Wiring Setup:

1. The project uses the Xiao ESP32-S3 board with a 4-bit digital-to-analog converter on GPIO pins D0, D1, D2, and D4, connected to a resistor ladder with resistors of 1 kΩ, 2 kΩ, 4 kΩ, and 8 kΩ.

2. Four push buttons are connected to pins D5, D8, D9, and D10. These buttons are each assigned to play a musical note: C, D, E, and G. The buttons are configured to be active-high with pull-down resistors.

3. The DAC output should be connected to the input of the actual sound-producing device, whether a speaker or a piezo buzzer, in addition to the fact that the ESP32 is powered on both 5V and 3.3V lines with a common ground for all associated components.

Modifications to the Original Code:

1. Conversion of DAC from 3 bits to 4 bits means that 4 GPIOs, namely, D0, D1, D2, and D4, are allowed, hence providing 16 unique output levels (0-15)

2. 4 push buttons were added to produce notes C, D, E, and G on the piano. Each note generates a different timer period: 120 µs for C, 106 µs for D, 95 µs for E, and 80 µs for G.

3. Sound frequencies were changed from a default value of 100 Hz to musically accurate ones: C = 523.251 Hz; D = 587.330 Hz; E = 659.255 Hz; G = 783.991 Hz.

4. An ESP32 Timer API was used for creating waveforms.

5. The main loop checks which key is being pressed, sets the timer interval to register this, and when no key is functioning, turns off the DAC (output 0).

6. Serial print statements were included to show the workings of note-playing in real-time for debugging purposes.