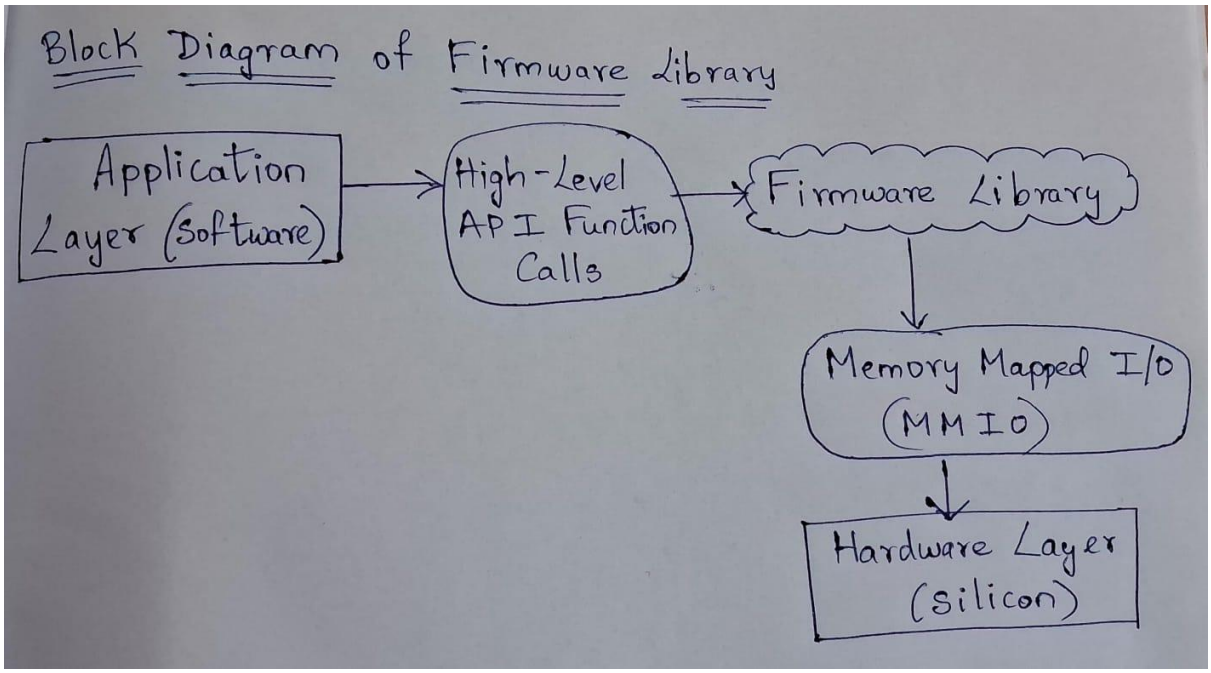
RISC-V Embedded Firmware Internship

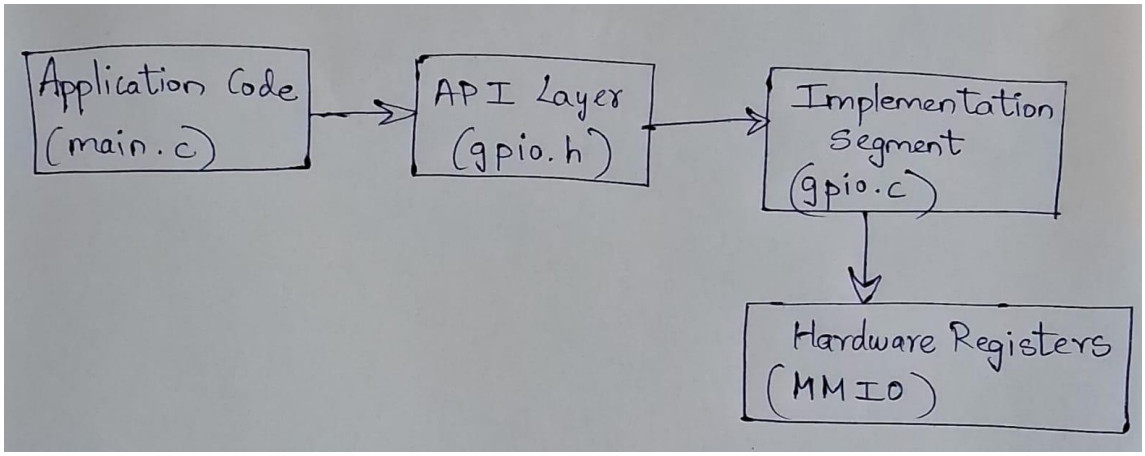
Task 1: Firmware Foundations & Environment Setup

1. **What is a firmware library?**
    - Firmware is the bridge between physical hardware and software logic. Firmware manages low-level logic and sensory inputs which is stored in non-volatile memory such as flash/ROM.
    - Firmware library is a structured collection of C functions which has direct control over the hardware register access through fixed memory addresses.



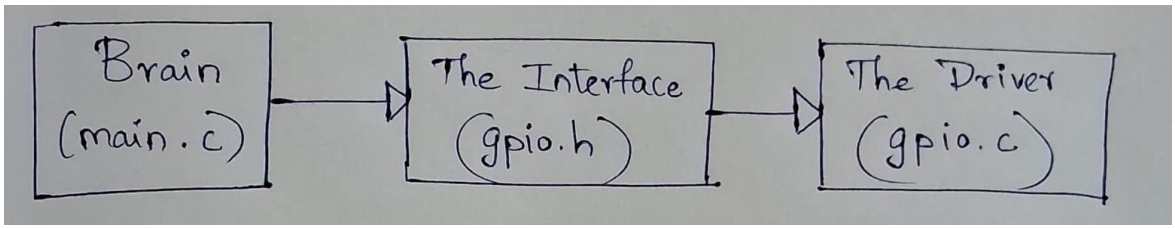
2. Why APIs are important in embedded systems?

- APIs are necessary in embedded systems because it can manage the interaction between various software and hardware components.
- The functions like digitalWrite() and delay() helps in register management instead of writing directly to the memory addresses(registers).



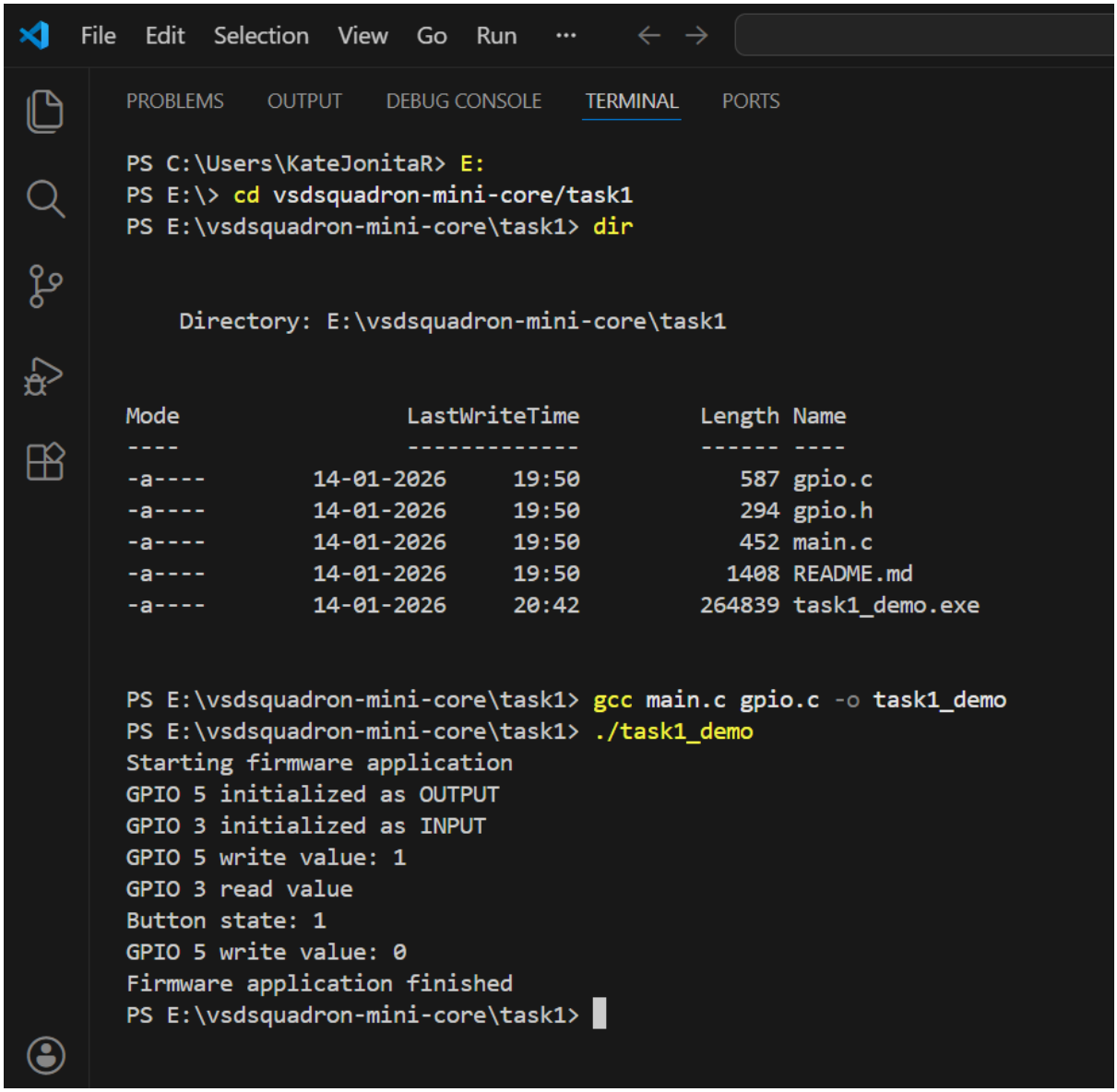
3. What was understood from the lab code?

- The code can be organized into layers for easy management and for a smoother execution of the task.
- main.c is used for function calling and implements system logic.
- gpio.h defines the interface and declares what functions exist.
- gpio.c performs register-level work.



Lab – Understanding a simple firmware library

Console logs showing simulated GPIO behaviour




```
PS E:\vsdsquadron-mini-core\task1> Executing by KateJonita> gcc main.c gpio.c -o task1_demo
PS E:\vsdsquadron-mini-core\task1> Executing by KateJonitaR ./task1_demo
Starting firmware application
GPIO 5 initialized as OUTPUT
GPIO 3 initialized as INPUT
GPIO 5 write value: 1
GPIO 3 read value
Button state: 1
GPIO 5 write value: 0
Firmware application finished
PS E:\vsdsquadron-mini-core\task1> Executing by KateJonitaR
```