Song Reader											
Cycle	**Play**	**State**	**note_done**	**advance**	**next_count**	**count**	**song_done**	**next_address**	**current address**	**ROM output**	**new_note**
0	1	Pause	0	0	0	0	0	0,0,00000	XXXXXX	XXXXXX	0
1	1	retreiving_note	0	0	0	0	0	0,0,00000	0,0,000000	XXXXXX	0
2	1	load new note	0	0	0	0	0	0,0,00000	0,0, 00000	{1'd 0, 6,d49, 6'd12, 3'd0}	1
3	1	waiting	0	0	0	0	0	0,0,00000	0,0, 00000	{1'd 0, 6,d49, 6'd12, 3'd0}	0
4	1	waiting	1	0	0	0	0	0,0,00000	0,0, 00000	{1'd 0, 6,d49, 6'd12, 3'd0}	0
5	1	increment_address	0	0	0	0	0	0,0,00001	0,0,00000	{1'd 1, 6,d49, 6'd12, 3'd0}	0
6	1	retreiving_note	0	0	0	0	0	0,0,00001	0,0,00001	{1'd 0, 6,d49, 6'd12, 3'd0}	0
7	1	load new note	0	0	2	0	0	0,0,00001	0,0,00001	{1'd 1, 6,d0, 6'd2, 3'd0}	0
8	1	waiting	0	1	1	2	0	0,0,00001	0,0,00001	{1'd 1, 6,d0, 6'd2, 3'd0}	0
9	1	waiting	0	1	0	1	0	0,0,00001	0,0,00001	{1'd 1, 6,d0, 6'd2, 3'd0}	0
10	1	waiting	1	0	2	0	0	0,0,00001	0,0,00001	{1'd 1, 6,d0, 6'd2, 3'd0}	0
	1	increment_address	0	0	0	0	0	0,0,00002	0,0,00001	{1'd 1, 6,d0, 6'd2, 3'd0}	0
	1										
		*assume we are playing first note of first song *assume the note just takes 1 cycle to finish playing									

Note Player

Cycle	play_enable	advance	Load_new_note	Duration_to_load	Next_count	Count	done_with_note
0	1	0	1	6'd4	4	0	0
1	1	0	0	6'd4	4	0	0
2	1	1	0	6'd4	3	4	0
3	1	1	0	6'd4	2	3	0
4	1	1	0	6'd4	1	2	0
5	1	1	0	6'd4	0	1	0
6	1	0	0	6'd4	0	1	0
7	1	1	0	6'd4	4	0	1
8	1	0	0	6'd4	4	0	0
9	1	0	1	6'd2	2	0	0
10	1	1	0	6'd2	1	2	0
11	1	1	0	6'd2	0	1	0
12	1	1	0	6'd2	2	0	1
13	1	0	0	6'd2	2	0	0
	1	0	1	6'd2	2	0	0

*first 13 cycles show the duration counter of the note_player module

**load_new_note flips 2 cycles after done_with_note because it takes 1 cycle to increment the address and 1 cycle to load the new note

***when we turn the enable off, all the values in the counter should stay the same and stop decrementing

Cycle	State	Load_new_note	Note to load	Next_note to load	Freq_rom out	sample_out	Sample ready	Generate_next	Notes
0	Load note	1	6'd1	0	XXXX	0	0	1	Load a new note
1	Load note	0	6'd1	6'd1	XXXX	0	0	0	Note is latched
2	Freq_ROM	0	6'd1	6'd1	{10'd009, 10'd395}	0	0	0	Step size is outputted
3	Sine_Reader	0	6'd1	6'd1	10'd009, 10'd395}	0	0	0	
4	Sine_ReaderOut	0	6'd1	6'd1	10'd009, 10'd395}	1	16'd00050	0	Sine reader generates a sample
5	Codec	0	6'd1	6'd1	10'd009, 10'd395}	0	16'd00050	1	Codec signals for a new sample
6	Sine_ROM	0	6'd1	6'd1	10'd009, 10'd395}	0	16'd00050	0	
7	Sine_ReaderOut	0	6'd1	6'd1	10'd009, 10'd395}	1	16'd1870	0	Add step size to get new sample
8	Load note	1	6'd6	6'd1	10'd009, 10'd395}	0	16'd1870	0	
9	Load note	0	6'd6	6'd6	10'd009, 10'd395}	0	16'd1870	0	Notice that sample does not
10	Freq_ROM	0	6'd6	6'd6	{10'd012, 10'd557}	0	16'd1870	0	
11	Sine_ROM	0	6'd6	6'd6	{10'd012, 10'd557}	0	16'd1870	0	
12	Sine_ReaderOut	0	6'd6	6'd6	{10'd012, 10'd557}	1	16'd24713	0	
13	Codec	0	6'd6	6'd6	{10'd012, 10'd557}	0	16'd24713	1	

Harmonic Chord Player

Clk	Reset	Play_enable	GNS	State	activate	Load_new_note	note	duration	State1	Load_new_note1	note1	duration1	note_done1	sample_ready1	count1	State2	Load_new_note2	note2	duration2	note_done2	sample_ready2	count2	State3	Load_new_note3	note3	duration3	note_done3	sample_ready3	count3	note_done	sample_ready
0	0	0	0	Loading	0	0	0	0	initial	0	0	0	0	0	0	initial	0	0	0	0	0	0	initial	0	0	0	0	0	0	1	0
1	0	1	0	Loading	0	1	6'd36	6'd4	np - Load note	1	6'd36	6'd4	1	0	4	waiting	0	0	0	1	0	3	waiting	0	0	0	1	0	0	0	0
2	0	1	0	loading	0	1	6'd32	6'd3	np - Load note	0	6'd36	6'd4	0	0	4	np - Load Note	1	6'd32	6'd3	0	0	3	waiting	0	0	0	1	0	0	0	0
3	0	1	0	loading	0	1	6'd30	6'd2	np - Freq_ROM	0	6'd36	6'd4	0	0	4	np - Load Note	0	6'd32	6'd3	0	0	3	np - Load note	1	6'd30	6'd2	0	0	2	0	0
4	0	1	0	loading	0	0	0	0	waiting	0	6'd36	6'd4	0	0	4	np - Freq_ROM	0	6'd32	6'd3	0	0	3	np - Load note	0	6'd30	6'd2	0	0	2	0	0
5	0	1	0	loading	0	0	0	0	waiting	0	6'd36	6'd4	0	0	4	waiting	0	6'd32	6'd3	0	0	3	np - Freq_ROM	0	6'd30	6'd2	0	0	2	0	0
6	0	1	0	np	1	0	0	0	np - Sine_Reader	0	6'd36	6'd4	0	0	4	np - Sine_Reader	0	6'd32	6'd3	0	0	3	np - Sine_Reader	0	6'd30	6'd2	0	0	2	0	0
7	0	1	0	np	1	0	0	0	np - Sine_ReaderOut	0	6'd36	6'd4	0	0	4	np - Sine_ReaderOut	0	6'd32	6'd3	0	0	3	np - Sine_ReaderOut	0	6'd30	6'd2	0	0	2	0	0
8	0	1	0	ch	1	0	0	0	ch - Sine_Reader	0	6'd36	6'd4	0	0	4	ch - Sine_Reader	0	6'd32	6'd3	0	0	3	ch - Sine_Reader	0	6'd30	6'd2	0	0	2	0	0
9	0	1	0	ch	1	0	0	0	ch - Sine_ReaderOut	0	6'd36	6'd4	0	1	4	ch - Sine_ReaderOut	0	6'd32	6'd3	0	1	3	ch - Sine_ReaderOut	0	6'd30	6'd2	0	1	2	0	1
10	0	1	1		1	0	0	0	repeat process	0	6'd36	6'd4	0	0	3	repeat process	0	6'd32	6'd3	0	0	2	repeat process	0	6'd30	6'd2	0	0	1	0	0
11	0	1	0		1	0	0	0		0	6'd36	6'd4	0	0	3		0	6'd32	6'd3	0	0	2		0	6'd30	6'd2	0	0	1	0	0
12	0	1	0		1	0	0	0		0	6'd36	6'd4	0	0	3		0	6'd32	6'd3	0	0	2		0	6'd30	6'd2	0	0	1	0	0
13	0	1	0		1	0	0	0		0	6'd36	6'd4	0	0	3		0	6'd32	6'd3	0	0	2		0	6'd30	6'd2	0	0	1	0	0
14	0	1	0		1	0	0	0		0	6'd36	6'd4	0	0	3		0	6'd32	6'd3	0	0	2		0	6'd30	6'd2	0	0	1	0	0
15	0	1	0		1	0	0	0		0	6'd36	6'd4	0	0	3		0	6'd32	6'd3	0	0	2		0	6'd30	6'd2	0	0	1	0	0

GNS = generate_next_sample
np = note_player
ch = create_harmonic

Create Harmonic							
Clk	**Reset**	**weight**	**play_enable**	**generate_next_sample**	**step_size**	**samp_ready2/3**	**harmonic_out**
0	0	2	1	0	20'd200	0	-
1	0	2	1	1	20'd200	0	-
2	0	2	1	0	20'd200	0	-
3	0	2	1	0	20'd200	1	harm1 + harm2 + harm3
4	0	1	1	1	20'd200	0	-
5	0	1	1	0	20'd200	0	-
6	0	1	1	0	20'd200	1	harm1 + harm2
7	0	0	1	1	20'd200	0	-
8	0	0	1	0	20'd200	0	-
9	0	0	1	0	20'd200	1	harm1
10	0	0	1	1	20'd200	0	-
*harm1 is the input sample_in bit shifted, and harm2 and harm3 are generated by sine_reader							
*sine_reader outputs on the third cycle							

Interactive Instrument Editing								
Cycle	**Reset**	**Switch1**	**Next_Weight**	**Weight**	**Up_button**	**Down_button**		
0	0	0	0	0	0	0	switch off, no changes	
1	0	0	0	0	1	0	switch off, no changes	
2	0	0	0	0	0	1	switch off, no changes	
3	0	1	1	0	1	0		
4	0	1	1	1	0	0		
5	0	1	1	1	0	0		
6	0	1	2	1	1	0		
7	0	1	1	2	0	1		
8	0	1	0	1	0	1		
9	0	1	0	0	0	1		
10	0	0	0	0	0	0	switch off, no changes	

SINE_ READER								
Cycle	**Play**	**State**	**Generate_next**	**next_addr**	**current addr**	**Sine_ROM output**	**sample_ready**	**sample**
0	1	Signal to generate next sample	1	0	XXXXXX	XXXXXX	0	0
1	1	Next_addr pushed to curr_addr	0	0	0	XXXXXX	0	0
2	1	Curr_addr passed into sine_ROM	0	0	0	16'd00000	1	16'd0000
3	1	sine_ROM output	1	0	0	16'd00000	0	0
4	1	Process repeats						