University of Michigan-Dearborn

CIS 450 - Operating System

Final Project Technical Documentation

Haidar Fakhruldin

Will Yang

Dilraj Dhillon
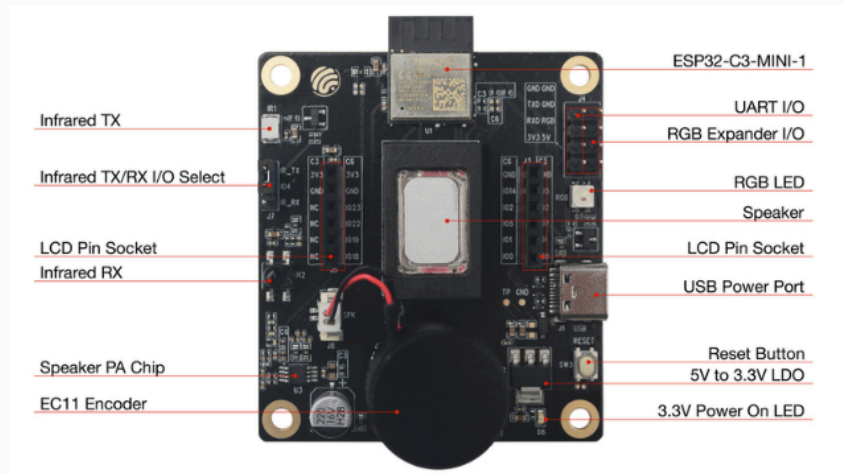
December 13, 2024

Github Repository: https://github.com/HaidarFak/450-final

System Architecture:

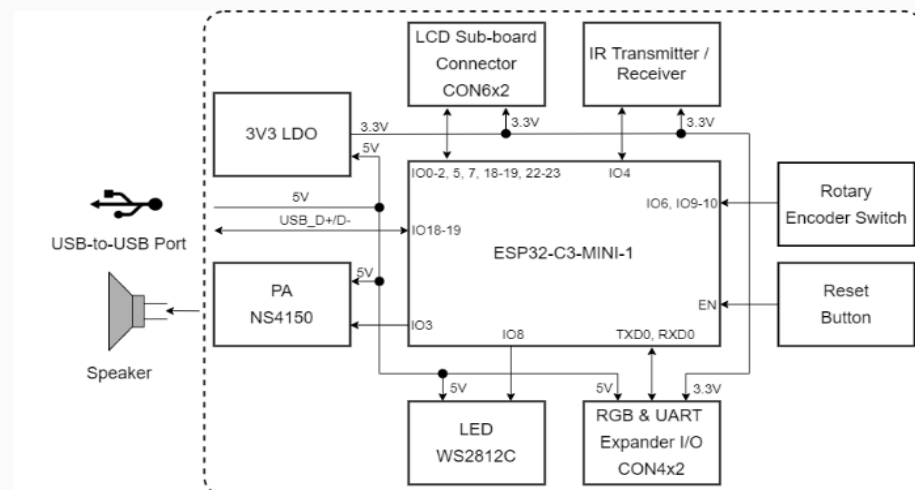**Components of the ESP32-C3-LCD kit:**

Mainboard

ESP32-C3-LCDkit_MB is the core of the kit, which integrates the ESP32-C3-MINI-1 module and provides ports for connection to the LCD subboard.

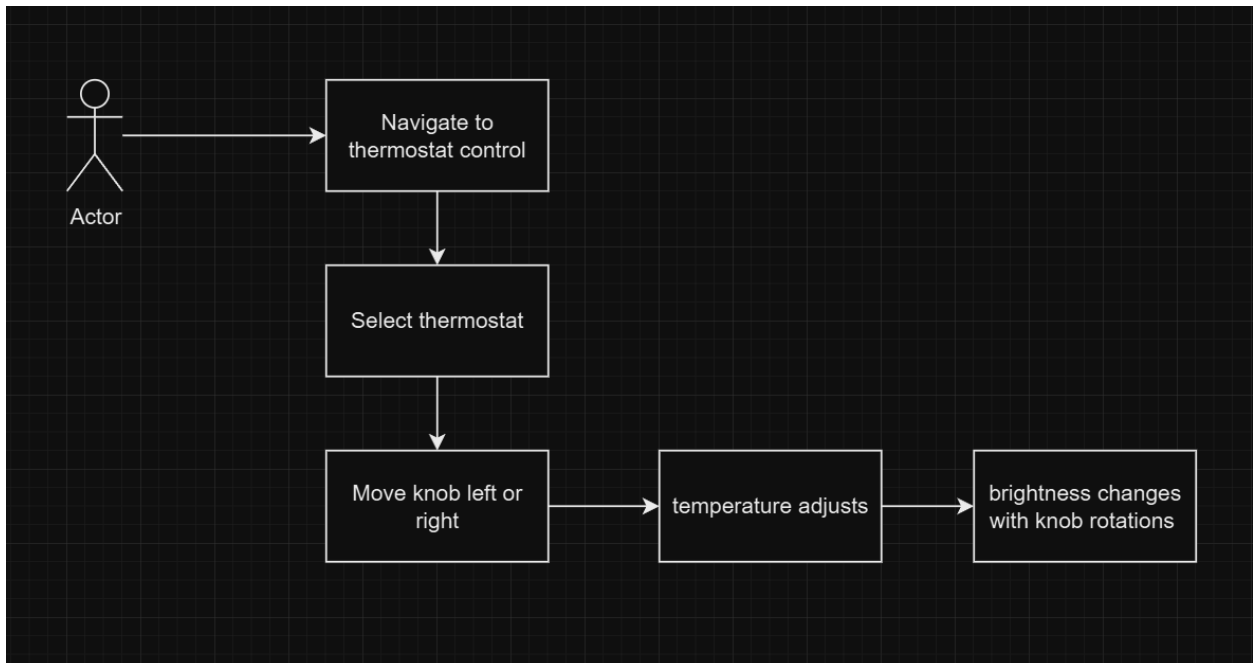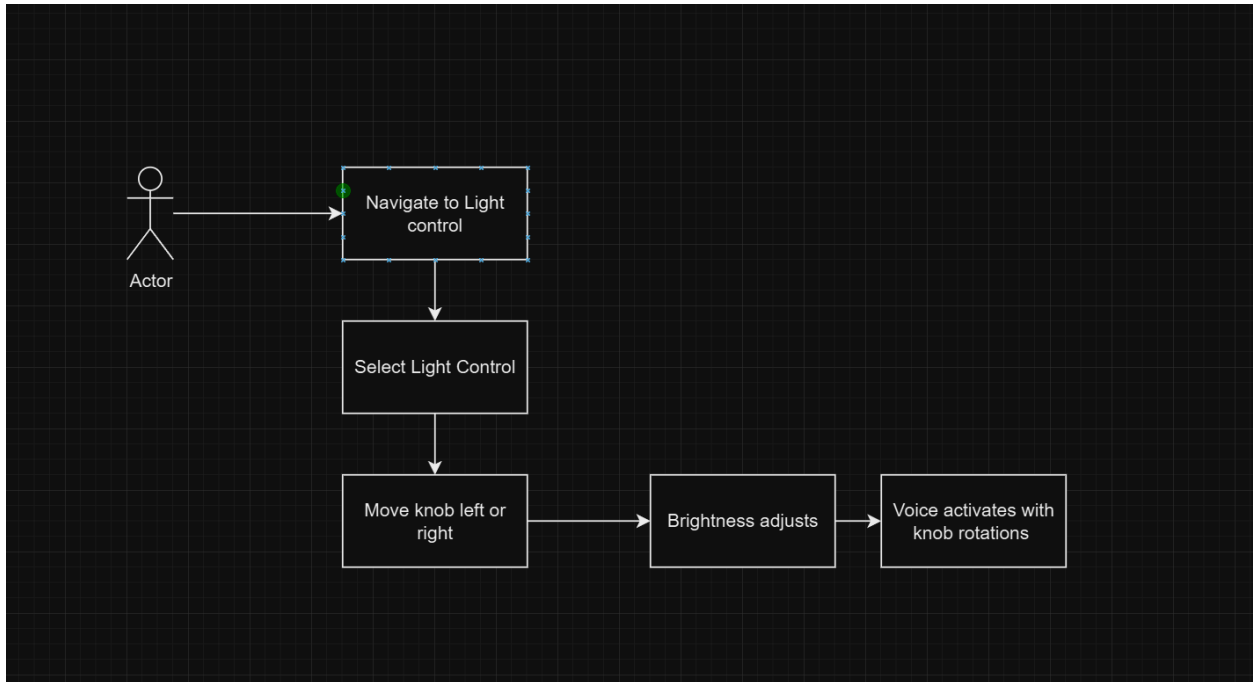*ESP32-C3-LCDkit - Front (Click to Enlarge)*

Block Diagram

The block diagram below shows the components of ESP32-C3-LCDkit and their interconnections.

ESP32-C3-LCDkit Block Diagram (Click to Enlarge)

https://docs.espressif.com/projects/esp-dev-kits/en/latest/esp32c3/esp32-c3-lcdkit/user_guide.html

Diagram 1

Actor → Navigate to Light control → Select Light Control → Move knob left or right → Brightness adjusts → Voice activates with knob rotations

Diagram 2

Actor → Navigate to thermostat control → Select thermostat → Move knob left or right → temperature adjusts → brightness changes with knob rotations

List of tasks implemented

Knob Input Task:

Purpose: Reads the rotary knob's position, which determines the desired LED light intensity.

Light Adjustment Task:

Purpose: Adjusts the LED brightness based on the input received from the Knob's Input.

Voice Announcement Task:

Purpose: Announce the current light level via a speaker to the user.

Interaction: Receives light-level data and generates voice output using preloaded audio files to announce the light level.

      The ESP32-C3-LCD kit comes with the FreeRTOS operating system. It is an open-source real-time operating system kernel designed for embedded systems. It is widely used for robotics, automotive systems, and more. The tasks above are grouped by a FreeRtos function, xEventGroupCreate(). It synchronizes multiple tasks by bit-level flags. Then the tasks are made sure to not access shared resources at the same time by the xSemaphoreCreateMutex() function in FreeRTOS. This is to prevent race condition.

This is Task States:
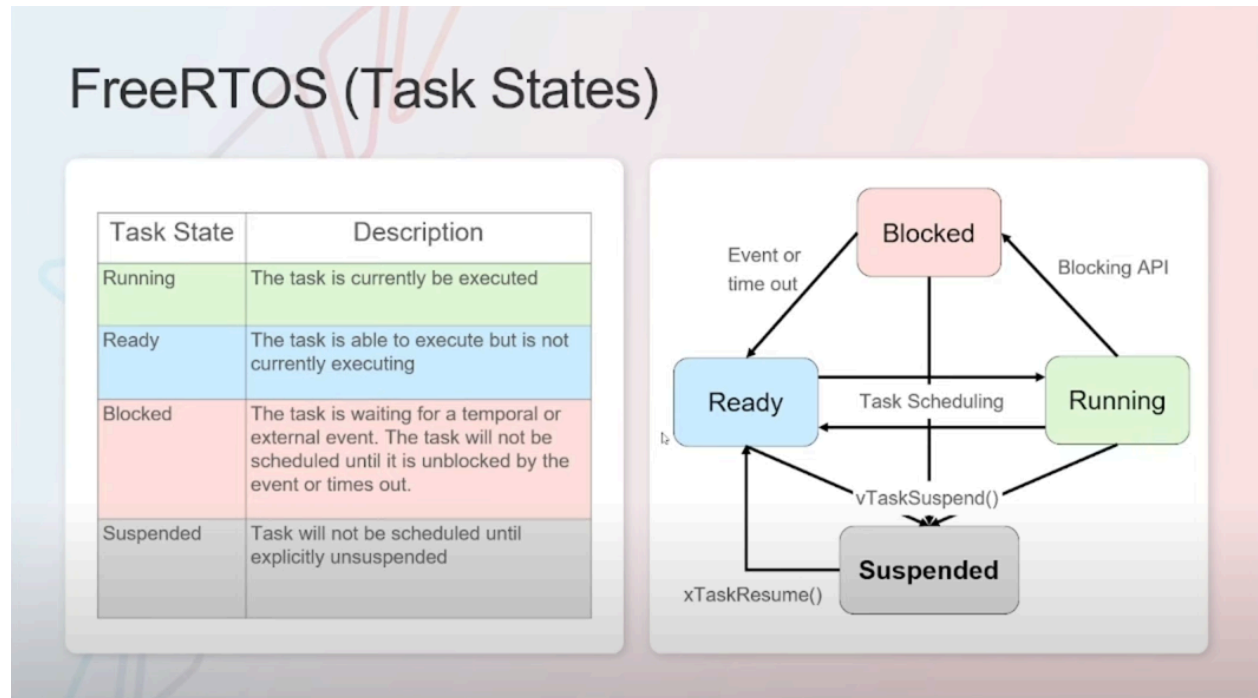
Reference: https://dl.espressif.com/public/DevCon23_ESP-IDF_Getting_Started.pdf
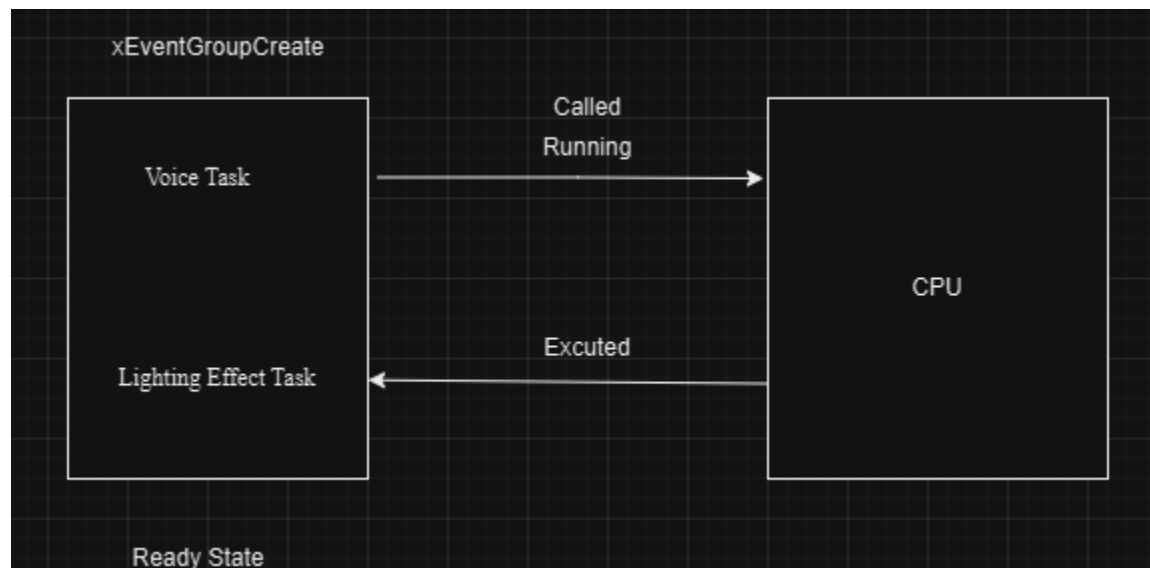
Concurrency Control:

We used mutexes to prevent issues with concurrency in regards to the lighting variable and the display to protect the shared resources between the two. Semaphores were used in the voice announcement feature so that the audio only plays when the brightness level is changed.

User Guide:

1.Power the device on

2.Turn the knob to select the lighting option and press down on the knob to open the adjustment

interface.

3. To increase the brightness of the light turn the knob to the right and to decrease the brightness

turn the knob to the left. After each level audio should be played corresponding to the brightness

level.

4. To exit press the knob down and you should return to the main menu

Bonus Feature:

Description:

The bonus feature that the group when with is the brightness of the led will work based on the set temperature. So when the temperature is set to 19 celsius degrees the light brightness will be set to 5 percent and so on for the other temperatures and brightness levels.

Usage instructions:

Once the device is turned on navigate the menu using the knob once hovering over the temperature menu option press down on the knob then adjust the temperature moving the knob either left or right, while the temperature is being adjusted the light level will adjust accordingly towards the correct brightness.