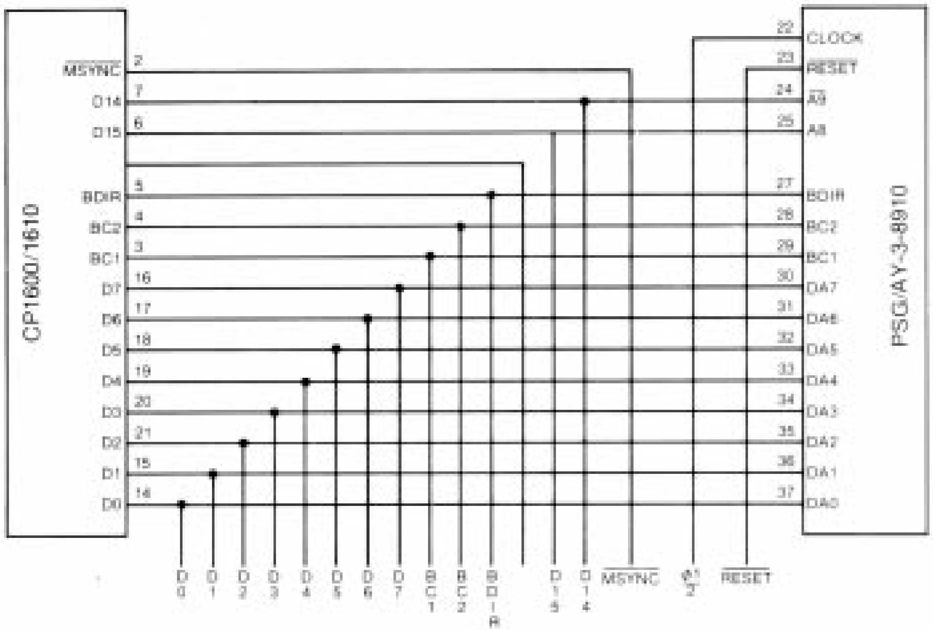
Fig. 20 CP1600/1610/AY-3-8910 INTERFACE



4.8 Interfacing to the M6800

An M6800 microprocessor can be interfaced with an AY-3-8910/8912 through the addition of an M6820 PIA chip. The I/O ports designated as PA0 to PA7 are used as the 8 bit bus lines and I/O ports PB0 to PB2 are used as the bus control lines. The software routines shown are used to control the latch address, write data, and read data functions for the AY-3-8910/8912.

4.8.1 LATCH ADDRESS ROUTINE

```
;AT ENTRY, B HAS ADDRESS VALUE
;
LATCH CLRA
   STAA 8005 ;GET D DIR A
   LDAA #FF
   STAA 8004 ;OUTPUTS
   LDAA #4
   STAA 8005 ;GET PERIPHERAL A
   STAB 8004 ;FORM ADDR
   STAA 8006
   CLRA
   STAA 8006 ;LATCH ADDRESS
   RTS ;RETURN
```

4.8.2 WRITE DATA ROUTINE

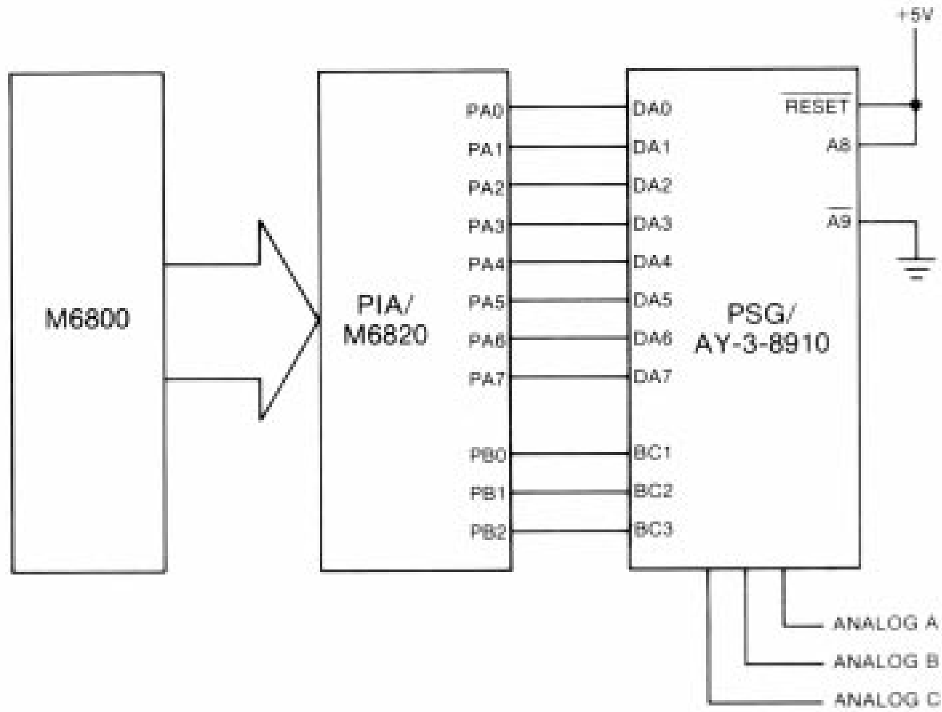
```
;AT ENTRY, B HAD DATA VALUE
;
WRITE STAB 8004 ;FORM DATA
   LDAA #6 ;DWS
   STAA 8006
   CLRA
   STAA 8006 ;WRITE DATA
   RTS ;RETURN

   !
```

4.8.3 READ DATA ROUTINE

```
;AFTER READ, B HAS READ DATA
;
READ STA A 8005 ;GET D DIR
   STA A 8004 ;INPUTS
   LDAA #4
   STA A 8005 ;GET PERIPHERAL
   DECA
   STA A 8006 ;READ MODE
   LDA B 8004 ;READ DATA
   CLRA
   STA A 8006 ;REMOVE READ MODE
   RTS        ;RETURN
```

Fig. 21 M6800/AY-3-8910 INTERFACE



4.9 Interfacing to the 8080 S100 Bus

The sample S100 bus design provides for reading and writing the PSG using only an 8080 "IN" or "OUT" instruction to the proper address. Another feature of the design is the provision for multiple PSG devices to be connected to a single bus. The system described is presently running two PSG's, one to each of two stereo channels.

As can be seen from the read and write routines in the illustrative program, the program overhead necessary to communicate with the PSG is minimal.

4.9.1 LATCH ADDRESS ROUTINE

```
PORTADDR EQU 80H ;ADDRESS TRANSFER PORT ADDRESS
PORTDATA EQU 81H ;DATA TRANSFER PORT ADDRESS
;
;THIS ROUTINE WILL TRANSFER THE CONTENTS OF
;8080 REGISTER C TO THE PSG ADDRESS REGISTER
PSGBAR      MOV    A,C ;GET C IN A FOR OUT
            OUT    PORTBAR ;SEND TO ADDRESS PORT
            RET
```

4.9.2 WRITE DATA ROUTINE

```
;
;
;ROUTINE TO WRITE THE CONTENTS OF 8080 REGISTER B
;TO THE PSG REGISTER SPECIFIED BY 8080 REGISTER C
;
PSGWRITE    CALL   PSGBAR ;GET ADDRESS LATCHED
            MOV    A,B, ;GET VALUE IN A FOR TRANSFER
            OUT    PORTDATA ;PUT TO PSG REGISTER
            RET
```

4.9.3 READ DATA ROUTINE

```
;
;
;ROUTINE TO READ THE PSG REGISTER SPECIFIED
;BY THE 8080 REGISTER C AND RETURN THE DATA
;IN 8080 REGISTER B
;
PSGREAD     CALL   PSGBAR
            IN     PORTDATA ;GET REGISTER DATA
            MOV    B,A GET IN TRANSFER REGISTER
            RET
```

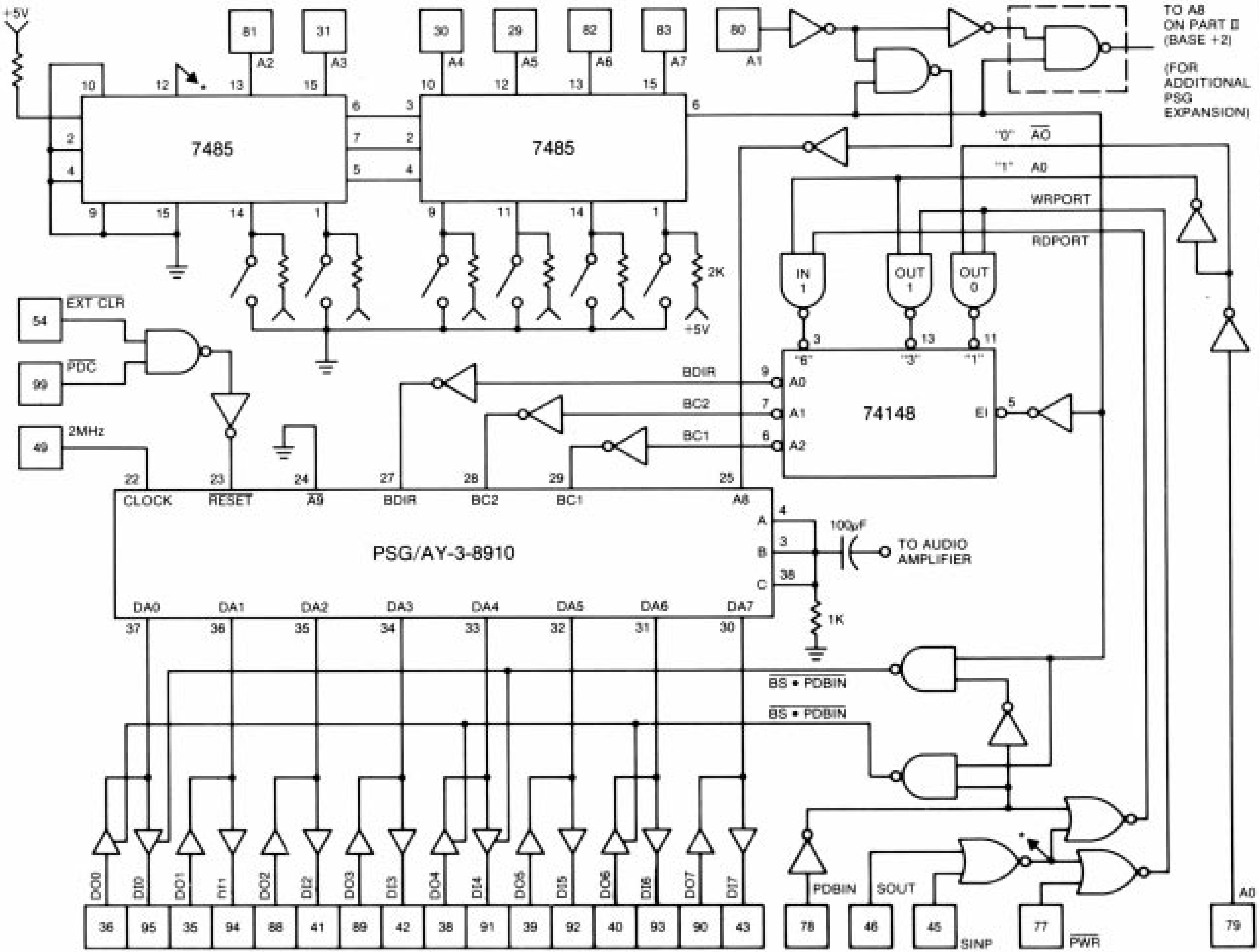


Fig. 22 8080 S100 BUS/AY-3-8910 INTERFACE

5 MUSIC GENERATION

The production of music involves the creation of series of frequencies which are pleasing to the human ear (setting critical evaluation aside). This involves essentially mathematical relationships, making the application ideal for digital devices. For example, the shifting up or down in octaves is a multiplication or division by a power of 2, which is a simple shift operation for most microprocessors.

Another factor in music generation is "communication". The composer must be able to convey his tune ideas so that a musician or group of musicians can reproduce the composer's ideas—often on widely differing instruments. This concept involves "tuning" the instruments to a standard set of frequencies and following a set rhythm pattern. The tuning frequency most widely used is based on the third octave note "A" of 440Hz, the "Equal Tempered Chromatic Scale".

Although it is easy to construct recognizable tunes using only one note at a time, the simultaneous sounding of more than one note to produce chords and counterpoint vastly increases the quality of the sound. This feature is easily achieved in the PSG since three channels are provided, each independently programmable.

5.1 Note Generation

Since notes are formed by sustaining a particular frequency for a preset period of time at a varying amplitude, the PSG performs this function with a series of simple register loads. The method used in many cases is to obtain register load values for first octave notes and to shift to the correct octave at playtime.

The chart in Fig. 23 lists a full 8 octaves of notes from a low of C1 (32.703Hz) to a high of B8 (7902.080Hz). Assuming an input clock frequency of 1.78977MHz (one half the standard "color" crystal frequency of 3.579545MHz), and applying the formulas of Section 3.1 for calculating Tone Period register load values, results in the register values shown. The nature of the PSG divider scheme produces a high degree of accuracy for low frequencies, less for high frequencies. This can be seen in the chart in the comparison of "ideal frequencies" and "actual frequencies", with the ideal frequencies being those of the Equal Tempered Chromatic Scale, and the actual frequencies being the "best fit" values from the formula calculation.

NOTE	OCTAVE	IDEAL FREQUENCY	ACTUAL FREQUENCY	12-BIT REGISTER VALUE IN OCTAL
C	1	32.703	32.698	6 5 3 5
C#	1	34.648	34.653	6 2 3 4
D	1	36.708	36.712	5 7 4 7
D#	1	38.891	38.895	5 4 7 4
E	1	41.203	41.201	5 2 3 3
F	1	43.654	43.662	5 0 0 2
F#	1	46.249	46.243	4 5 6 3
G	1	48.999	48.997	4 3 5 3
G#	1	51.913	51.908	4 1 5 3
A	1	55.000	54.995	3 7 6 2
A#	1	58.270	58.261	3 6 0 0
B	1	61.735	61.733	3 4 2 4
C	2	65.406	65.416	3 2 5 6
C#	2	69.296	69.307	3 1 1 6
D	2	73.416	73.399	2 7 6 4
D#	2	77.782	77.789	2 6 3 6
E	2	82.406	82.432	2 5 1 5
F	2	87.308	87.323	2 4 0 1
F#	2	92.498	92.523	2 2 7 1
G	2	97.998	98.037	2 1 6 5
G#	2	103.826	103.863	2 0 6 5
A	2	110.000	109.991	1 7 7 1
A#	2	116.540	116.522	1 7 0 0
B	2	123.470	123.467	1 6 1 2
C	3	130.812	130.831	1 5 2 7
C#	3	138.592	138.613	1 4 4 7
D	3	146.832	146.799	1 3 7 2
D#	3	155.564	155.578	1 3 1 7
E	3	164.812	164.743	1 2 4 7
F	3	174.616	174.510	1 2 0 1
F#	3	184.996	184.894	1 1 3 5
G	3	195.996	195.903	1 0 7 3
G#	3	207.652	207.534	1 0 3 3
A	3	220.000	220.198	0 7 7 4
A#	3	233.080	233.043	0 7 4 0
B	3	246.940	246.933	0 7 0 5
C	4	261.624	261.357	0 6 5 4
C#	4	277.184	276.883	0 6 2 4
D	4	293.664	293.598	0 5 7 5
D#	4	311.128	310.724	0 5 5 0
E	4	329.624	329.973	0 5 2 3
F	4	349.232	349.565	0 5 0 0
F#	4	369.992	370.400	0 4 5 6
G	4	391.992	392.494	0 4 3 5
G#	4	415.304	415.839	0 4 1 5
A	4	440.000	440.397	0 3 7 6
A#	4	466.160	466.087	0 3 6 0
B	4	493.880	494.959	0 3 4 2
C	5	523.248	522.714	0 3 2 6
C#	5	554.368	553.766	0 3 1 2
D	5	587.328	588.741	0 2 7 6
D#	5	622.256	621.449	0 2 6 4
E	5	659.248	658.005	0 2 5 2
F	5	698.464	699.130	0 2 4 0
F#	5	739.984	740.800	0 2 2 7
G	5	783.984	782.243	0 2 1 7
G#	5	830.608	828.598	0 2 0 7
A	5	880.000	880.794	0 1 7 7
A#	5	932.320	932.173	0 1 7 0
B	5	987.760	989.918	0 1 6 1
C	6	1046.496	1045.428	0 1 5 3
C#	6	1108.736	1107.532	0 1 4 5
D	6	1174.656	1177.482	0 1 3 7
D#	6	1244.512	1242.898	0 1 3 2
E	6	1318.496	1316.009	0 1 2 5
F	6	1396.928	1398.260	0 1 2 0
F#	6	1479.968	1471.852	0 1 1 4
G	6	1567.968	1575.504	0 1 0 7
G#	6	1661.216	1669.564	0 1 0 3
A	6	1760.000	1747.825	0 1 0 0
A#	6	1864.640	1864.346	0 0 7 4
B	6	1975.520	1962.470	0 0 7 1
C	7	2092.992	2110.581	0 0 6 5
C#	7	2217.472	2237.216	0 0 6 2
D	7	2349.312	2330.433	0 0 6 0
D#	7	2489.024	2485.795	0 0 5 5
E	7	2636.992	2663.352	0 0 5 2
F	7	2793.856	2796.520	0 0 5 0
F#	7	2959.936	2943.705	0 0 4 6
G	7	3135.936	3107.244	0 0 4 4
G#	7	3322.432	3290.023	0 0 4 2
A	7	3520.000	3495.649	0 0 4 0
A#	7	3729.280	3728.693	0 0 3 6
B	7	3951.040	3995.028	0 0 3 4
C	8	4185.984	4142.992	0 0 3 3
C#	8	4434.944	4474.431	0 0 3 1
D	8	4698.624	4660.866	0 0 3 0
D#	8	4978.048	5084.581	0 0 2 6
E	8	5273.984	5326.704	0 0 2 5
F	8	5587.712	5593.039	0 0 2 4
F#	8	5919.872	5887.410	0 0 2 3
G	8	6271.872	6214.488	0 0 2 2
G#	8	6644.864	6580.046	0 0 2 1
A	8	7040.000	6991.299	0 0 2 0
A#	8	7458.560	7457.385	0 0 1 7
B	8	7902.080	7990.056	0 0 1 6

Fig. 23 EQUAL TEMPERED CHROMATIC SCALE (f_{CLOCK}=1.78977MHz)

5.2 Tune Entry/ Playback

One of the methods of entering a composition into a computer memory would be to utilize a keyboard to pass number and alphabetic information concerning the composer's wishes. An alternate method would be to scan a positional series of switches (like a piano keyboard) to determine note, volume and duration data.

Since flexibility in tune entry is desired, it is important to allow the composer to specify certain constants of entry such as octave, pitch or tempo, and have these entries normalized to a known value.

5.3 Tune Variations

One of the significant features of a microcomputer based music player is the ability to modify the tune once it has been recorded. Among the simpler variations are:

5.3.1 OCTAVE SHIFT

If an octave constant is added to the octave of the recorded note prior to storing the value in the PSG register, dynamic pitch changes can be obtained. The programming effect would be to shift one bit left for each lower octave and one bit right for each higher octave. For example, the effect will be that a tune written to play on a piano will sound like bells if a multiple octave up modification is performed.

5.3.2 KEY

One measure of the virtuosity of a musician is his ability to modify the "key" or suboctave shift of a composition. The logical description of key transposition is to shift each note up or down by a predetermined number of notes from the original. For example, a piece written in C and played in C# would have all C notes shifted to C#, C# shifted to D, etc. (Note that the case must be considered where B of one octave is shifted to C of the next higher octave.) All of these operations require that the one of twelve note identification must be retained in the recorded representation.

5.3.3 TEMPO

The duration of each recorded note is best expressed in terms of "ticks" of an overall "tempo clock". At playtime, the total duration can be obtained by programatically multiplying the individual note to "slow down" or "speed up" the tune without changing the crucial time relationship between the notes. This can be accomplished by imbedding the note timing loops within the tempo timing loops for simple operation.

5.3.4 CHORDS

There are certain combinations of notes which when played simultaneously produce pleasant combinations. These "chords" can be easily formed from a base note by performing octave and key changes on two notes, which are played with the main note. These relationships are illustrated in Fig. 24, which lists the various note constants which will produce musical chords. A chord with a particular quality may be formed by playing its root, a 3rd Minor or Major, and other notes from the chord chart. For example, a C Major chord is formed from C(÷2), E(÷2), and G(÷2).

Fig. 24 **CHORD SELECTION CHART**

Chord Selection	Root	3rd Minor	3rd Major	4th	5th	6th	7th
C	C (÷2)	D# (÷2)	E (÷2)	F (÷2)	G (÷2)	A (÷2)	A# (÷2)
C#	C# (÷2)	E (÷2)	F (÷2)	F# (÷2)	G# (÷2)	A# (÷2)	B (÷2)
D	D (÷2)	F (÷2)	F# (÷2)	G (÷2)	A (÷2)	B (÷2)	C (÷1)
D#	D# (÷2)	F# (÷2)	G (÷2)	G# (÷2)	A# (÷2)	C (÷1)	C# (÷1)
E	E (÷2)	G (÷2)	G# (÷2)	A (÷2)	B (÷2)	C# (÷1)	D (÷1)
F	F (÷2)	G# (÷2)	A (÷2)	A# (÷2)	C (÷1)	D (÷1)	D# (÷1)
F#	F# (÷4)	A (÷4)	A# (÷4)	B (÷4)	C# (÷2)	D# (÷2)	E (÷2)
G	G (÷4)	A# (÷4)	B (÷4)	C (÷2)	D (÷2)	E (÷2)	F (÷2)
G#	G# (÷4)	B (÷4)	C (÷2)	C# (÷2)	D# (÷2)	F (÷2)	F# (÷2)
A	A (÷4)	C (÷2)	C# (÷2)	D (÷2)	E (÷2)	F# (÷2)	G (÷2)
A#	A# (÷4)	C# (÷2)	D (÷2)	D# (÷2)	F (÷2)	G (÷2)	G# (÷2)
B	B (÷4)	D (÷2)	D# (÷2)	E (÷2)	F# (÷2)	G# (÷2)	A (÷2)

5.4 Sound Variation

5.4.1 RELATIVE CHANNEL VOLUME

The independently programmable amplitude control for each channel allows up to 16 levels if using the processor controlled amplitude mode (bit 4 of registers 10, 11 or 12=0). In the case of a decaying or steady note, when a note is played or "fired", a frequency may be set up in the coarse and fine tune registers and then an amplitude value placed in the respective register 10, 11 or 12. The value which is placed to play the tune can be an independent variable, allowing channels to play their respective melody lines with varying force.

5.4.2 DECAY

The main difference between a "piano" sound and an "organ" sound is the speed with which the note loses volume. If all of the notes can be decayed at a uniform rate, the automatic envelope generator can be set to produce a decaying waveform. Each of the three channels can have the same decay constant but differing playing times to simulate the same instrument with differing note-strike times.

5.4.3 OTHER EFFECTS

The addition of variable noise to any or all of the channels can produce modification effects such "breathing" with a wind instrument. Or noise can be used alone to produce a drum rhythm. The fact that the noise dominant frequencies are variable allows "synthesizer" type effects with simple processor interaction.

Other pleasing effects include vibrato and tremolo, the cyclical variation of the frequency and volume. Because an intelligent microprocessor is controlling the effect, they can be all keyed to the tune itself or to other external stimuli.