**Design Review 1**

**Gamblers**

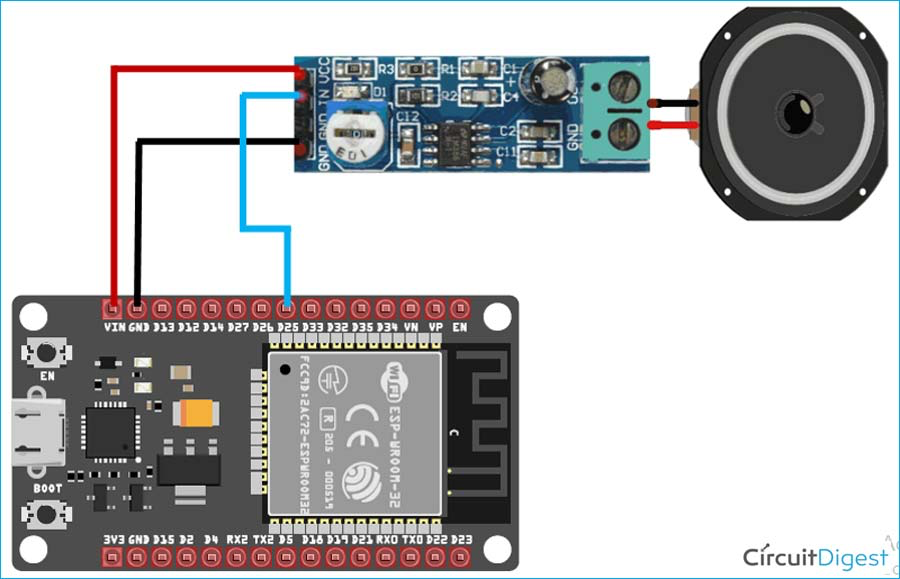
**VGA/Monitor Subsystem:**



The central component of our project is the screen that displays the desired betting information we receive from the users in our website. Using the ESP32 WROOM, we are able to connect via analog to the display and create a menu with up to a 800x600 pixel resolution and 14-bit color with a DAC converter or a resistor ladder. The monitor receives power through its own connection to an outlet, while the ESP32 WROOM receives power via our powering scheme (see Power section). The display is coded based mainly on the examples found in PlatformIO library ESP32Lib by Bitluni. One of the current concerns is the amount of pin connections the monitor requires, therefore, we might expand to use two ESP32-WROOM microcontrollers, one for information processing and the other for monitor control.

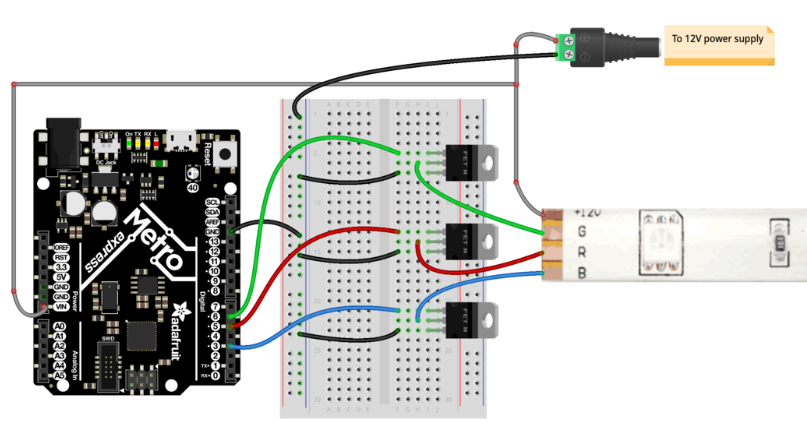
**Speakers:**

For the audio section of the project, we will be driving a tiny little serviceable speaker from pre-loaded audio files. These audio files are converted to hex code via this program: <https://mh-nexus.de/en/hxd/> and saved as a *wav* file. This can then be exported as a “C” file and read into PlatformIO. Team Gamblers is taking advantage of the XT\_DAC\_Audio\_Library in order to read these sound files. As for hardware, the ESP32 is connected to a power source and an I2S amplifier. This amplifier in turn drives the aforementioned speaker.



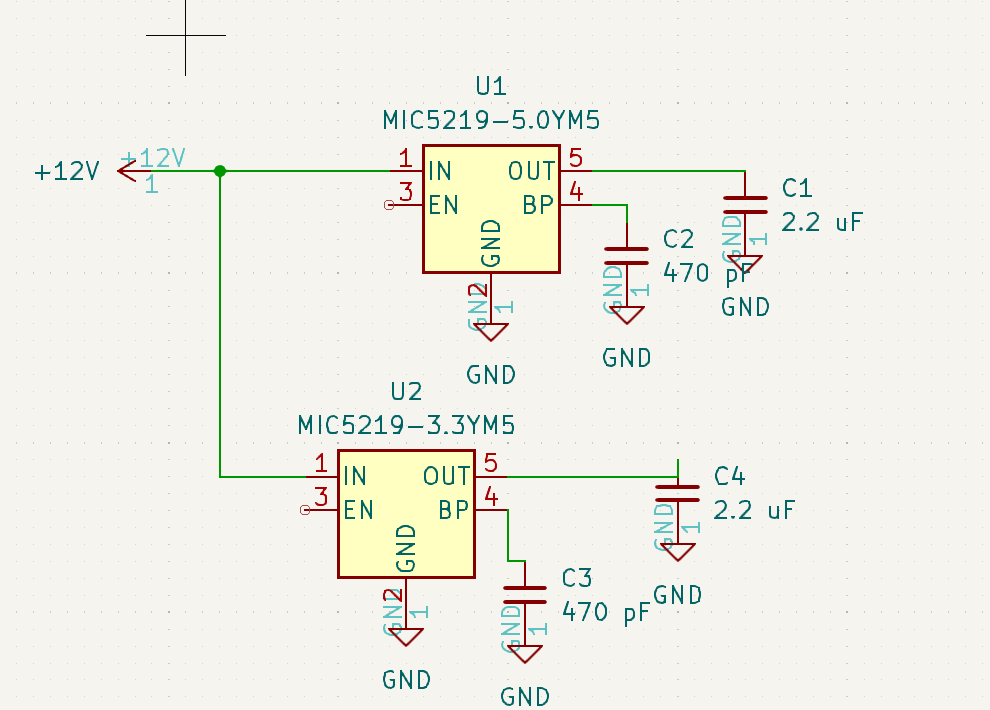
**LED Lights**

One of the subsystems is going to be LED lights that give visual feedback to the user based on the outcome of the bets. This will be driven by an RGB light strip that will utilize 2 of the GPIO pins from the microcontroller. By using these 2 GPIO pins it can easily display either Red or Green light around the screen to help indicate whether the user won or lost a leg in their bet list. The lights are said to draw 20 mA at 12 V per section of 3 lights, therefore, it will need to draw power from the power regulation circuit. Team Gamblers will only need to utilize the Arduino library to get this subsystem to work and it will be interfaced in conjunction with the speakers to create more feedback for the users. The hardware needed for this subsystem is the ESP32, the RGB lights, and also two MOSFETs that will help drive the strip.



**Power Regulation:**

The power regulation circuit is going to be used to power all subsystems except the monitor. This will be done by taking in a 12 V Wall Wart and utilizing two linear voltage regulators to step the power down from 12 V and then to a 5 V source and a 3.3 V source. The two Voltage regulators are the MIC5219-5.0 and MIC5219-3.3, however, this could be subject to change depending on the necessary power draw needed for the other subsystems.



**Sports Data API**

To acquire the live sports data, the ESP32 will make API calls through <https://developer.sportradar.com> . The calls will be made using the WifiClientSecure.h library from Arduino and be adjusted based on the bets submitted by the user. Once the call has been made, one or multiple parsing functions will glean the relevant scores or statistics from the response and store them on the microcontroller for later use.

**User Website Info**

A website will be used for the users to enter their betting information to be read by the ESP32. The website will allow users to create a screen name and enter bets from existing dropdown options. Then the website will display all user bets as the user enters them, and allows the user to delete bets. For each input bet, creates a class that has all details stored as properties. Future steps will include adding more betting options for users and improving the interface to make the website more user friendly.

