FWC COURSES

MODULE : 1 Two Month Duration

Module 1: Digital Design and Programming

This course covers the basics of digital design and programming, including installation, documentation, digital design, applied logic, assembly programming, embedded C programming, internet of things, ARM programming, and Verilog programming.

SNO	Module Name	Content	Credits	Weeks	Theory/Lab	Platform
1	Installation	Termux, LaTeX, Python Installation	0.5	1	Lab	Android
2	Documentation	Latex Exercises using Neovim	0.5		Lab	Android
3	Digital Design	Combinational and Sequential Logic using the Arduino Framework	2	1	Lab	Arduino board
4	Applied Logic	Porting digital logic to the arduino using Platformio	2	1	Lab	Arduino board
5	Assembly Programming	Introduction to AVR-Assembly, ATMEGA328P peripheral programming, Timers, Memory Management	2	1	Lab	Arduino board
6	Embedded C Programming	Introduction to AVR-GCC	1	1	Lab	Arduino board

7	Internet of Things	Establishing a wireless sensor network using the Vaman-ESP	1	1	Lab	Vaman Board
8	ARM Programming	Simple hardware interfacing using the Vaman-Cortex-M4	1	1	Lab	Vaman Board
9	Verilog Programming	Digital Design using the Vaman-EOS-S3 FPGA	2	1	Lab	Vaman Board
	Total		12	8		

Module 2: Advanced Digital Design and Programming

(2 Months)

This course covers advanced topics in digital design and programming, including advanced digital logic, microcontrollers, and embedded systems.

SNO	Module Name	Content	Credits	Duration (in weeks)	Platform	Theory/Lab
1	Python Programming	Numpy for vector/matrix operations	2	1	Termux	Theory +Lab
2	C Programming	Using pointer arrays for vector/matrix operations	2	1	Termux	Lab
3	Data Structures	Pointers and lists for vector/matrix operations	4	2	Termux	Lab

4	Math Computing on hardware	78 Inter chip communication on the Vaman board for vector/matrix operations	4	4	Vaman	Lab
	Total		12	8		

5G Module 3 Course Content (Including mini project)

4 Months only (8 credits).

Tittle : Communication Engineering and Standards Hands on training on 5G/ 6G Projects

TRAINING SCHEDULE FOR C & DS LANGUAGE : 1 Month 4 weeks

SN	Module	Output Expected	Time
1	Linux commands	Practice & Demonstrate usage of a few commands	2 Day (2Hours)
2	FLOWCHARTS	PPT on Types of flowcharts and flowchart symbols	1 Day (1 Hour)
3	Draw.io	Demo – How to draw block diagram	1 Day(1 Hour)
4	Git Lab	Demo - Use Github to store your all programs everyday	1 Day(1 Hour)
5	C- Functions	Practice Programs & show your programs	1 Day(1 Hour)
6	C - BIT FIELDS	Practice Programs & show your programs	1 Day(1hour)
7	Data Structures	Practice Programs & show your programs	3 Days(4 hours)
9	Pointers	Practice Programs & show your programs	3 Days(4hour)
8	Bitwise Operators	Practice Programs & show your programs	1 Day(1hour)
10	Linked List	Practice Programs & show your programs	1 Day(1hour)
11	Socket programming	Programs with Wireshark Capture	6 Days(6hours)
12	Threads	Programs with Wireshark Capture	3 Days(3hours)
13	Tools Usage	Wireshark Usage, gcc usage, gdb usage	2 Day(2hours)
TOTAL DAYS			26 Days (26 hours)

NOTE:

- **C - BIT FIELDS**

1. Design Data structures for IPv4, IPv6

- **Data Structures and pointers**

1. Implement a stack using a singly linked list, insert only even numbers.
2. Implement a Queue using a singly linked list, insert only odd numbers.
3. Write nested structure programs.
4. 4 Data Structures Lab Exercises

- **Bitwise Operators**

1. Swapping two bits of a byte using C program

"Input number: 0x0A (Hexadecimal)
Binary of input number: 0000 1010
After swapping of bit 1 and 2
Binary will be: 0000 1100
Output number will be: 0x0C (Hexadecimal)"

1. C program to Count the Number of Trailing Zeroes in an Integer

Socket programming

1 . Write a UDP send and receive programs,

- send 64 Bytes of data at a time.
- Print the payload sending side.
- Print the payload "Receive side".

2. Write a TCP send and receive programs,

- send 64 Bytes of data at a time.
- Print the payload sending side.
- Print the payload "Receive side".

Threads

1. Write a c program1

Lunch two Threads one thread should wait for udp messages and other thread should send " Message from program 1 " to program 2

2. Write a c program 2

- Lunch two Threads one thread should wait for udp messages and other thread should send " Message from program 2 " to program 1

Tools Usage

1. Filter TCP, UDP Packets using Wireshark.

2.Write a program and use GDB to debug the program.

3.Write a program and use valgrind to check Memory leaks.

5G Module 3 Course Content (Including mini project)

covering theoretical foundations in the 2 month

hands-on mini projects in the remaining 1 months.

This structure provides students with a strong understanding of 5G network components, protocols, and functionalities before applying their knowledge through practical projects.

will suggest 3rd-party tools and mini project ideas to reinforce learning.

Training Schedule: 2 Months (48 Days, 5 Days per Week)

Week	Day 1	Day 2	Day 3	Day 4	Day 5
Months					
Week 1	Introduction to 5G Technology and Applications	5G Protocol Stack: Layers and Functions	Key Features of 5G: Speed, Latency, and Reliability	5G Channels: Logical, Transport, and Physical	Overview of Layer 2 and Layer 3: What Do They Do?
Week 2	Introduction to MAC: Why It's Important	Multiplexing and Demultiplexing	Logical Channels: How Data Is Organized	TB formation,	Real-World Examples: How MAC Is Used in 5G
Week 3	Introduction to RLC: Managing Data Flow	RLC Modes: TM, UM, and AM	Error Handling in RLC: Retransmissions	RLC in Action: Examples and Use Cases	How RLC Works with MAC and PDCP
Week 4	Introduction to PDCP:	How PDCP	Why PDCP Duplication Improves		

	Secure and Reliable Data	**Compresses and Encrypts Data**	**Performanc e**	**PDCP in Real Life: Use Cases**	**Wrap-Up: Recap of Layer 2 Concepts**
Wee k 5	**Introduction to SDAP: Managing QoS in 5G**	**QoS Flows: How Priorities Are Set**	**SDAP in Simple Terms: Examples and Use Cases**	**Recap: How Layer 2 Components Work Together**	**Q&A and Layer 2 Troubleshooting Tips**
	Revised Test				
Wee k 6	**Introduction to RRC: Connectin g Devices**	**RRC States: How Devices Stay Connected**	**Handover: How 5G Maintains Connections**	**RRC Security: Keeping Data Safe**	**RRC in Real Life: Examples and Use Cases**
Wee k 7	**NAS Overview: Managing Mobility and Sessions**	**Registration and Authentication in NAS**	**NAS Security: Ensuring Safe Communicatio n**	**How NAS Talks to the Core Network**	**NAS in Real Life: Examples and Use Cases**

	Revised Test				

Month 3		***Mini project***	***Mini project***	***Mini project***	***Mini project***

5G Module 3 Course Content (Including mini project)

1.1 Course Goals

1. Develop a comprehensive understanding of the **5G ecosystem** (UE, gNB, CN).

2. Understand **5G protocol layers** (Layer 1, Layer 2, and Layer 3) for UE, gNB and Netfork functions of CN.

3. Explain the role of **key technologies**: Carrier Aggregation (CA), Dual Connectivity (DC), and Non-Standalone Architecture (NSA).

4. Gain hands-on experience with 5G simulations and tools through **mini projects**.

1.2 Month 2: Theory and Fundamentals (Lectures, Discussions, and Lab Demos)

1.2.1 Week 1 & 2 : Introduction to 5G and Architecture Overview

- **Topics:**
    - **a.** Evolution of mobile networks (2G to 5G)

b. Overview of 5G capabilities (high-speed data, low latency, IoT support)
c. 5G architecture: UE, gNB, and Core Network (CN)

- **Outcomes:**
  a. Students understand the broad view of 5G architecture and its impact on network design.

1.2.2 Week 3&4: UE Functionalities and Layer-wise Protocol Stack

- **Topics:**
  - Layer 3 (NAS/RRC): Mobility, session, and connection management.
  - Layer 2 (MAC, RLC, PDCP): Data link and flow control, segmentation, error correction.
  - Layer 1 (Physical): Modulation, coding, and signal transmission.
- **Outcomes:** Students understand UE's layered structure and functionalities, focusing on how data flows and is managed.

1.2.3 Week 5 & 6: gNB (Next-generation Node B) Functionalities and 5G Interfaces

- **Topics:**
  - gNB roles: Handling radio resources, communicating with UE and CN.
  - 5G interfaces: Uu, NG, Xn, N1-N3, and signaling paths.
  - Functional splits (CU-DU separation).
- **Outcomes:** Students understand the gNB's role in interfacing with the UE and CN and the details of 5G interfaces.

1.2.4 Week 7&8: Core Network (CN) Functions and Advanced Concepts

- **Topics:**
  - CN components: AMF (Access and Mobility), UPF (User Plane), SMF (Session Management), and others.
  - Network slicing and Service-Based Architecture (SBA) in 5G.
  - Advanced Concepts: Carrier Aggregation (CA), Dual Connectivity (DC), Non-Standalone Architecture (NSA).
- **Outcomes:** Students gain knowledge of CN functionalities and advanced technologies in 5G.

1.3 Mini Projects and Practical Assignments (Hands-on Labs and Group Projects)

1.3.1 Project Preparation and Tool Familiarization (4 weeks)

- **Overview of Tools:** Introduce students to 3rd-party tools, such as:
  - **Simu5G** or **ns-3** (for simulating network layers and protocols)

    - **OpenAirInterface** (OAI) for practical understanding of gNB and core network functions
    - **srsRAN** (software suite for 4G/5G) to explore functionalities in a realistic network setup
- **Lab Setup and Practice:** Set up tools, conduct sample simulations, and perform basic network testing.
- **Expected Outcome:** Students become comfortable using tools and simulators for their projects.

1.3.2 Mini Projects (4 weeks) :

`

- **Project 1: UE Layer-wise Functionality Testing**
    - **Objective:** Simulate and test UE layers' functionalities (RLC, PDCP, Physical layer) in a controlled environment.
    - **Expected Outcome:** Students develop a thorough understanding of data flow and error handling in UE.
    - **Suggested Tool:** ns-3 or Simu5G for layer simulation and testing.

- **Project 2: Analyzing Carrier Aggregation (CA) and Dual Connectivity (DC)**
    - **Objective:** Set up and test scenarios with CA and DC to see their impact on data throughput and latency.
    - **Expected Outcome:** Students understand the role of CA and DC in enhancing bandwidth and reducing latency.
    - **Suggested Tool:** Simu5G or OAI for running aggregation and connectivity experiments.

- **Project 3: gNB and Core Network Integration**
    - **Objective** Set up a virtual gNB and core network (OAI or srsRAN) and simulate UE connectivity.
    - **Expected Outcome:** Students see how the gNB interacts with the core network, covering session and mobility management.
    - **Suggested Tool:** OpenAirInterface or srsRAN to simulate real-time gNB and CN communication.

- **Project 4: 5G Interface Testing and Performance Analysis**
    - **Objective:** Simulate communication across 5G interfaces (Uu, NG, Xn) and analyze performance metrics like latency and jitter.
    - **Expected Outcome:** Students gain practical experience with interface data flow and analyze performance improvements.
    - **Suggested Tool:** Simu5G or ns-3 for interface simulation and data analysis.

1.3.3 Final Project Presentation and Report Submission (1 week)

- **Presentation:** Each team presents their project, showcasing the key functionalities they tested, performance data collected, and learning outcomes.
- **Report:** Detailed report summarizing project goals, methodologies, results, and insights.

1.4 Expected Learning Outcomes

1. **Theory:** Gain foundational knowledge of 5G technology, including components and interfaces.

2. **Application:** Apply theoretical knowledge to real-world scenarios through 5G simulations.

3. **Analysis:** Develop skills in analyzing 5G network performance, utilizing CA, DC, and NSA.