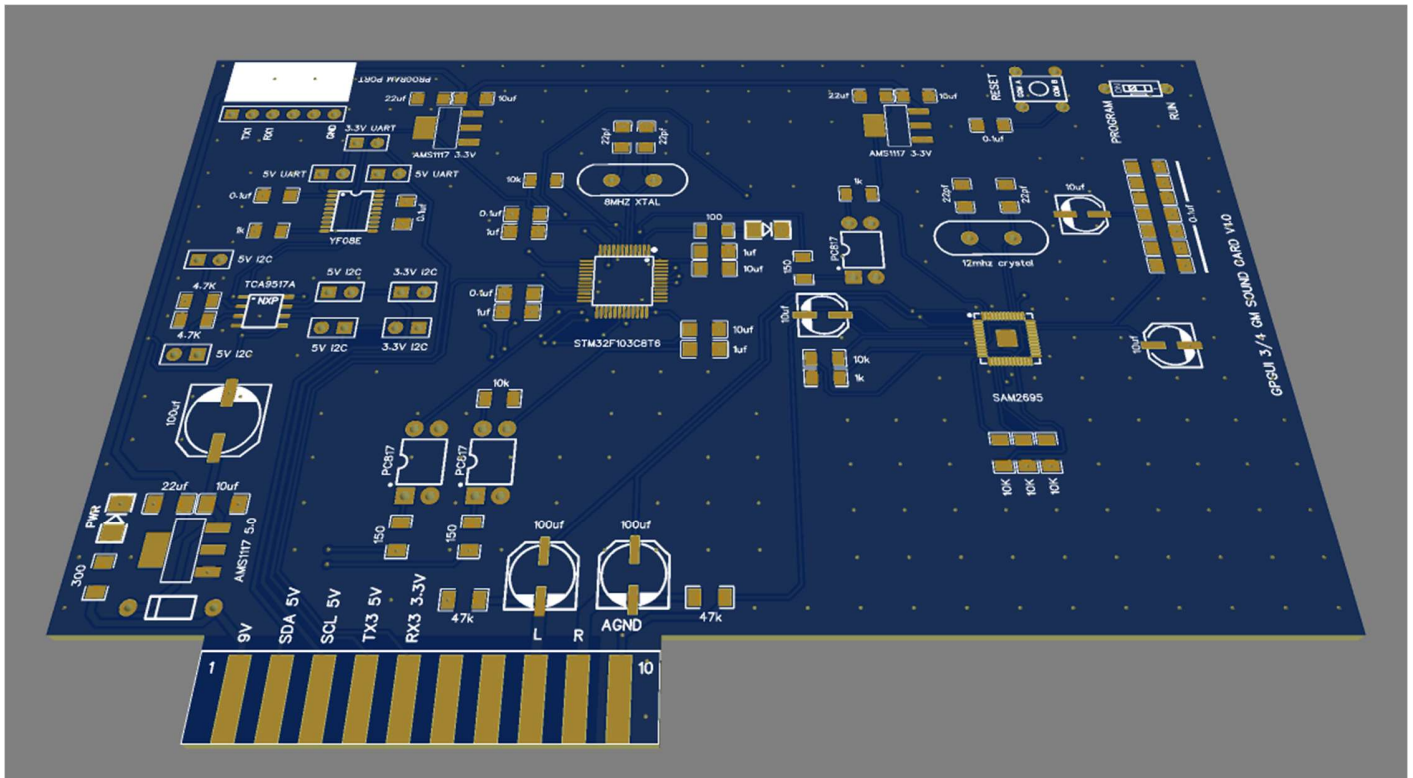


# GPGUI 3 SYSTEM SOUNDS



- System sounds are produced using either the general MIDI sound card, or the pc buzzer through system A CPU.
- Midi channel 16 is reserved for system sounds.
- All CC messages on channel 16 read from the midi in port will be ignored.

## Contents

Values saved in memory for the GPGUI Midi Soundcard.....	2
Values saved in memory for Buzzer.....	2
System Sounds Midi Instruments.....	3
Functions that play sounds .....	5

## Values saved in memory for the GPGUI Midi Soundcard.

System Sounds Variables • system_Sound_Settings.h	EEPROM Address	Data type	Value
system_Sounds_To_Midi_Or_Buzzer	0	bool	0 - 1
system_Sounds_Enable	1	bool	0 - 1
system_Sounds_Window_Open_Close_Tone_Enable	2	bool	0 - 1
system_Sounds_Buttons_Tone_Enable	3	bool	0 - 1
system_Sounds_Slider_Tone_Enable	4	bool	0 - 1
system_Sounds_Instrument	5	char	0 - 19
system_Sounds_Volume	6	char	0 - 127
system_Sounds_Velocity	7	char	0 - 127
system_Sounds_Reverb_Level	8	char	0 - 127
system_Sounds_Chorus_Flanger_Level	9	char	0 - 127
system_Sounds_Modulation_Level	10	char	0 - 127
system_Sounds_Button_Enter_Note	11	char	0 - 127
system_Sounds_Button_Back_Note	12	char	0 - 127
system_Sounds_Button_On_Note	13	char	0 - 127
system_Sounds_Button_Off_Note	14	char	0 - 127
system_Sounds_Slider_On_Note	15	char	0 - 127
system_Sounds_Slider_Off_Note	16	char	0 - 127
system_Sounds_Slider_Up_Note	17	char	0 - 127
system_Sounds_Slider_Down_Note	18	char	0 - 127
system_Sounds_Slider_Click_To_Value_Note	19	char	0 - 127
system_Sounds_Window_Open_Note	20	char	0 - 127
system_Sounds_Window_Close_Note	21	char	0 - 127
system_Sounds_System_Start_Up_Tune	22	char	0 - 127
system_Sounds_System_Sound_Font	23	char	0 - 127

## Values saved in memory for Buzzer.

System Sounds Buzzer Variables • system_Sound_Settings.h	EEPROM Address		Data type	Value
	MSB	LSB		
button_Enter_Pitch	24	25	int	0 - 5000
button_Back_Pitch	26	27	int	0 - 5000
button_On_Pitch	28	29	int	0 - 5000
button_Off_Pitch	30	31	int	0 - 5000
slider_On_Pitch	32	33	int	0 - 5000
slider_Off_Pitch	34	35	int	0 - 5000
slider_Click_To_Value_Pitch	36	37	int	0 - 5000
window_Open_Pitch	38	39	int	0 - 5000
window_Close_Pitch	40	41	int	0 - 5000
system_Start_Up_Pitch	42	43	int	0 - 5000
tone_duration	44	45	char	0 - 255

## Saving and reading soundcard settings.

This function sets the system sound settings to a preset sound font during program run time.

Parameters:	Description
<ul style="list-style-type: none"><li>• <code>char</code> sound_Font</li></ul>	Sound font data structure reference number.

```
char sound_Font = 1;  
set_System_Sounds(sound_Font); //writes sound font data structure to ram globals in settings file
```

- This function writes the current system sound settings to memory during run time.

```
write_System_Sound_Settings_To_Memory(); //writes ram globals to eeprom
```

**(NOTE: NOT TO BE CALLED IN A LOOP!)**

## System Sounds Midi Instruments.

System Sounds Midi Instrument Name	Value 0 - 19	Patch 0 - 127
Clavinet	0	7
Celesta	1	8
Glockenspiel	2	9
Music Box	3	10
Vibraphone	4	11
Marimba	5	12
Xylophone	6	13
Tubular Bells	7	14
Orchestral Harp	8	46
Timpani	9	47
Banjo	10	105
Shamisen	11	106
Koto	12	107
Kalimba	13	108
Tinkle Bell	14	112
Agogo	15	113
Steel Drums	16	114
Woodblock	17	115
Nylon Acoustic Guitar	18	24
Electric Piano 1	19	4

## Functions that play sounds.

**Note:** these functions use either the buzzer or the Midi card if enabled( Or both can be enabled).

- `play_Button_Enter_Tune();`
- `play_Button_Back_Tune();`
- `play_Button_On_Tune();`
- `play_Button_Off_Tune();`
- `play_Slide_On_Tune();`
- `play_Slide_Off_Tune();`
- `play_Slide_Up_Tune();`
- `play_Slide__Down();`
- `play_Slide_Click_To_Value_Tune();`
- `play_Window_Open_Tune();`
- `play_Window_Close_Tune();`
- `play_Power_Up_Tune();`

How to initialize the soundcard.

How to route the midi in port to the soundcard.