Simulate and Analyze the Performance of Various Multiple Access Techniques (FDMA, TDMA, CDMA) using Python

1. Problem Definition

Develop Python simulations to illustrate and analyze the behavior of Frequency Division Multiple Access (FDMA), Time Division Multiple Access (TDMA), and Code Division Multiple Access (CDMA). For each technique, visualize resource allocation, compute basic performance indicators (throughput, utilization, or SIR), and interpret tradeoffs.

2. Learning Objectives

- Understand the principles and resource allocation of FDMA, TDMA, and CDMA. - Implement simple simulations in Python to visualize frequency/time/code domains. - Compare performance under simple assumptions (e.g., equal user demands, fixed bandwidth, idealized channels). - Interpret trade-offs: spectral efficiency, interference, delay, fairness.

3. Learning Outcomes

After completing this experiment, students will be able to model and visualize multiple access schemes, compute basic metrics, and justify the preferred scheme for a given requirement profile.

4. Theory

FDMA partitions the total available bandwidth into non-overlapping sub bands allocated to users. It minimizes intra cell interference at the cost of guard bands and limited flexibility. **TDMA** shares a single channel in time; users take turns transmitting in non−overlapping slots within a frame, introducing delay but allowing narrowband hardware. **CDMA** spreads each user's signal with a (quasi-)orthogonal code over the whole band; users overlap in time and frequency, trading processing gain against multiuser interference.

5. Experimental Setup & Assumptions

Assume a system bandwidth of 12 MHz. For FDMA, we allocate channels to 6 users with guard bands. For TDMA, a 6 slot frame is used. For CDMA, direct sequence spread spectrum is illustrated with processing gain L=64. A rough interference-limited SIR model is used to study how the number of CDMA users affects link quality.

Figure 1: FDMA – Spectrum Partitioning

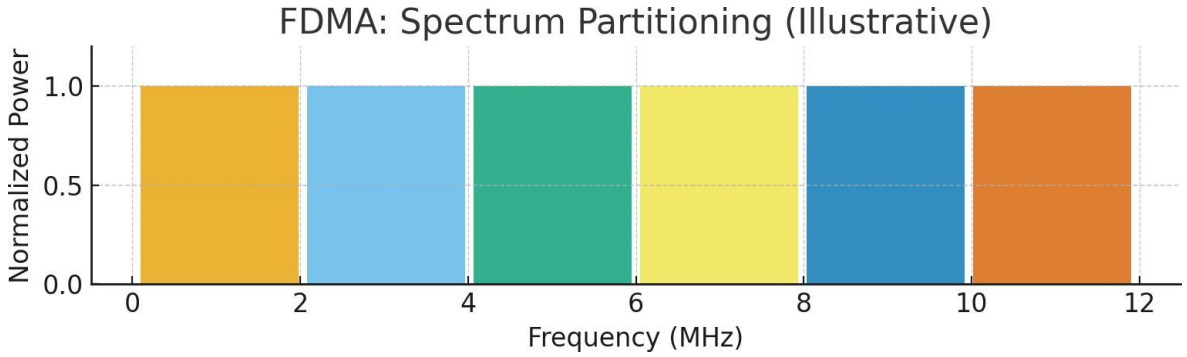


Figure 2: TDMA – Time Slots in a Frame

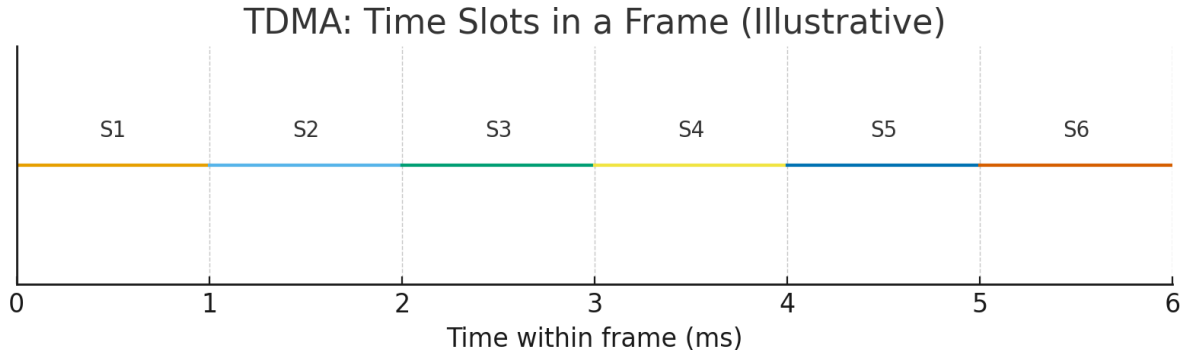


Figure 3: CDMA – DSSS Spreading Example (L=64)

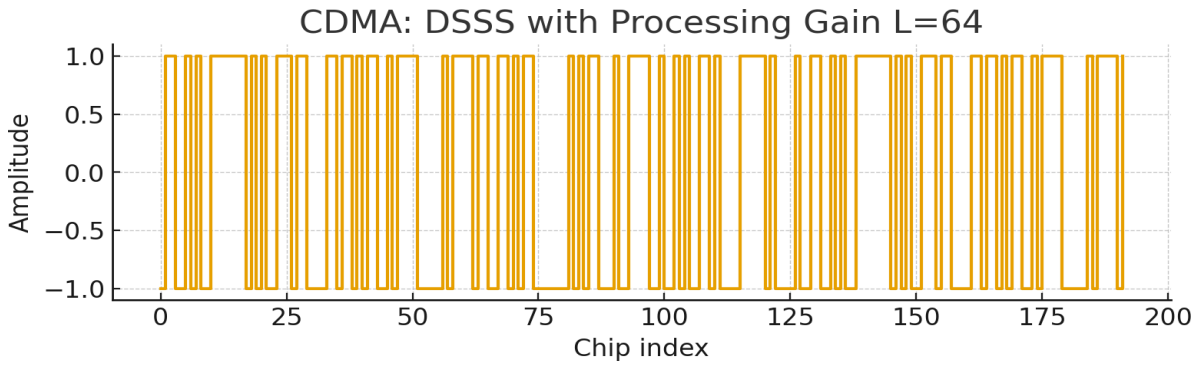
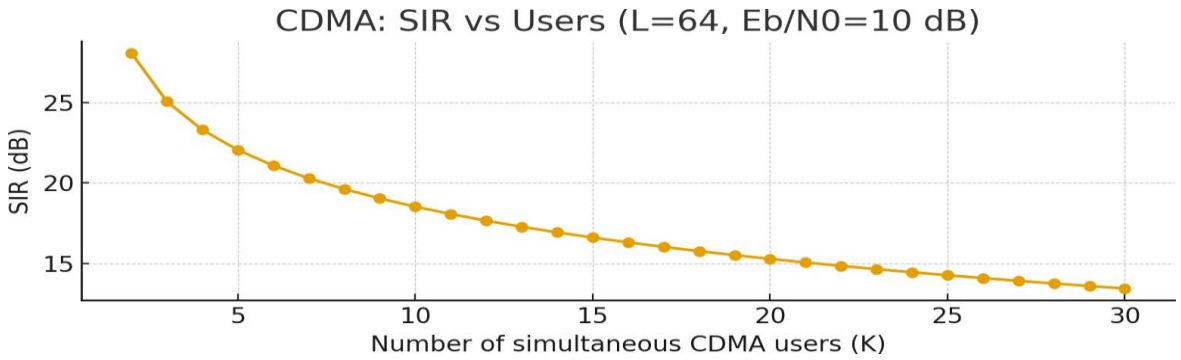


Figure 4: CDMA – SIR vs Number of Users


CDMA: SIR vs Users (L=64, Eb/N0=10 dB)
SIR (dB)
25
20
15
5
10
15
20
25
30
Number of simultaneous CDMA users (K)

6. Observations & Discussion

• FDMA: Simple isolation in frequency, predictable performance, but requires careful filtering and wastes spectrum in guards. • TDMA: Simple narrowband radios and deterministic access; suffers from slotting delay and requires tight timing. • CDMA: High user capacity under asynchronous access; multiuser interference and code correlation properties dominate performance.

7. Viva / Assignment Questions

Q1. Compare guard-band overhead in FDMA vs. timing overhead in TDMA.

Q2. Define processing gain in CDMA and explain its effect on SIR.

Q3. List at least two practical issues in implementing TDMA frames.

8. Conclusion

We implemented Python simulations to visualize FDMA, TDMA, and CDMA allocation mechanisms and analyzed simple performance metrics. The results highlight classic trade offs among spectrum partitioning (FDMA), time scheduling (TDMA), and code domain sharing (CDMA).