4-Week MCU Communication & Porting Project Schedule

▧ Week 1: Core C Concepts + OSI Model + UDP Chat

🎯 Focus Areas:

- OSI Model Understanding (with mapping to UDP)
- All foundational C brush-up topics
- Build a **C-based UDP Chat App**

▧ Topics:

- OSI Layers: mapping to real protocols
- State Machine
- Events, Array, Pointers.
- RTOS vs Superloop
- Scheduler, Threads
- Data Structures: Queue, Linked List, Circular Buffer
- Pointers: structure/function pointers, callback functions
- Bit operations & byte packing

🛠 Deliverables:

- 🧠 Document: Summary of OSI Model with protocol examples
- ☑ Code:
  - Basic Queue, Linked List, Circular Buffer implementations
  - Event-driven state machine for message exchange
  - UDP Chat App (Client-Server):
    - Send/Receive text
    - Uses command-line arguments

▨ Week 2: Communication Tech – Wi-SUN, BLE, RAIL, S2LP

🎯 Focus Areas:

- Learn BLE (LE), Sub-GHz (S2LP), RAIL, Wi-SUN
- Hands-on: Run **BLE and Sub-GHz range tests** on SiLabs platform

▧ Topics:

- BLE vs BLE LE: Advertising, GATT, connection events
- Sub-GHz RF concepts (S2LP), range, interference
- Silicon Labs RAIL APIs: init, send, receive
- Wi-SUN basics and stack setup on EFR32
- Range Test app architecture (BLE and Sub-GHz)

🛠 Deliverables:

- ☑ BLE Range Test using Silicon Labs BLE stack
- ☑ RAIL Sub-GHz test using S2LP or SiLabs

- Wi-SUN Test Setup (node join, CLI ping)
- Scripts to configure and evaluate range on BLE/Sub-GHz

▨ Week 3: RTOS & Porting – Zephyr, Mbed, Contiki-NG, Wi-SUN

🎯 Focus Areas:

- Understand and port project across embedded RTOS platforms
- Explore RTOS APIs: threading, message queue, timers

▧ Topics:

- RTOS Comparison: Zephyr vs Mbed vs Contiki-NG
- Platform Abstraction Layers
- Portability Design: Use of platform.c, radio_hal.h
- Build same app on:
    - **Zephyr (EFR32/STM32)**
    - **Contiki-NG**
    - **Mbed OS**

🛠 Deliverables:

- Portable shell app (UART CLI, state machine)
- Runs on all 3 RTOS (Zephyr, Mbed, Contiki)
- Porting guide: key files changed, OS abstractions
- Wi-SUN device join or basic radio test on Zephyr

▧ Week 4: Final Application – File Transfer using UDP + Full Integration

🎯 Focus Areas:

- Full system integration using advanced C concepts
- Implement CLI-driven UDP file transfer using command-line arguments
- Use callback, buffers, queues, retry logic

▧ Topics:

- File I/O in embedded C
- UDP fragmentation and reassembly
- Timeout/retry mechanism using timers
- BLE or Wi-SUN-based transport switching (optional)
- Logging, statistics, error handling

🛠 Deliverables:

- Complete File Transfer App
    - Args: filename, source path, destination path
    - Uses circular buffer, state machine, timer/callback
- BLE + Wi-SUN switch based on config
- Final Test Report

📦 Final Project Result:
“Cross-Platform Smart Node” supporting:

- BLE & Sub-GHz Range Test (Week 2)
- CLI Chat + File Transfer via UDP (Week 1 & 4)
- Platform Portability: Zephyr, Mbed OS, Contiki-NG (Week 3)
- Integration with Silicon Labs RAIL, Wi-SUN Stack, STM32 + S2LP