Cairo University

Faculty of Engineering

Cairo University

Graduation Project: Digital Randomizer

Project Overview:

Project Title: Digital Randomizer

Project Deadline: 11:59 PM Friday 22/9/2023

Project Team: Individual

Project Description:

The Digital Randomizer project aims to develop a digital device that randomize integer numbers using an ATmega328P microcontroller. There are 2 MCUs, one is the processing device and the other one is the human interface one. The HMI MCU is connected to 1 7-segments, 5 LEDs, 3 push buttons, servo motor, buzzer. And the processing MCU is connected to 3 LEDs.

Project Objectives:

- **Randomizer:** Create a digital Randomizer.
- **Random Number Generation:** Implement a random number generation algorithm to ensure fair and unbiased results without using built-in functions.
- **Display:** Use One 7-segments and 5 LEDs to visually represent the number.
- **Audio Feedback:** Integrate a buzzer to provide an audible indication of generated number with a signal bias from the read value of potentiometer.
- **User Interaction:** Design user-friendly 3 push buttons control the digital Randomizer.
- **Low Power Consumption:** Optimize the device for low power consumption to prolong battery life.

Deliverables:

1. Simulation
2. Video for running simulation and showing all the cases.
3. Code

Project Components:

- 2 ATmega328P microcontroller (Arduino UNO)
- 5 LEDs for displaying numbers.
- 3 LEDs for show the state of the processing MCU.
- Buzzer for audible feedback.
- Servo motor for angular feedback.
- 3 Push buttons for rolling the Randomizer.

Methodology:

- **The user should press on the "Generate" push button, then the HMI MCU sends to the processing MCU to generate a random number.**
- **Once the processing MCU receives the request and validate it, the 1st connected led to the processing MCU should turn ON, and after one second, the second LED should be turned on, which indicates the random number is generated and ready to be sent, then after one second, the processing MCU should respond to the HMI MCU with the random number and trun on the 3rd led. Turn off all the leds after one second. If there is something wrong in the synchronization between the 2 MCUs, let the 3 LEDs toggle for 5 seconds (toggle every 500 ms).**
- **After the HMI received the generated number, it should be represented in 5 digits so show digit by digit and turn on the led corresponding to the order of the digit in the number from the 5 LEDs and each digit should last for one second. The buzzer should release a sound with duty cycle equal to the ratio of the digit value / 10 and after the 5 seconds, then the duty cycle should be the random number / 65535.**

- **HMI should also move the servo to position (random number % 180) in a time period of 5 seconds.**
- **If the user press on the “Next” push button, then the HMI MCU should ignore the current task and then show the next digit quickly. Same if the user press on the “Previous” push button.**
- **If the user pressed on “Generate” push button before the MCU finish the previous task, ignore the current task and start the new task immediately.**
- **Keep saving the current state of the system (The gnereated random number, the current displayed digit, …) in the EEPROM and if the system is reset or turned off, then turned on, you can start from the last state.**

Note: you must use all of ADC, timers, UART, I2C and external interrupts.

Conclusion:

The Digital Randomizer project aims to create an entertaining and engaging device that simulates the Randomizer in a digital format. This project leverages the capabilities of the ATmega328P microcontroller, timers, ADC, 7-segments, LEDs, potentiometer, and a buzzer to provide a unique user experience. With the communication protocol synchronization and how to make the system stateful and continue from the last state of the system.

Thanks, and Good Luck

Ahmed Alaa