Electrical Overview

Year: 2014 **Semester:** Fall **Team:** 10 **Project:** Digital Effects Processor
Creation Date: September 9, 2014 **Last Modified:** October 7, 2015
Author: <redacted> **Email:** rdacted@purdue.edu

Assignment Evaluation:

Item	Score (0-5)	Weight	Points	Notes
Assignment-Specific Items				
Electrical Overview				
Electrical Considerations				
Interface Considerations				
System Block Diagram				
Writing-Specific Items				
Spelling and Grammar				
Formatting and Citations				
Figures and Graphs				
Technical Writing Style				
Total Score				

5: Excellent 4: Good 3: Acceptable 2: Poor 1: Very Poor 0: Not attempted

General Comments:

1.0 Electrical Overview

For this product, we will be utilizing a 32-bit microcontroller for the purpose of processing the audio and applying the effects for outputting to the user. This microcontroller will also be in charge of managing the peripherals of the device and the interfacing between the different components.

For the translation of the analog audio signal into a digital signal for the application of the audio effects, we will be utilizing a 16-bit audio codec. Within this codec, there is a 16-bit analog to digital converter (ADC) and a 16-bit digital to analog converter (DAC), both of which communicate with the microcontroller through I2S [1].

Other components that will be used in the device for the audio processing are a 512k words by 16-bit SRAM to store the audio data (16-bit); a display to output information to the user (most likely an LCD, however this is decision is yet to be finalized); a series of encoders and momentary switches for the user interface; and a relay to control audio signal flow in and out of the pedal.

In the power supply system, we will be using a low dropout regulator to step the voltage down from 9V to the 3.3V required for powering the microcontroller and the audio codec. The reasoning behind this choice of regulator is that when the device is being powered by battery, we can take advantage of almost the entire lifetime of the battery without interfering with or infringing on the dropout voltage. From the battery datasheet [2], we can see that the battery is still operational as it hits 5V. At this voltage, the difference of 1.7V is not large enough to operate many of the alternatives to an LDO. Nevertheless, due to the low efficiency of these devices, research will continue into alternatives such as switching regulators and drop-in replacements. Whether or not we choose to use these will be dependent on finding one that needs minimal support components while possessing a small dropout voltage (< 1.5V).

As an audio device, electrical noise will be a significant issue. Due to this, there will be substantial decoupling in the power supply. This will be in addition to bypass capacitors on the inputs and outputs of the regulator and 9V supply. This will have the effect of decreasing the noise that could interfere with sound quality and effect generation.

With regards to data collection and transmission, the ADC of the audio codec will capture and convert the 16-bit audio data and then transmit it to the microcontroller where the data is processed and effects applied. From the microcontroller, the data is stored in the SRAM for a period and number of times as dictated by the desired effect. Lastly the data is passed back to the DAC section of the audio codec where it is translated back to an analog signal ready for audio output. The microcontroller will be additionally used to implement delay algorithms that make up effects such as reverb, delay and chorus, as well as basic filtering algorithms. It will also send serial data to the display via SPI.

2.0 Electrical Considerations

For this project, we will be utilizing both battery and wall power. For both of these power systems, we will be inputting 9V and stepping it down to the voltages required for each component. The main reason behind using a 9V supply is that this is the industry standard for guitar pedal electronics, and will enable compatibility between our device and others on the market [3]. Also, due to the fact that our device will be used heavily for extended periods of time, the battery will discharge and the supply will decay over a small timeframe. Using a higher nominal voltage will allow for increased usable battery life. In addition to this, it will also make battery charge management easier due to the ability to split the input voltage for comparison.

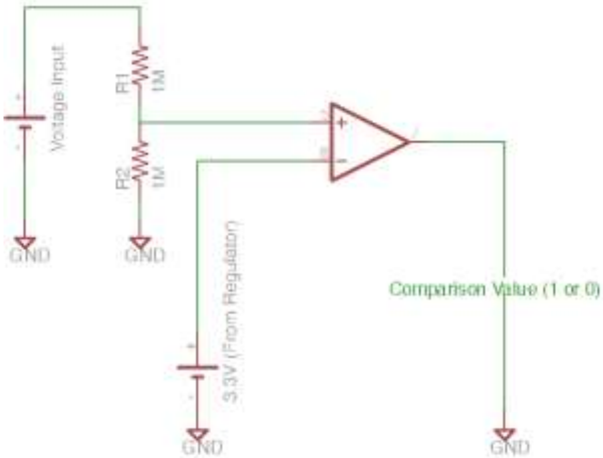

Figure 1: Battery Charge Management Pseudo Circuit

The microcontroller and the audio codec both require 3.3V for power and therefore this voltage will have to be supplied from this 9V power supply [1] [4]. Due to the faster than desired decay in battery voltage, the battery will be the secondary power system for the device, for use in testing and sound checks. The wall supply is anticipated to take on the role of powering the device for performances. By designing the power supply such that it can be used for both the battery input and wall input, we are able to keep the design simple, as well as minimizing its footprint.

Voltages between the nominal 9V and approximately 6.6V will be able to operate the audio circuitry. For this reason, we will not be regulating this rail, as regulation would reduce the operating range. Instead, we will monitor the battery state / input voltage and when it has decayed to a designed threshold we will signal battery low and power down. This will give enough battery life to properly warn the user and shut down to avoid damage to circuitry.

The operating frequency for our system is dictated by the microcontroller. As discussed above, the microcontroller that we are expecting to use has a maximum operating frequency of 48 MHz [4]. Since we will want the highest signal processing per sample possible, we will operate at this maximum frequency. Also, since the project will be drawing the majority of its power from the wall, battery performance is less of a concern and microcontroller performance can be optimized.

Due to the expected usage case of our project, the battery power supply will have to be able to power down completely. Although our project requires a high input voltage relative to the supply voltage of many of the components, the current draw is small and therefore power wasted is expected to be small.

Over voltage is not expected to be an issue due to robust components. Our input voltage needs to be above 6.6V to be able to operate the audio circuitry. Higher than normal voltages (above 9V) will not cause an issue due to the regulator on the 3.3V rail being good up to 40V [5] and the audio being healthy up to about 12V.

The current requirements for each of the devices are included in the table below:

Device	**Maximum**
Audio Codec [1]	10
Microcontroller [4]	130
SRAM [6]	50
LCD Driver [7]	10
Total	**200**

Table 1: Electrical Loading Data

With regards to loading due to external devices, the pedal interfaces with the guitar and an amp however these are self powered and therefore are not a loading concern.

3.0 Interface Considerations

Our two main interfaces will be SPI and I2S. SPI will be used for the display as it is sufficient for our needs and is simple to use. I2S will be used to transmit stereo audio to and from the codec. This is due to being a common protocol for audio codecs that offers many different bit depths (16, 24, 32) as well as different sampling rates (48kHz, 96kHz, 192kHz etc.) [1]. In addition to these, we will use be a parallel interface for memory to achieve quick read/write times.

The audio codec will require three different clocks [1]. The master clock (MCLK) will require 256*(sampling frequency) where the sampling frequency is equal to 48kHz, resulting in a master clock of 12.288MHz. The bit clock (BCLK) is equal to (channels)*(bit-depth)*(sampling rate) which is 2*16*48kHz or 1.536 MHz. The third clock, the input / output channel clock or the word select clock (LRCK) needs to be equal to the sampling rate, or 48kHz.

For the SPI, we will most likely utilize a 1MHz clock and the SRAM, while having a maximum read/write speed of 100MHz, will derived for the micro clock and we expect it to equal approximately 48MHz [6].

For the SPI interface, we hope to use no parity, single stop bit. However, this cannot be confirmed until the display is chosen and finalized. With the I2S interface, we will be using 32-bit packet sizes, with two 16-bit audio samples per packet.

4.0 Sources Cited:

[1] A. Kasei. (2006, November). *AK4556 3V 192kHz 24Bit CODEC* [Online]. Available: http://www.akm.com/akm/en/file/datasheet/AK4556VT.pdf

[2] Energizer. (2014). *Energizer LA522 Advanced Lithium* [Online]. Available: http://data.energizer.com/PDFs/la522.pdf

[3] Wampler Pedals (2014). *Classic Series: Clarksdale* [Online]. Available: http://www.wamplerpedals.com/manuals/clarksdale.pdf

[4] Atmel. (2014, July). *Atmel SAM D21 E/F/G SMART ARM-Based Microcontroller* [Online]. Available: https://www.atmel.com/Images/Atmel-42181-SAM-D21_Datasheet.pdf

[5] STMicroelectronics. (2014, February). *LFXX Very Low Drop Voltage Regulator with Inhibit Function* [Online]. Available: http://www.st.com/web/en/resource/technical/document/datasheet/CD00000546.pdf

[6] Alliance. (2012, February). *AS7C38098A 512K x 16 Bit High Speed CMOS SRAM* [Online]. Available: http://www.mouser.com/ds/2/12/as7c38098a-3863.pdf

[7] Philips. (1999, April 12). *PCD8544 48x84 Pixels Matrix LCD Controller/Driver* [Online]. Available: https://www.sparkfun.com/datasheets/LCD/Monochrome/Nokia5110.pdf

Appendix 1: System Block Diagram

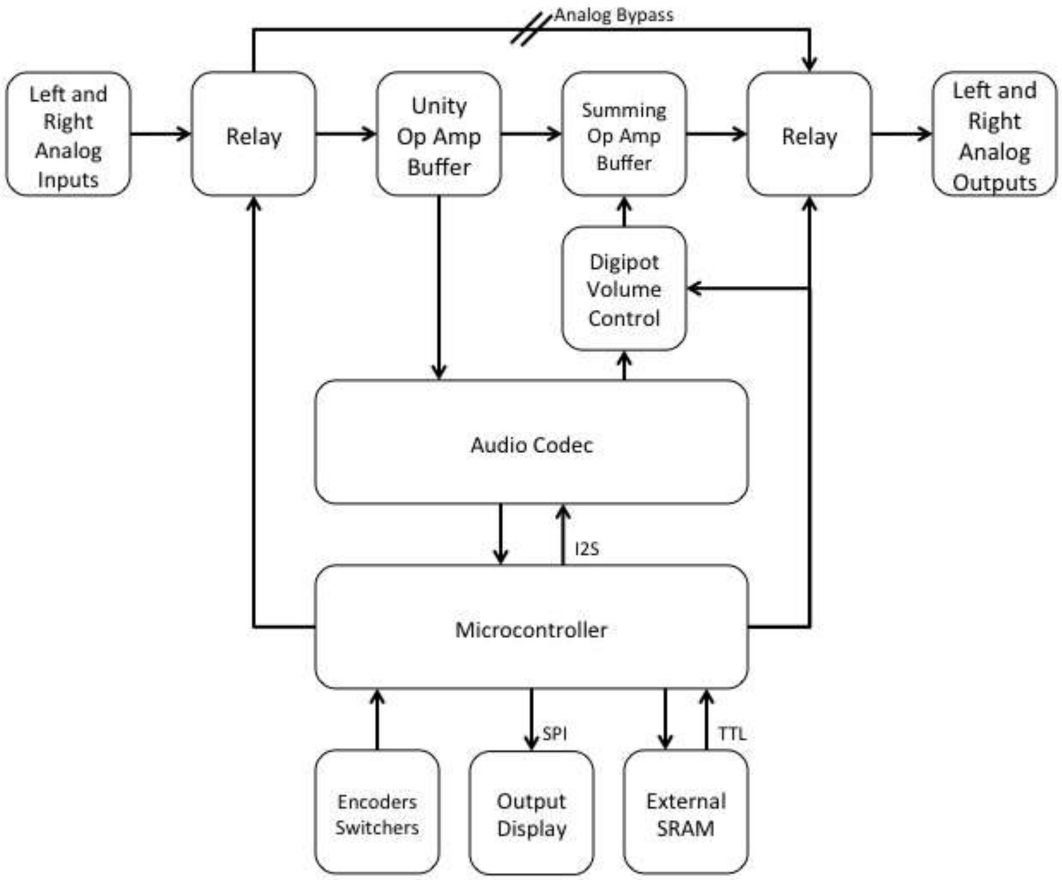

Figure 2: System Block Diagram