Electrical Overview

Year: 2019 Semester: Fall Team: 19 Project: Digitopoly

Creation Date: September 12, 2019 Last Modified: September 13, 2019

Author: Carson Kelley Email: kelley96@purdue.edu

Assignment Evaluation:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Item** | **Score (0-5)** | **Weight** | **Points** | **Notes** |
| **Assignment-Specific Items** |
| **Electrical Overview** |  | x3 |  |   |
| **Electrical Considerations** |  | x3 |  |   |
| **Interface Considerations** |  | x3 |  |   |
| **System Block Diagram** |  | x3 |  |   |
| **Writing-Specific Items** |
| **Spelling and Grammar** |  | x2 |  |   |
| **Formatting and Citations** |  | x1 |  |   |
| **Figures and Graphs** |  | x2 |  |   |
| **Technical Writing Style** |  | x3 |  |   |
| **Total Score** |  |   |

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

General Comments:

*Relevant overall comments about the paper will be included here*

1.0 Electrical Overview

Our design is split into 2 main components which are the main game board and the wireless dice. For the game board we will be using a 32-bit microcontroller which will act as the main controller for the game board, keeping track of the current game state. There will be a Raspberry Pi connected to it which will handle displaying the game board. The display is a standard computer monitor. The main microcontroller will take in the user input from buttons hooked up to GPIO pins, interpret those inputs, and then update the game state as required. It will then send the necessary information to the Pi to update the display accordingly. The game pieces are moved by an XY plotter which has an electromagnet that is controlled by a switch which is turned on and off by a GPIO pin. The plotter has two stepper motors which are each controlled by a stepper motor controller, which handles direction and speed, which will be indicated by the main micro using GPIO pins. In order to communicate with the dice wirelessly a Bluetooth receiver/transceiver will be connected to the main microcontroller via UART.

The wireless dice will each have a 32-bit microcontroller that has a Bluetooth receiver/transceiver built into it. This microcontroller will be used to interpret the gyroscope and accelerometer data from a separate IMU chip, sent over SPI. The microcontroller will interpret the value rolled from this data and will send it to the game board microcontroller via Bluetooth. This is the setup for both dice and they will be polled sequentially by the main micro, eg. When a user needs to roll their turn and thus roll data is expected from the dice.

2.0 Electrical Considerations

Operating Frequencies:

Power Budget:
-*Main game board*

 The power for the main game board is provided from a 120V AC wall plug. This will be used to power the monitor used as the main display. This will also be split and then converted to DC power for the rest of the components in the main board. The different DC voltages required for the main board are 12V, 9V, 5V and 3.3V. These voltages will be obtained by using regulators to step down from 12V to each individual voltage. The power will need to be constant for the STM and the Raspberry Pi 4, but this should not be an issue due to the voltages coming from regulators that will be supplied with consistent power. The components are broken down by voltages below.

Total current consumed by main board components:

230mA + 3A + 2 \* 250mA + 240mA + 33.6mA = **4.004A**

Total power consumed by main board components:

2.76W + 15W + 2.250W +792mW + 110.88mW= **20.913W**

- At 12V

* WF-P20/15 electromagnet 230mA

12V \* 230mA = 2.76W

- At 9V

* 2 x DRV8825 stepper motor controllers at 250mA

9V \* 250mA \* 2 = 2.250W

- At 5V

* Raspberry Pi 4 at 3A, this is the suggested minimum value for the current supplied, even though it will likely draw less than 3A [3].

5V \* 3A = 15W

- At 3.3V

* STM32F407BGT6 microcontroller at 240mA [4]

3.3V \* 240mA = 792mW

* RN4020 Bluetooth Receiver/transceiver at 33.6mA, this value is the maximum value drawn by the transceiver at full power [7]. We will likely reduce the power of the transceiver so the current draw is down to 16mA, because the extra power will likely be unnecessary in our application. 33.6mA is being used to represent worst case usage.

 3.3V \* 33.6mA = 110.88mW

-*Wireless Dice*

The power will be provided by a rechargeable battery that is wirelessly charged. This will provide 5V which will be stepped down to 3.3V and 1.8V.

The maximum current draw for each die is: 7.7mA + 0.810mA = **8.51mA**

The total power consumed by each die is: 3.3V \* 7.7mA + 1.8V \* 0.81mA = **26.87mW**

-3.3V

* SPBTLE-1S microcontroller at 7.7mA [5]

3.3V \* 7.7mA = 25.41mW

-1.8V

* LSM6DSL IMU at 810μA [6]

1.8V \* .810mA = 1.458mW

3.0 Interface Considerations

The main interfaces in this project are:

1. STM to Raspberry Pi: The Raspberry Pi will receive game state information from the STM via wired UART, with a baud rate limit of 115,200 Hz [2].
2. Raspberry Pi to monitor: The Raspberry Pi has a built in micro-HDMI output port, which will connect to the DVI-D port in the computer monitor via a converter/dongle. DVI-D has a maximum data rate of 1.65 GBit/second [1].
3. Die to STM. The die will communicate wirelessly to the STM via Bluetooth using a UART interface. The microcontroller on the die already contains a Bluetooth communication module, while the STM will use a RN4020 module. The baud rate will be at 115,200 Hz [2].
4. STM to motor. The STM will use PWM to send movement commands to the DRV8825 stepper motor controller via the necessary GPIO pin. The rate for this is expected to vary between 50 Hz and 3 kHz depending on the movement required, since the frequency controls the speed of the motor.
5. User controls (buttons) to STM. We will connect button outputs to GPIO pins on the STM, and track rising edges as button presses, with debouncing.

4.0 Sources Cited:

[1] Drhdmi.eu. (2019). Digital Visual Interface. [online] Available at: http://www.drhdmi.eu/dictionary/dvi.html [Accessed 12 Sep. 2019].

[2] Sciencedirect.com. (2019). *Baud Rate - an overview*. [online] Available at: https://www.sciencedirect.com/topics/engineering/baud-rate [Accessed 12 Sep. 2019].

[3] “Raspberry Pi 4 Model B specifications – Raspberry Pi,” *Raspberry Pi 4 Model B specifications – Raspberry Pi*. [Online]. Available at: https://www.raspberrypi.org/products/raspberry-pi-4-model-b/specifications/. [Accessed: 13-Sep-2019].

[4] “STM32F407VG,” *STMicroelectronics*. [Online]. Available at: https://www.st.com/en/microcontrollers-microprocessors/stm32f407vg.html. [Accessed: 13-Sep-2019].

[5] “SPBTLE-1S,” *STMicroelectronics*. [Online]. Available at: https://www.st.com/en/wireless-transceivers-mcus-and-modules/spbtle-1s.html. [Accessed: 13-Sep-2019].

[6] “LSM6DSL,” *STMicroelectronics*. [Online]. Available: https://www.st.com/en/mems-and-sensors/lsm6dsl.html. [Accessed: 13-Sep-2019].

[7] “RN4020,” *RN4020 - Bluetooth Module*. [Online]. Available: https://www.microchip.com/wwwproducts/en/RN4020#datasheet-toggle. [Accessed: 13-Sep-2019].

Appendix 1: System Block Diagram