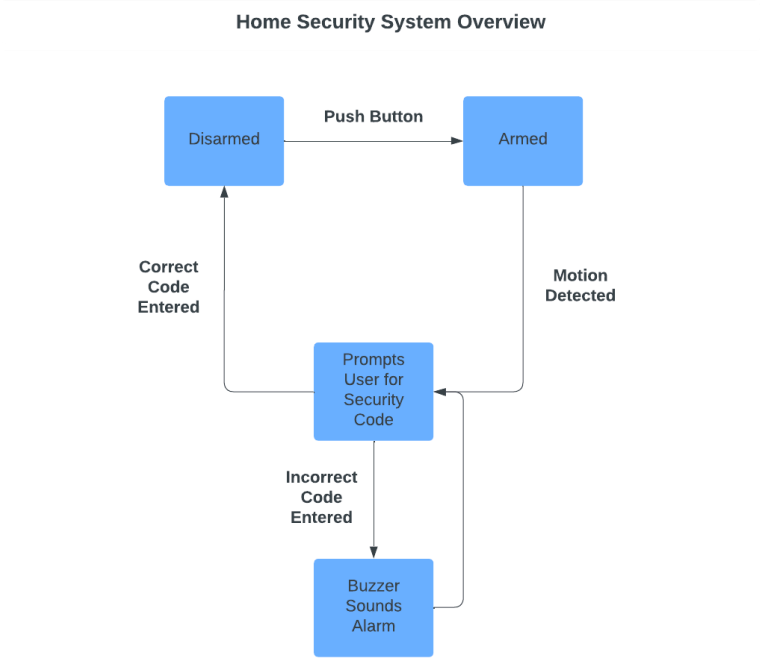
ECEN 501L Final Project Report: Home Security System

Overview of Design



The design we chose to go with involved four main states: Disarmed, Armed, Alarm, and Siren. We used a multitude of tasks to balance these states and ensure safe and reliable transitions. As a prerequisite, we had to find ways to intertwine tasks to share resources and time. This entailed the utilization of critical sections, semaphores, thread flags and a queue. We also made use of the onboard LEDs, IMU accelerometer, hardware timer, wifi module, and external interrupt functionality.

The system is first initialized, starting with the main() function where the hardware peripherals, FreeRTOS kernel and tasks are initialized. System tasks are created and scheduled, and the RTOS scheduler starts and takes control of the system.

The first state entered is the default DISARMED state. The LED is off and the wifi server is initialized so that the user can monitor or control the system via the web interface. The state will change if a certain event occurs, a button press. When the user presses the button the GPIO interrupt triggers a callback and releases a semaphore. The state transitions to ARMED. In the ARMED state the StartArmed task continuously monitors the accelerometer for motion using ReadAccelero(). If motion is detected beyond the specified threshold the state then transitions to ALARM and activates an alarm/LED to signal this change. In the ALARM state the StartAlarm task engages and awakes a user interaction, prompting a 4-digit code via UART to be entered in order to disarm the system. If the correct code is entered, the system transitions back to the ARMED or DISARMED state. If the incorrect code is entered the user has up to five attempts to disarm the alarm and after five incorrect entries the system transitions to the SIREN state. In the SIREN state a buzzer/LED is activated.

At any point the user is able to interact with the system using a web interface. The user can view the current state which could be DISARMED, ARMED, ALARM, or SIREN. They can also control the system by doing actions such as toggling the LED, stopping the server or changing states.
If the state is DISARMED, the LED is off. If stat is ARMED the LED is on. If the state is ALARM the LED is blinking and in SIREN state another LED or GPIO pin is toggled rapidly.

Implementation Details

Module Names

- HAL_GPIO_EXTI_Callback
- ToggleState
- ActivateBuzzer
- Wifi_start
- Wifi_server
- WebServerProcess
- WebServerProcessUpdate
- SendWebPage
- SPI3_IRQHandler
- ReadAccelero
- ModelAccelero
- StartDefaultTask
- StartArmed
- StartAlarm
- StartArmed

Technologies Used

The embedded technologies utilized within this home security system include the STM32, HAL, GPIO, UART, I2C, SPI< USB OTG and timers. The STM HAL provides high-level APIs to interact with the STM32 hardware peripherals like GPIO, SPI, I2C, TIM, and UART. CMSIS-RTOS allows for the use of RTOS for task management with multithreading, synchronization, and semaphores implemented for concurrency.

Sensors are another technology that is used, specifically the accelerometer which is used to detect motion and trigger the prompt for user input if motion is detected. A wireless communication technology is used for wifi module integration. This allows an HTML web server to be created, which dynamically updates the web page to reflect system state using HTTP GET and POST methods.

Semaphores are used for inter-thread communication, allowing tasks to release and acquire semaphores. Queues are utilized for task to task communication. Event driven programming is also used, with interrupts and RTOS events triggering state transitions and task actions. Structure programming allows for code organization with clearly defined functions for initialization, hardware abstraction and state handling.

Special Features

One of the special features of this system is the state machine that controls system modes. The different system states are DISARMED where the system is inactive, ARMED where the system is monitoring for specific events such as motion detected by the accelerometer, ALARM which is triggered when an anomaly like the motion is detected, and SIREN which is an elevated alarm state for severe conditions where the security code is incorrectly entered.

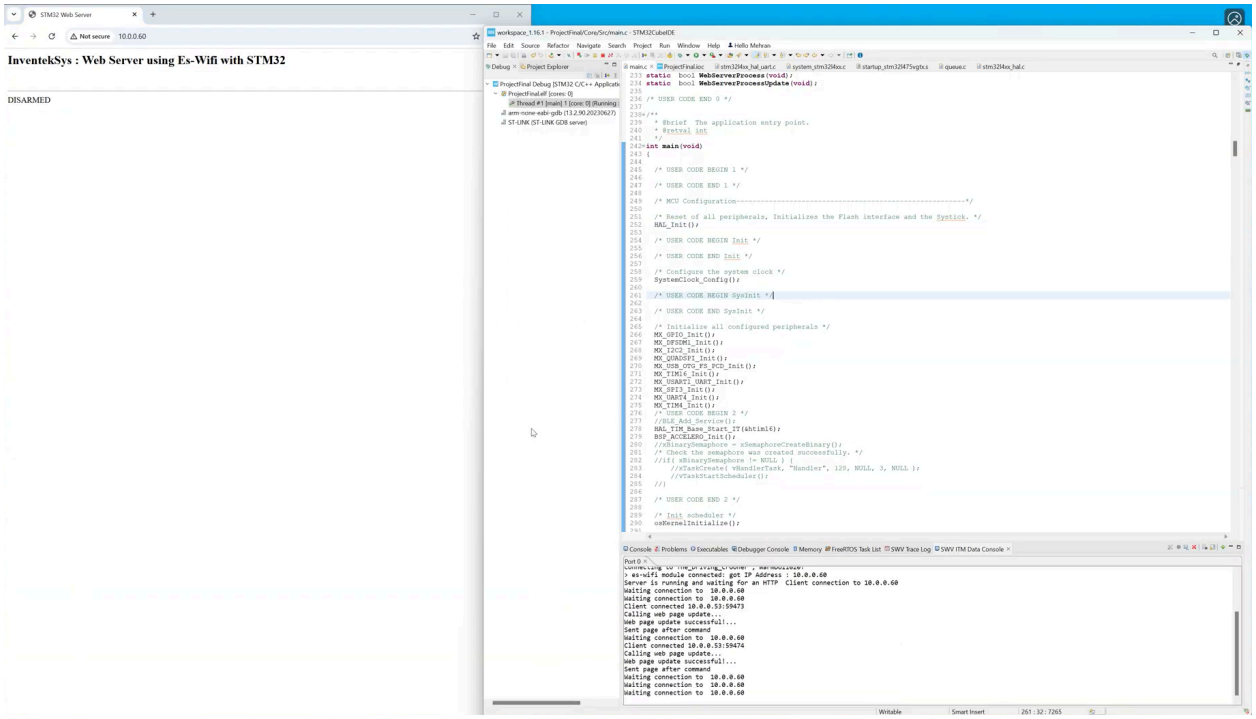
Another special feature is the web server for IoT integration. The system includes a basic HTTP web server, serving an HTML page with status updates. This feature also allows for remote control via HTTP POST requests. The server displays the system status and allows the user to indirectly control the system’s state through the web interface by disabling or enabling the alarm.

The use of the accelerometer to detect motion is another feature of the system. There is logic included that is able to read accelerometer values and detect motion based on a designated tolerance threshold, allowing the system to differentiate between STATIONARY and MOVING states of the device. There is also a moving count threshold to determine when to trigger the alarm.

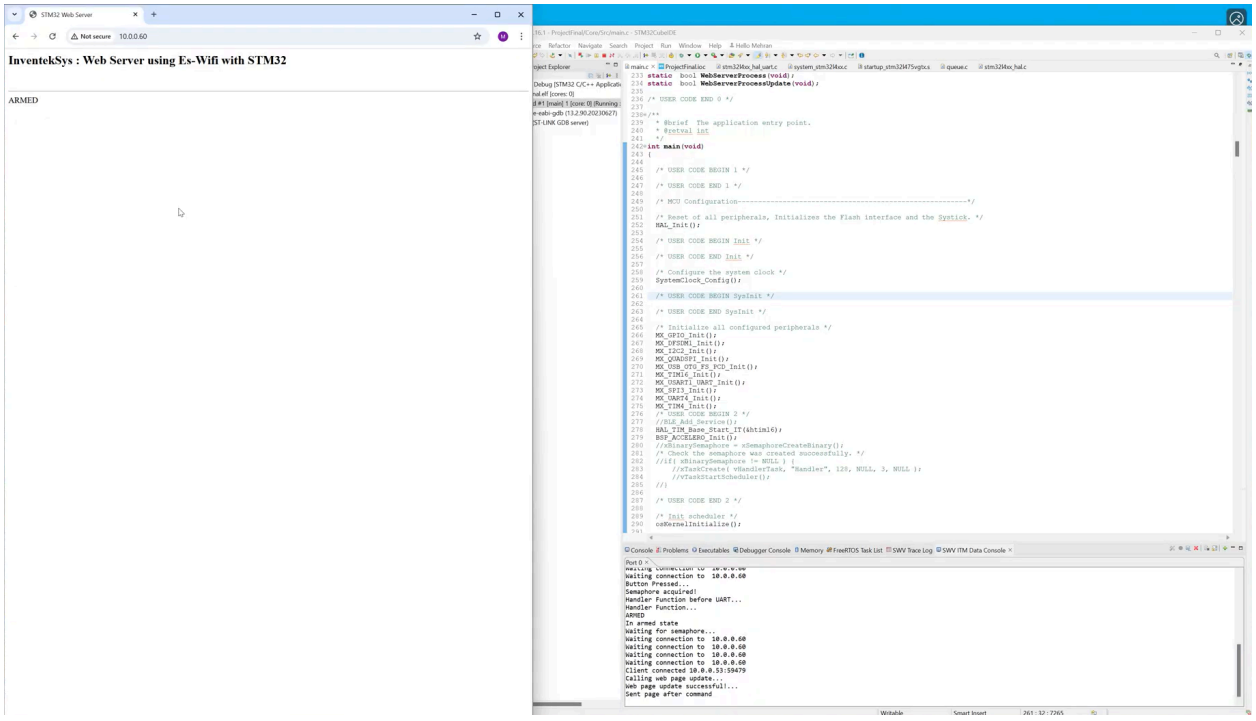
There is also an interactive security mechanism feature that implements a 4-digit security PIN verification system to disarm the security system if motion is detected. The user is prompted to enter a PIN via the UART if the accelerometer detects motion above the desired threshold. The correct PIN will disarm the system, while multiple incorrect attempts escalate the alarm.

Outputs

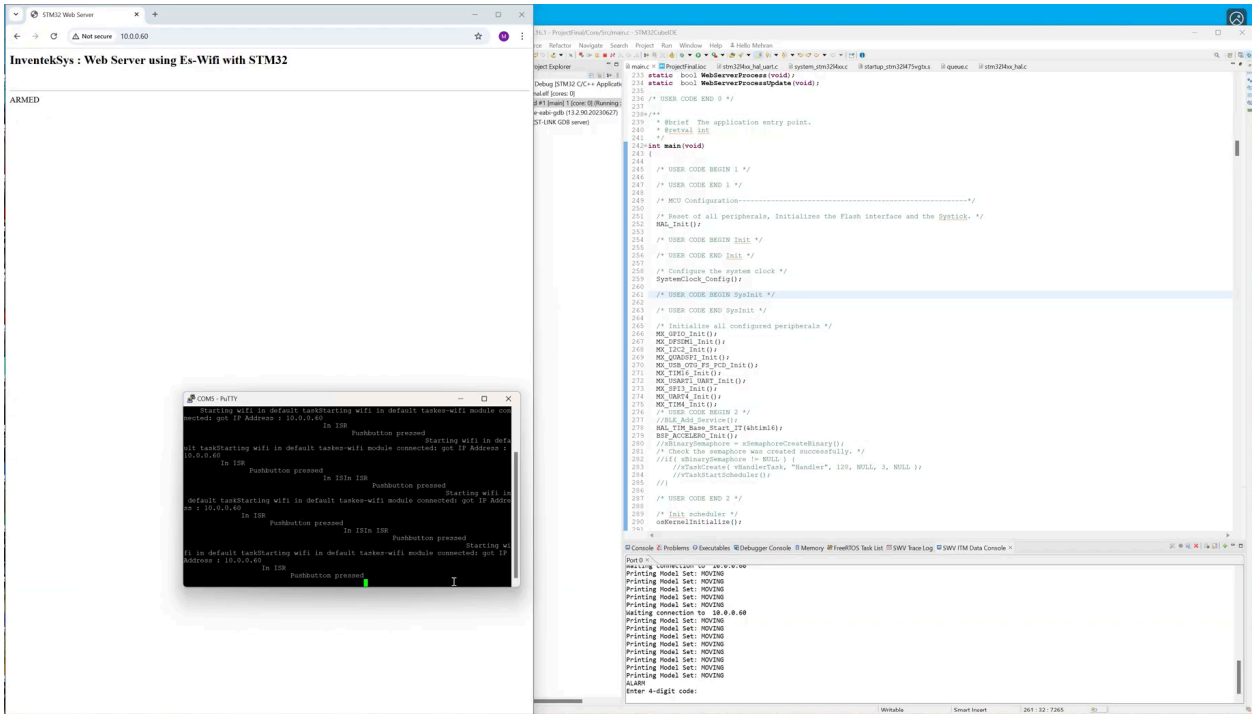
When system is started:



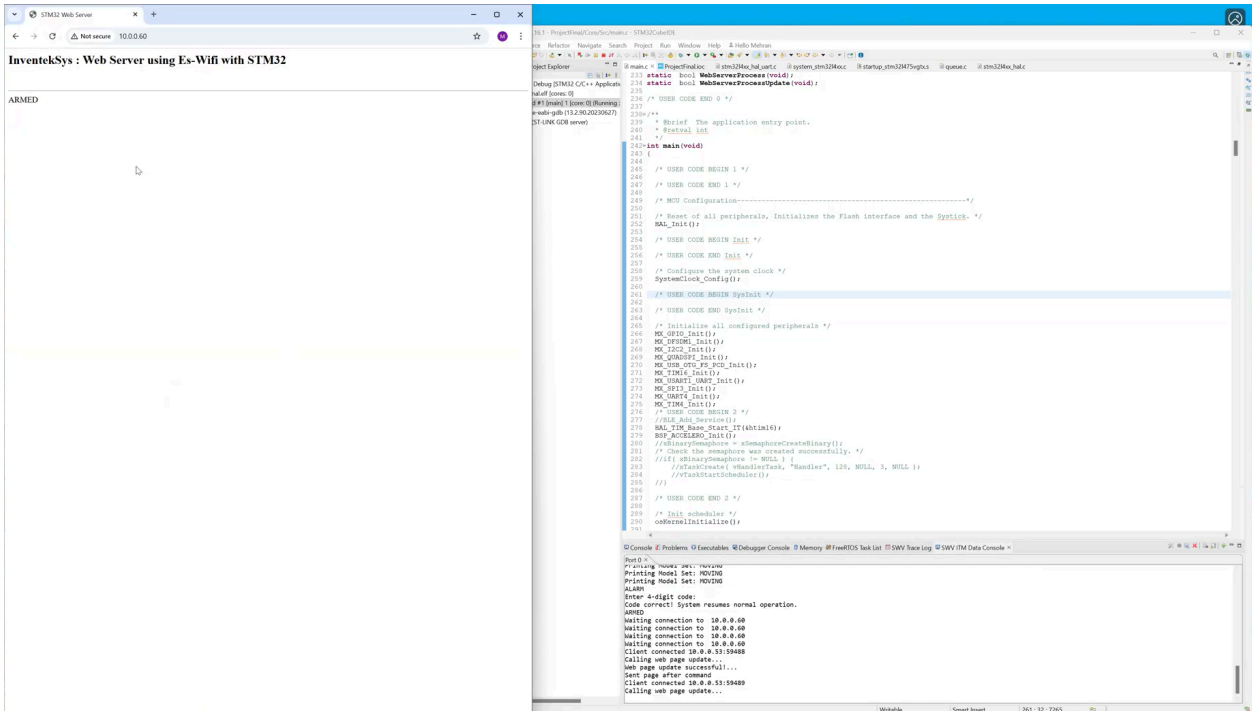
When button is pushed:



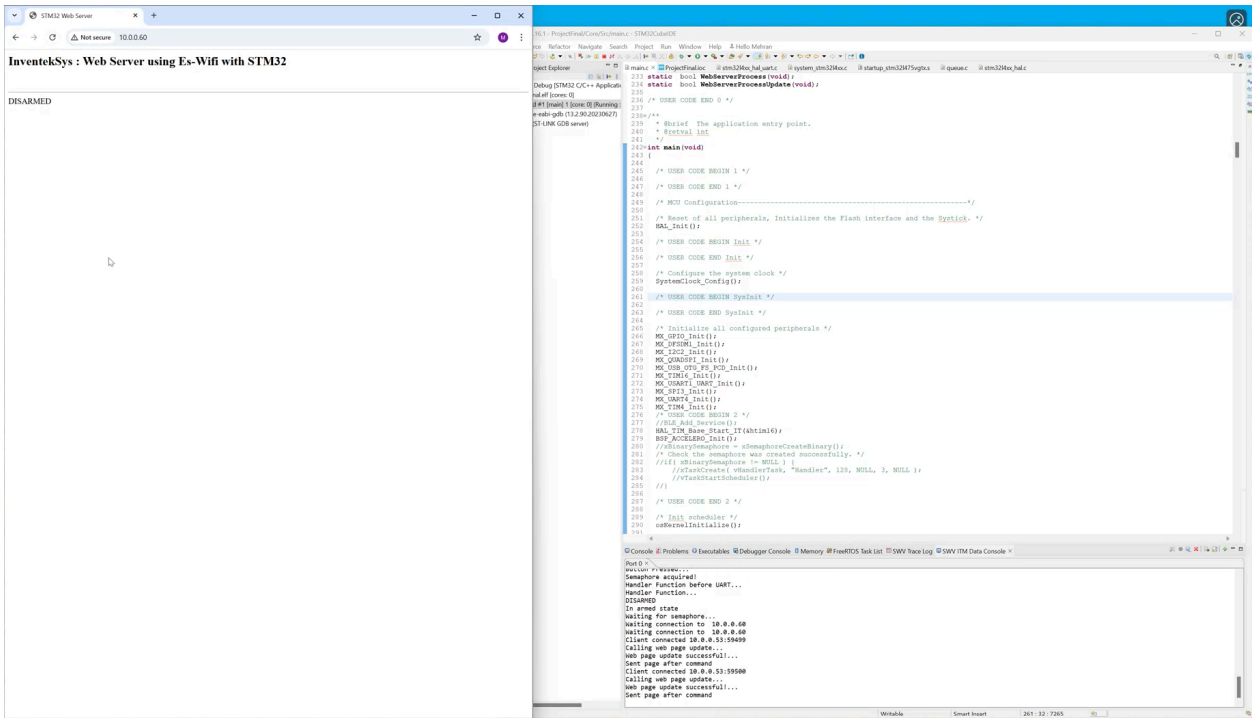
If microcontroller board is shaken:



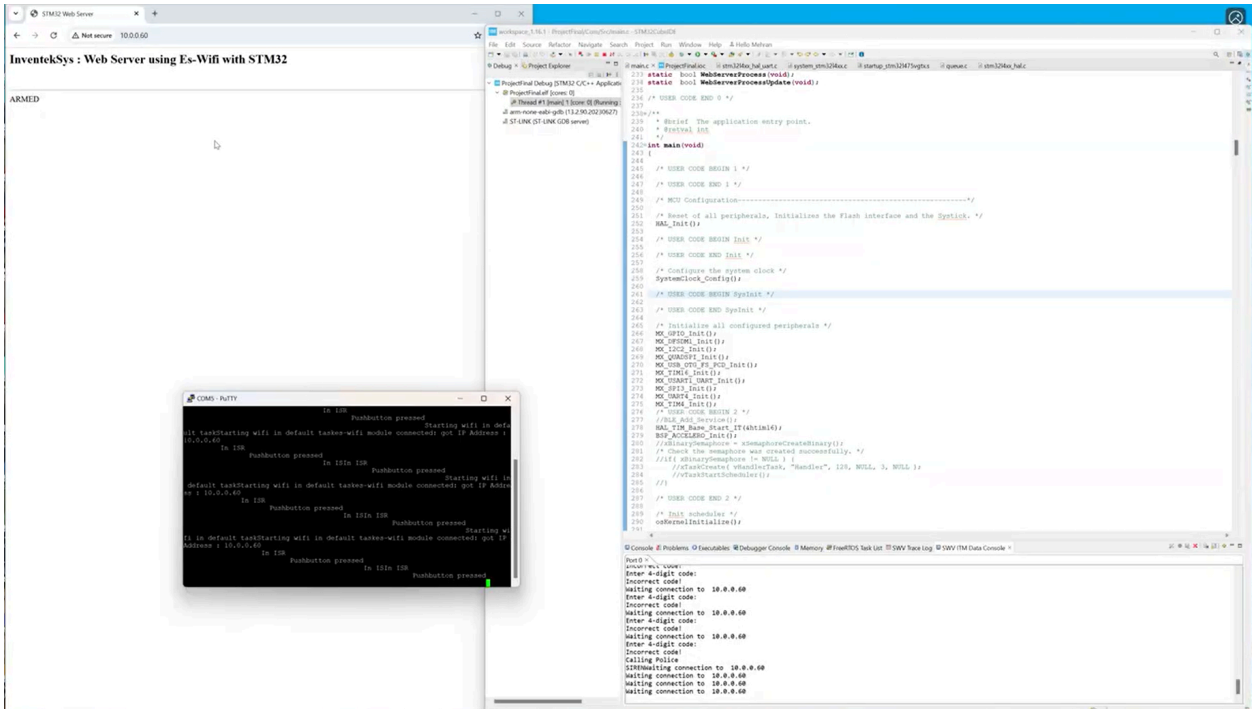
After correct code is entered:

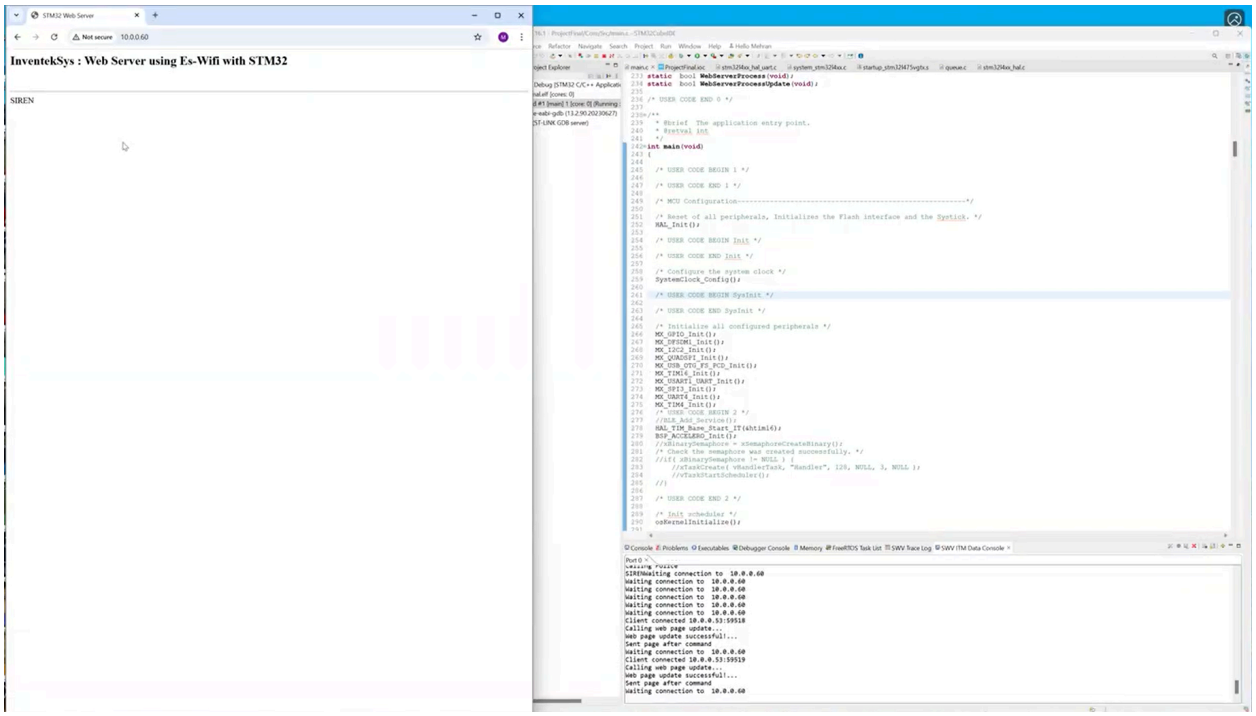


Using push button to disarm:



If system is armed, motion is detected, and 5 incorrect codes are entered, enter SIREN state:





Conclusion

This security system is a real time, IoT enabled security and monitoring system using an STM32 microcontroller and FreeRTOS. The system has four primary states: DISARMED, ARMED, ALARM, and SIREN transitioning based on motion detected by the accelerometer or user input. A button press toggles between DISARMED and ARMED, while detected motion triggers the ALARM state, which requires the user to disarm the system with a 4-digit code ented through the UART. The user failing to enter the correct code within 5 attempts escalates the system to the SIREN state which activates a buzzer/LED blinking. The system also features a web interface served via wifi. This allows the user to remotely monitor and control the state. This design is well suited for applications like home security, motion sensitive monitoring, or IoT enabled alert systems.

References

- https://www.freertos.org/
- https://github.com/STMicroelectronics/STM32CubeL4/tree/master/Projects/B-L475E-IOT01A/Applications/WiFi
- https://www.keil.com/pack/doc/CMSIS_Dev/RTOS2/html/index.html
- https://www.st.com/en/evaluation-tools/b-l475e-iot01a.html#documentation