SLOW" command, or an additional "EH" in front of the "EH" in the second syllable which is stressed. A second way is to preceded the allophone with the "STRESS" and RELAX commands. The STRESS command duplicates the affect of slightly contracting the muscles of the mouth and the relax command duplicates the affects of slightly relaxing the muscles of the mouth. For example: "STRESS, E" sounds more like (but not quite) the "I" sound. Likewise, "RELAX, I" sounds more like (but not quite) an "E" sound. Note that if you elect to use the "STRESS" or "RELAX" command in combination with a phoneme that has been doubled, then two commands will be needed, one in front of each of the phonemes.

Remember that you must always think about how a word sounds, not how it is spelled. For example, The “N” sound in "Link" is actually the "NGE" sound and the ending sound in the word "letter" uses the diphthong "ER". Some sounds may not even be represented in words by any letters, as the "Y" in "computer”. You will want to experiment with all the possible sounds to discover which version works best in any particular cluster of allophones.

Envelopes

The four-part ADSR (Attack, Decay, Sustain, Release) envelope generator provides an optimum trade-off between flexibility and ease of amplitude control. Appropriate selection of envelope parameters allow the simulation of a wide range of percussion and sustained instruments. The violin is a good example of a sustained instrument. The violinist controls the volume by bowing the instrument. Typically, the volume builds slowly, reaching a peak then drops to an intermediate level. The violinist can maintain this level for as long as desired, then the volume is allowed to slowly die away. A snapshot of this envelope is shown below:

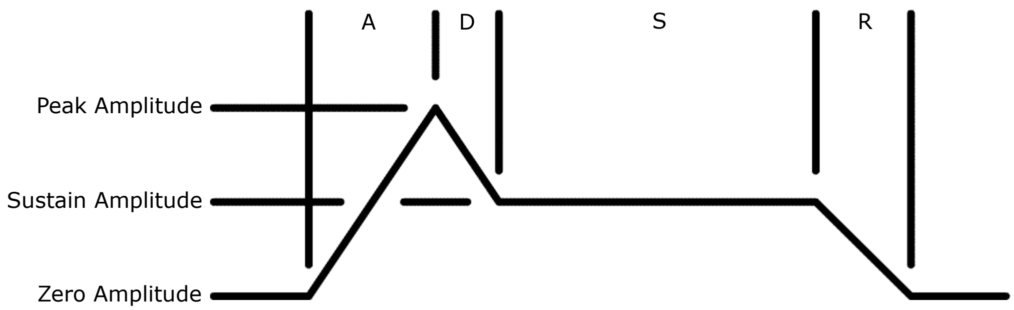


Figure 4-1 ADRS Envelope.

This volume envelope can be easily reproduced by the ADSR as shown below, with typical envelope rates:

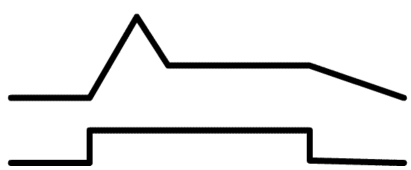


Figure 4-2 Envelope with typical rates.

Note that the tone can be held at the intermediate sustain level for as long as desired. The tone will not begin to die away until the envelope is released. With minor alterations this basic envelope can be used for brass and woodlands as well as strings.

An entirely different form of envelope is produced by percussion instruments such as drums, cymbals and gongs. The percussion envelope is characterized by a nearly instantaneous Attack, immediately followed by a decay to zero volume. Percussion instruments cannot be sustained at a constant amplitude. For example, the instant a drum is struck, the sound reaches full volume and decays rapidly regardless of how it was struck. A typical drum envelope is shown below:

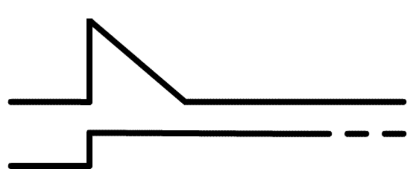


Figure 4-3 Envelope for drums.

Note that the tone immediately begins to decay to zero amplitude after the peak is reached, regardless of when the envelope is released.

The amplitude envelope of pianos and harpsichords is somewhat more complicated, but can be generated quite easily with the ADRS. These instruments reach full volume when a key is first struck. The amplitude immediately begins to die away slowly as long as the key remains depressed. If the key is Released before the sound has fully died away, the amplitude will immediately drop to zero. The piano envelope is shown below:

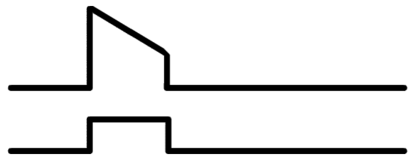


Figure 4-4 Envelope for piano.

Note that the tone decays slowly until the envelope is released, at which point the amplitude drops rapidly to zero.

The most simple envelope is that of the organ. When a key is pressed, the tone immediately reaches full volume and remains there. When the key is Release, the tone drops immediately to zero volume. This envelope is show below:

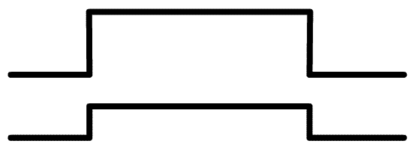


Figure 4-5 Envelope for organ.

As well as simulations of acoustic instruments, the ADRS envelope has the ability to crate original sounds which do not correspond to any "real" instruments. A good example would be the "backwards" envelope. This envelope has a slow Attack and rapid decay which sounds very much like an instrument that has been recorded on tape then played backwards. This envelope is shown below:

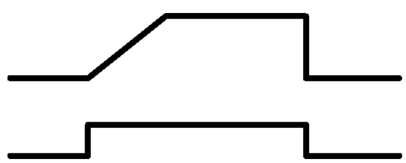


Figure 4-6 Backwards envelope.

Many unique sounds can be created by applying the amplitude envelope of one instrument to the harmonic structure of another. This produce s sounds similar to familiar acoustic instruments, yet notable different.

Soundgin has 8 envelope generators. Six that modify the amplitudes of the six oscillators and 2 that modify the amplitudes of Mixers A and B.

Four registers are used to control each of the envelope generators.

The Envelope Control register contains bits that track the current status of the envelope generator. These bits are under internal software control and it is not necessary to manipulate them directly.

The Attack Register Bits 0-3 select 1 of 16 Attack rates for the envelope generator. The Attack rate determines how rapidly the amplitude rises from zero to the Attack amplitude when the envelope generator is started. The 16 Attack rates are listed in Table 2. Bits 4-7 specify the Attack amplitude. The Attack amplitude levels range from zero to peak amplitude in 16 linear steps, with a value of 0 selecting zero amplitude and a value of 15 selecting an amplitude of 120.

The Decay Register Bits 0-3 select 1 of 16 Decay rates for the envelope generator. The Decay phase follows the Attack phase and the Decay rate determines how rapidly the output falls from the Attack amplitude to the specified Decay level. The 16 Decay rates are listed in Table 2. The oscillator will Sustain the level achieved by the Decay phase until the envelope is released. The Decay/Sustain levels are identical to the Attack levels.

The Release Register Bits 0-3 select 1 of 16 Release rates for the envelope generator. The Release phase follows the Decay phase. The output of the oscillator will remain at the selected Decay amplitude as long as the envelope is not released. When the envelope is released, the amplitude of the oscillator will fall from the Decay/Sustain amplitude to the Release amplitude at the selected Release rate. The 16 Release rates and levels are identical to the Decay rates and levels.

There are 4 different Serial Commands that control the Envelopes; Play, Play Note, Play Frequency and Release. They first three all start the envelope at the Attack phase. Two of which also load the oscillator with a frequency value. The last command, Release, sets the envelope into the Release Phase.

NOTE: The cycling of the envelope generator can be altered at any point via the Start/Release commands. The envelope generator can be started and released without restriction. For example, if the envelope is released before the envelope has finished the Attack cycle, the Release cycles will immediately begin, starting from whatever amplitude had been reached. If the envelope is then started again (before the Release cycle has reached zero amplitude), another Attack cycle will begin, starting from whatever amplitude had been reached.

Envelope Rate Table:

Duration	Attack ms	Decay/Release ms
0	2	6
1	8	24
2	16	48
3	24	72
4	38	114
5	56	168
6	68	204
7	80	240
8	100	300
9	250	750
10	500	1500
11	800	2400
12	1000	3000
13	2800	8400
14	5600	16800
15	11200	33600

Soundgin Musical Note Frequency Table

Musical Note		Frequency	FreqReg
0	C0	16.352	17,557
1	C0#	17.324	18,601
2	D0	18.354	19,708
3	D0#	19.445	20,879
4	E0	20.602	22,121
5	F0	21.827	23,436
6	F0#	23.125	24,830
7	G0	24.500	26,306
8	G0#	25.957	27,871
9	A0	27.500	29,528
10	A0#	29.135	31,284
11	B0	30.868	33,144
16	C1	32.703	35,115
17	C1#	34.648	37,203
18	D1	36.708	39,415
19	D1#	38.891	41,759
20	E1	41.203	44,242
21	F1	43.654	46,873
22	F1#	46.249	49,660
23	G1	48.999	52,613
24	G1#	51.913	55,741
25	A1	55.000	59,056
26	A1#	58.270	62,567
27	B1	61.735	66,288
32	C2	65.406	70,230
33	C2#	69.296	74,406
34	D2	73.416	78,830
35	D2#	77.782	83,517
36	E2	82.407	88,484
37	F2	87.307	93,745
38	F2#	92.499	99,320
39	G2	97.999	105,225
40	G2#	103.826	111,483
41	A2	110.000	118,112
42	A2#	116.541	125,135
43	B2	123.471	132,576
48	C3	130.813	140,459
49	C3#	138.591	148,811
50	D3	146.832	157,660
51	D3#	155.563	167,035
52	E3	164.814	176,967
53	F3	174.614	187,490
54	F3#	184.997	198,639
55	G3	195.998	210,451
56	G3#	207.652	222,965
57	A3	220.000	236,223
58	A3#	233.082	250,270
59	B3	246.942	265,152
64	C4	261.625	280,919
65	C4#	277.183	297,623
66	D4	293.665	315,321
67	D4#	311.127	334,070
68	E4	329.627	353,934
69	F4	349.228	374,980
70	F4#	369.994	397,279
71	G4	391.995	420,901
72	G4#	415.305	445,931
73	A4	440.000	472,446
74	A4#	466.164	500,540
75	B4	493.883	530,303
80	C5	523.251	561,836
81	C5#	554.365	595,244
82	D5	587.329	630,641
83	D5#	622.254	668,140
84	E5	659.255	707,870
85	F5	698.456	749,962
86	F5#	739.989	794,557
87	G5	783.991	841,803
88	G5#	830.609	891,860
89	A5	880.000	944,893
90	A5#	932.328	1,001,079
91	B5	987.767	1,060,606
96	C6	1,046.502	1,123,673
97	C6#	1,108.730	1,190,489
98	D6	1,174.659	1,261,280
99	D6#	1,244.508	1,336,280
100	E6	1,318.510	1,415,739
101	F6	1,396.913	1,499,924
102	F6#	1,479.977	1,589,113
103	G6	1,567.982	1,683,607
104	G6#	1,661.219	1,783,721
105	A6	1,760.000	1,889,785
106	A6#	1,864.655	2,002,158
107	B6	1,975.533	2,121,213
112	C7	2,093.004	2,247,346
113	C7#	2,217.460	2,380,979
114	D7	2,349.317	2,522,560
115	D7#	2,489.015	2,672,560
116	E7	2,637.020	2,831,478
117	F7	2,793.825	2,999,847
118	F7#	2,959.955	3,178,227
119	G7	3,135.963	3,367,215
120	G7#	3,322.437	3,567,440
121	A7	3,520.000	3,779,571
122	A7#	3,729.310	4,004,316
123	B7	3,951.067	4,242,426

Soundgin Allophones

Code	Allophone	Duration	Sample	Code	Allophone	Duration	Sample
192	**a**	70	Hat, Fast, Fan	224	**ne**	70	Nip, Danger, Thin
193	**aa**	70	Father, Fall	225	**no**	70	No, Snow, On
194	**ae**	165	Gate, Ate, Ray	226	**nge**	70	Think, Ping
195	**air**	200	Hair, Stair, Repair	227	**ngo**	70	Hung, Song
196	**au**	70	Hot, Clock, Fox	228	**oe**	170	Go, Hello, Snow
197	**be**	45	Bear, Bird, Beed	229	**oi**	225	Boy, Toy, Voice
198	**bo**	45	Bone, Book Brown	230	**oo**	70	Book, Could, Should
199	**eb**	40	Cab, Crib, Web	231	**ou**	200	Our, Ouch, Owl
200	**ob**	40	Bob, Sub, Tub	232	**or**	185	Corn, Four, Your
201	**ch**	70	Church, Feature, March	233	**pe**	70	People, Computer
202	**de**	45	Deep, Date, Divide	234	**po**	70	Pow, Copy
203	**do**	45	Do, Dust, Dog	235	**r**	70	Ray, Brain, Over
204	**ed**	40	Could, Bird	236	**se**	70	See, Vest, Plus
205	**od**	40	Bud, Food	237	**so**	70	So, Sweat
206	**e**	70	Met, Check, Red	238	**sh**	50	Ship, Fiction, Leash
207	**ee**	70	See, Even, Feed	239	**t**	45	Part, Little, Sit
208	**er**	199	Fir, Bird, Burn	240	**th**	40	Thin, month
209	**f**	70	Food, Effort, Off	241	**thh**	70	There, That, This
210	**ge**	55	Get, Gate, Guest,	242	**ts**	110	Parts, Costs, Robots
211	**go**	55	Got, Glue, Goo	243	**u**	70	Luck, Jump, Plus
212	**he**	70	Help, Hand, Hair	244	**ue**	70	Food, June
213	**ho**	70	Hoe, Hot, Hug	245	**v**	70	Vest, Even, Twelve
214	**i**	70	Sit, Fix, Pin	246	**w**	70	Wool, Sweat
215	**ie**	190	Mice, Fight, White	247	**y**	20	Yes, Yarn, Million
216	**j**	70	Dodge, Jet, Savage	248	**z**	70	Zoo, Zap
217	**ke**	55	Can't, Clown, Key	249	**zh**	70	Azure, Treasure
218	**ko**	55	Comb, Quick, Fox	250	**PA0**	12	
219	**ek**	45	Speak, Task	251	**PA1**	48	
220	**ok**	45	Book, Took, October	252	**PA2**	62	
221	**le**	70	Lake, Alarm, Lapel	253	**FD0**	1	
222	**lo**	70	Clock, Plus, Hello	254	**FD1**	100	
223	**m**	70	Milk, Famous, Broom	255	**FD2**	600	

Soundgin Pin Details

Pin	Name	Description
1	NC	No Connection
2	NC	No Connection
3	NC	No Connection
4	NC	No Connection
5	GND	Ground
6Q		Q Status Line
7	CTS	Clear to Send
8	RTS	Request to Send
9	TX	Transmit Serial Data
10	RX	Receive Serial Data
11	Reset	Chip Reset
12	Baud	Baud Select 9600/2400
13	Demo	Demo Select On/Off
14	+5V	5 Volt Power
15	Osc 2	Crystal Pin 2
16	Osc 1	Crystal Pin 1
17	Snd	Audio Output is > 0 Volume
18	Audio	PWM Audio Output

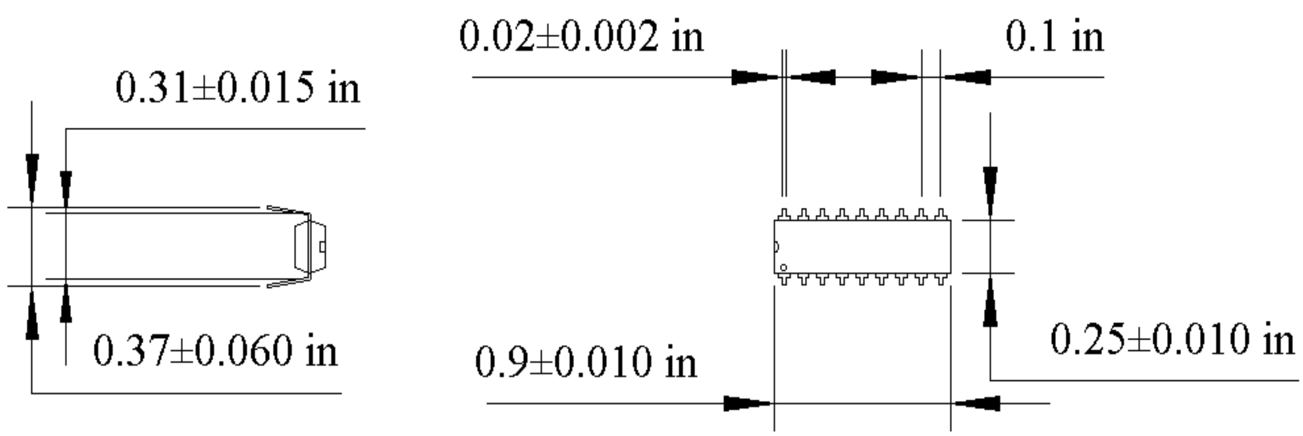
Electrical Specification:

Supply voltage	2.0 to 5.5 VDC
Supply Current:	<5ma. Plus loads
Sink/Source Current:	25ma.

Mechanical Specification:

Thermal storage:	-60 to +140 Degrees
Thermal operating	-18 to +60 Degrees

The thermal specifications are preliminary may change as testing is completed.



18 Pin Plastic DIP Package Mechanical Specifications

Reset: Master Reset

The Master Reset provides a way to reset the Soundgin to power up conditions. This forces the internal control circuit to reset and clear the input buffer. This is a logical input that requires a High or Low logic level and is active low. For normal operation, reset must be connected to V+ through a resistor.

Demo: Demo Mode Select

The Demo Select input is read by the internal control system and when connected to Ground, will cause the Soundgin to sequentially play all the Allophones. This pin is a logical inputs that requires a High or Low logic level. For normal operation, leave disconnected, or connect Demo to V+ through a resistor.

Baud: Baud Rate Select

The Baud Rate Select input is read by the internal control system and when connected to Ground, will cause the Soundgin's serial port to communicate at 2400 baud . This pin is a logical inputs that requires a High or Low logic level. For 9600 baud operation, leave disconnected, or connect Demo to V+ through a resistor.

RX and TX: Serial Receive and Transmit

RX is the Serial Input and TX is the Serial Output. These are logical lines that requires High and Low voltage level.

WARNING: These are **not** RS-232 signal level voltages.
See section on RS-232 for more details.

CTS: Buffer Half Full

The Buffer Half Full / CTS output is used for flow control with a serial port by connecting to the CTS line back to the computer or controller. This line is a logical output with either a   High or Low logic level.

RTS:

The RTS input is reserved for future use.

Sounding: Audio Output > 0 Volume

The Sounding output is used to indicate when the Soundgin is making a sound. This line is a logical output with either a High or Low logic level and is Active High.

Audio Out:

Audio Output modulates the Soundgin's output sound on a square wave carrier of 160khz. The duty cycle of this carrier is varied by the modulation of the sound output. This duty cycles typically varies by 70% and can vary up to 100% depending on the level of output represented by the audio wave. A simple "two pole" low pass filter (see Example Schematics) is all that is required before an amplifier to obtain quality sound.

Q:

The Q output is a user controllable output. This line is a logical output with either a   High or Low logic level.

Timing Charts

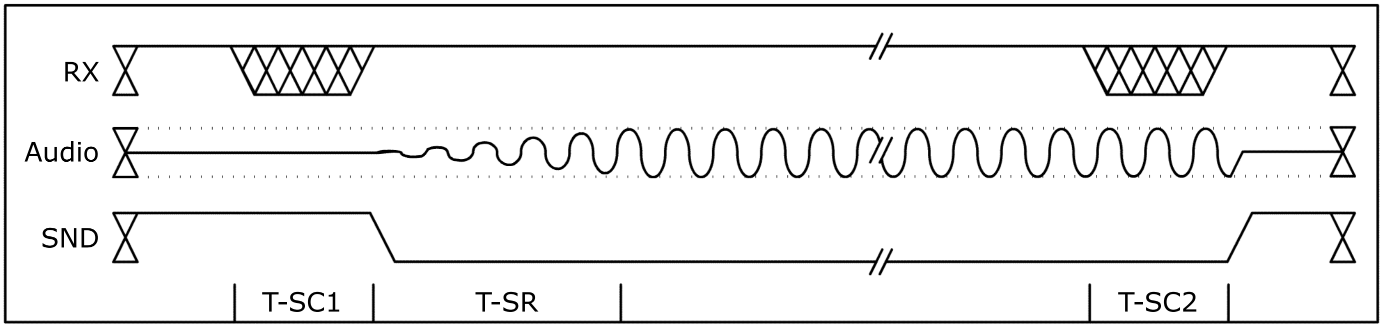


Figure 6-1: Turning Tone On and Off

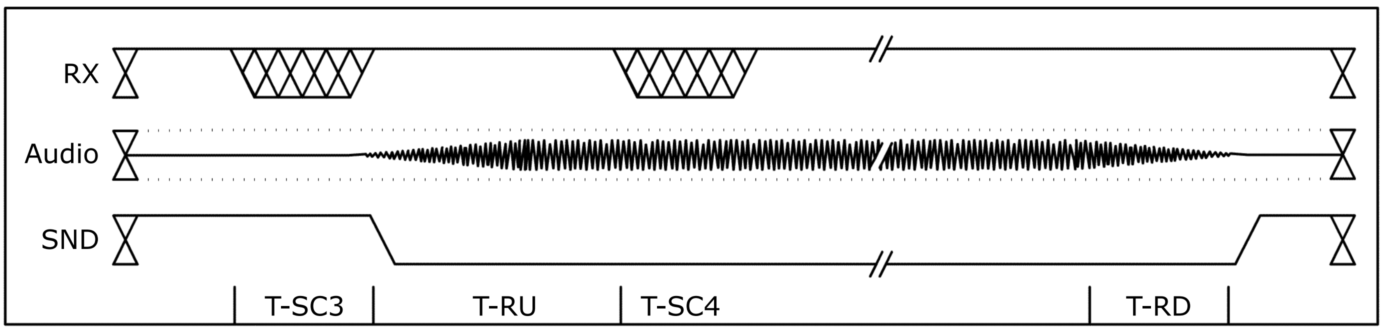


Figure 6-2: Allophone Speech

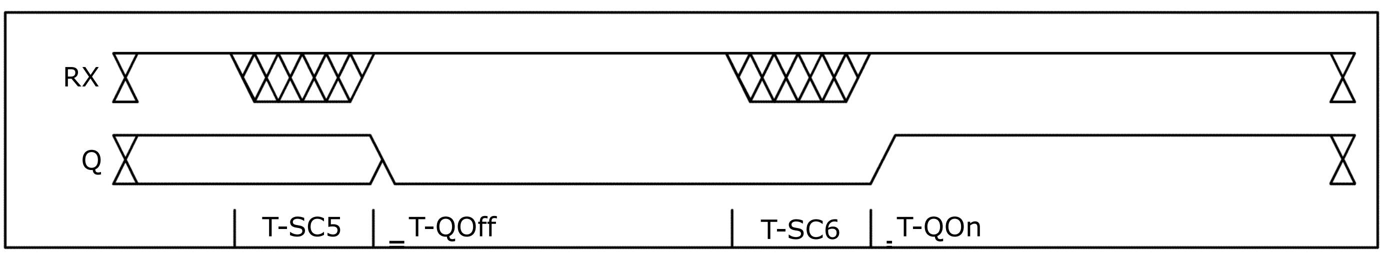


Figure 6-3: Turning Q On and Off

- T-SC1 Serial command to ramp 500Hz sound from 0 amplitude to 127 amplitude in 10ms
- T-SR Sound ramping to full volume in 10ms
- T-SC2 Serial command to set 0 amplitude
- T-SC3 Serial command to play Allophones
- T-SC4 Serial command to play PA2 after Allophones have completed
- T-RU Sound ramping up to play Allophones
- T-RD Sound ramping down from playing Allophones
- T-SC5 Serial command to turn Q line off.
- T-SC6 Serial command to turn Q line on.
- T-QOff Q line turning off.
- T-QOn Q line turning on.

Timing Charts (cont.)

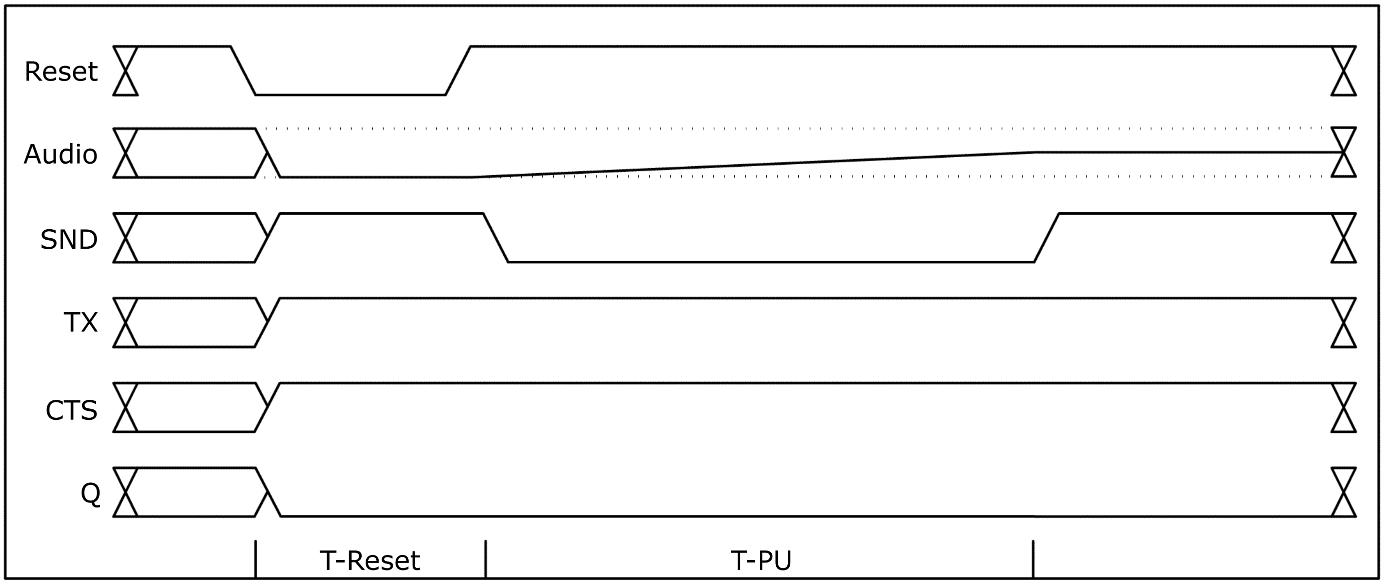


Figure 6-4: *Reset*

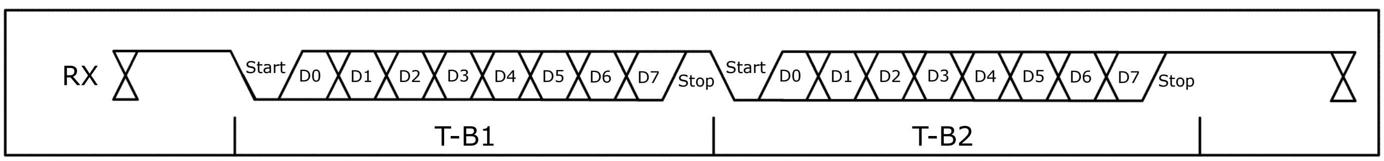


Figure 6-5: *Serial Data Reception*

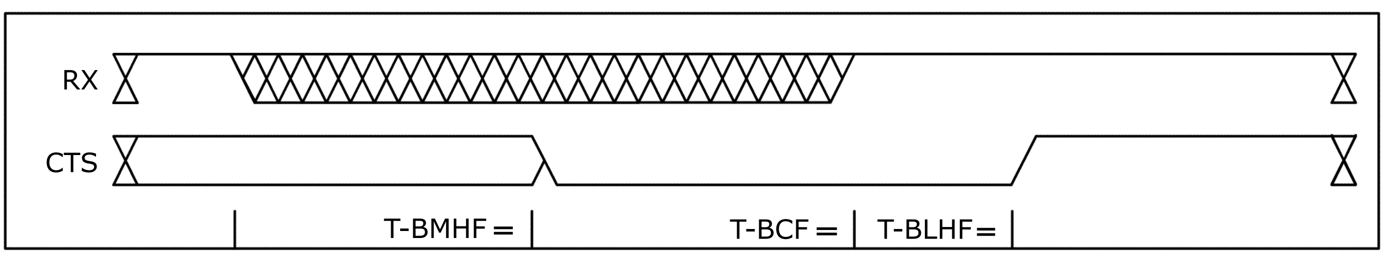
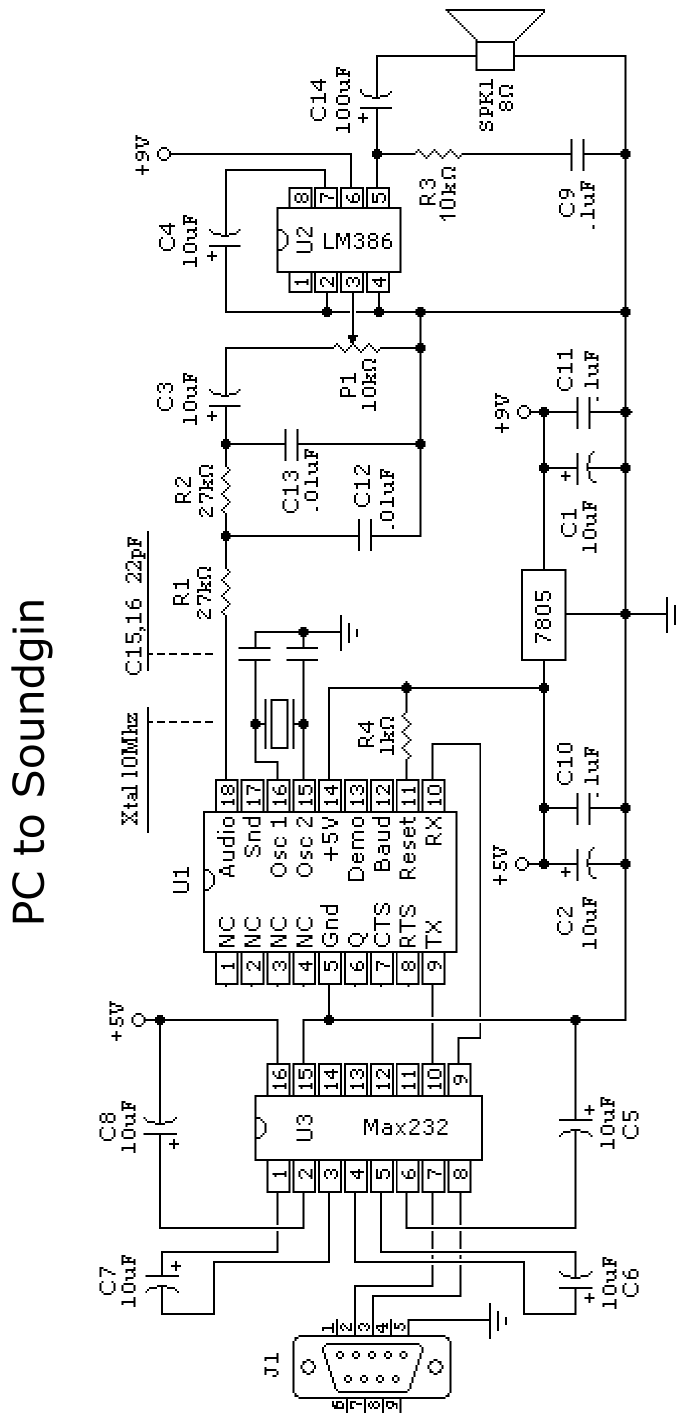


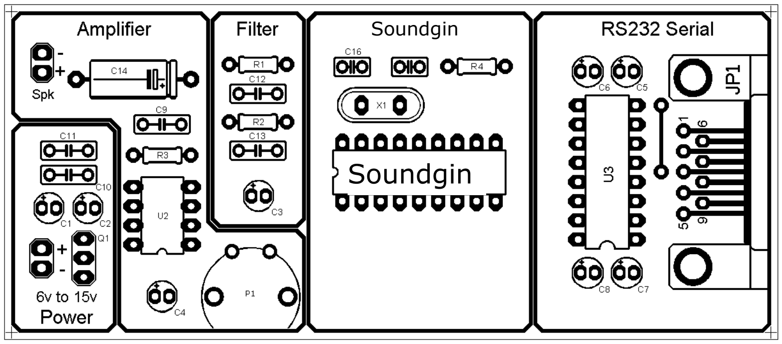
Figure 6-6: *Buffer Full*

- T-Reset Time the Reset line is held low
- T-PU 600ms ramp from Audio at 0-Volts to Audio at 2.5-Volts
- T-B1 Byte #1
- T-B2 Byte #2
- T-BMHF Buffer More than Half Full (9-Bytes)
- T-BCF Buffer completly full (16-Bytes)
- T-BLHF Buffer Less than Half Full (8-Bytes)

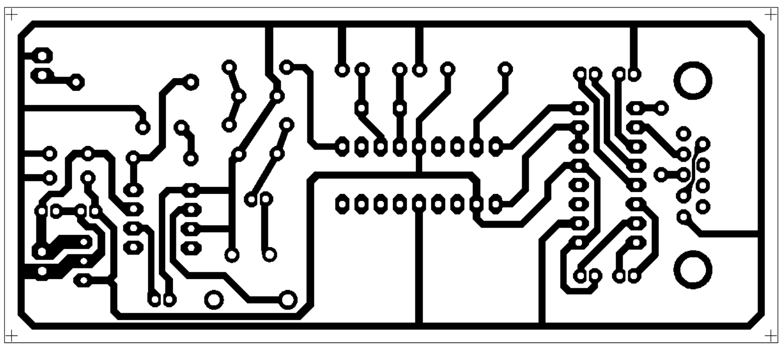
Circuit for controlling the Soundgin with an RS232 connection



Circuit Board layout for Soundgin with an RS232 connection



Parts Layout

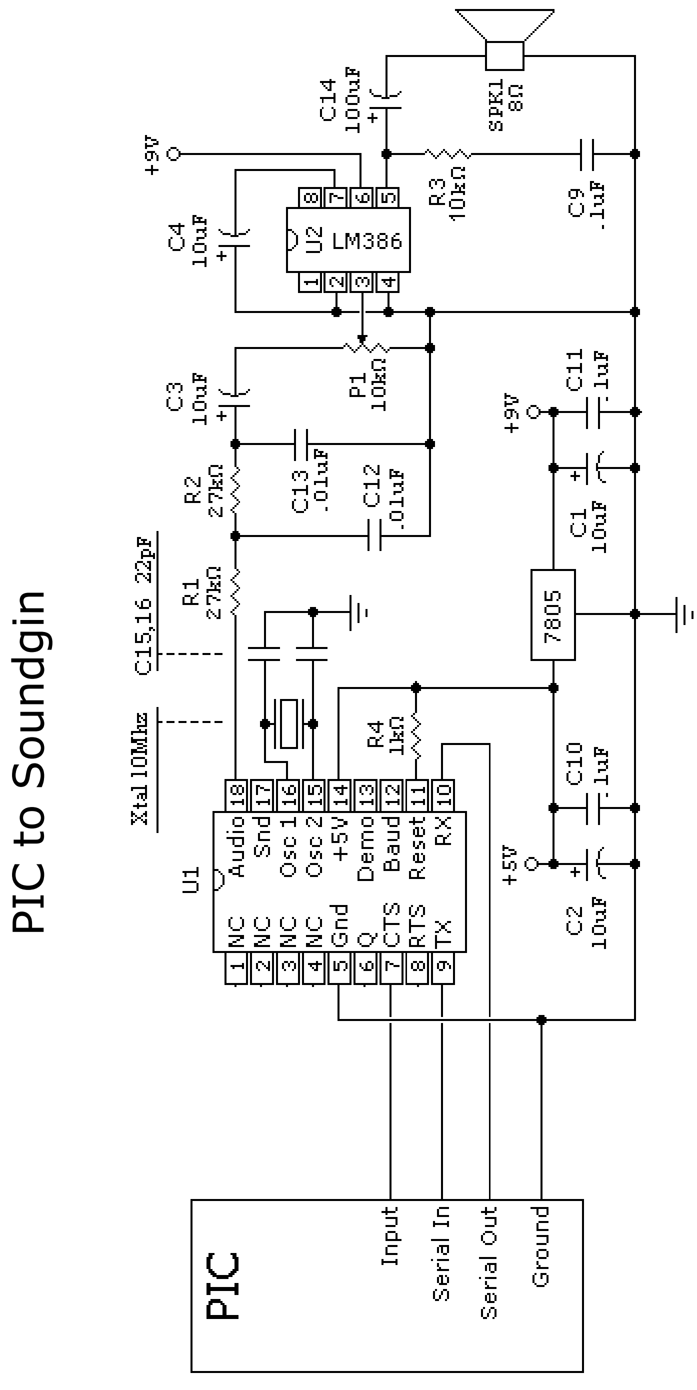


Board Traces.

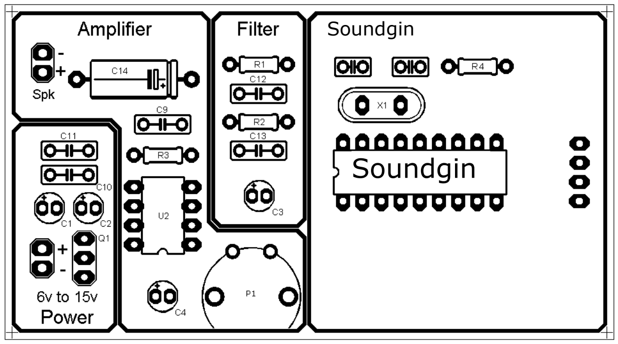
Note that the board is shown from the parts side and that the traces are seen loooking through the Circuit Board.

Device	Part	Quantity	Value	Description
Capacitors	C1-C8	8	10uf 16v	Radial Lead Electrolytic > 6v
	C9-C11	3	0.1uf	Radial Lead Ceramic > 6v
	C12, C13	2	0.01uf	Radial Lead Ceramic > 6v
	C14	1	100uf	Axial Lead Electrolytic > 6v
	C15, C16	2	22pf	Radial Lead Ceramic > 6v
Resistors	R1,R2,R4	2	27k Ohms	Radial Lead 10%-5% 1/8th-1/4 Watt
	R3,R5	1	10 Ohms	Radial Lead 10%-5% 1/8th-1/4 Watt
Potentiometers	P1	1	10k Ohms	EVLHFAA06 With dial knob.
Connectors	JP6	1	9-Pin D	Female Right Angle
ICs	U1	1	BBIC	Soundgin
	U2	1	LM386N-3	Amplifier
	U3	1	MAX202	RS232 Converter
Crystals	X1	1	10Mhz	
Regulators	Q1	1	7805	1-Amp

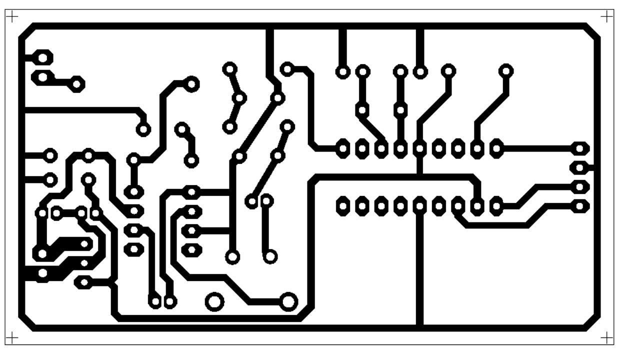
Circuit for controlling the Soundgin with a PIC



Circuit Board layout for controlling the Soundgin with a PIC



Parts Layout



Board Traces.

Note that the board is shown from the parts side and that the traces are seen looking through the Circuit Board.

Device	Part	Quantity	Value	Description
Capacitors	C1-C4	4	10uf 16v	Radial Lead Electrolytic > 6v
	C9-C11	3	0.1uf	Radial Lead Ceramic > 6v
	C12, C13	2	0.01uf	Radial Lead Ceramic > 6v
	C14	1	100uf	Axial Lead Electrolytic > 6v
	C15, C16	2	22pf	Radial Lead Ceramic > 6v
Resistors	R1,R2,R4	2	27k Ohms	Radial Lead 10%-5% 1/8th-1/4 Watt
	R3,R5	1	10 Ohms	Radial Lead 10%-5% 1/8th-1/4 Watt
Potentiometers	P1	1	10k Ohms	EVLHFAA06 With dial knob.
Connectors	JP6	1	9-Pin D	Female Right Angle
ICs	U1	1	BBIC	Soundgin
	U2	1	LM386N-3	Amplifier
Crystals	X1	1	10Mhz	
Regulators	Q1	1	7805	1-Amp