Weekly Progress Report

Project: TinyML-Based Project on FPGA Board with RISC-V Core

Name: Aman Chauhan
Branch: Btech. Computer Science and Engineering (CSE Core)
Roll Number: 22BCE0476
College: Vellore Institute of Technology, Vellore
Mentor: Dr. Sudip Roy

Week: *2nd report*
Date: *30th May 2025*

Agenda

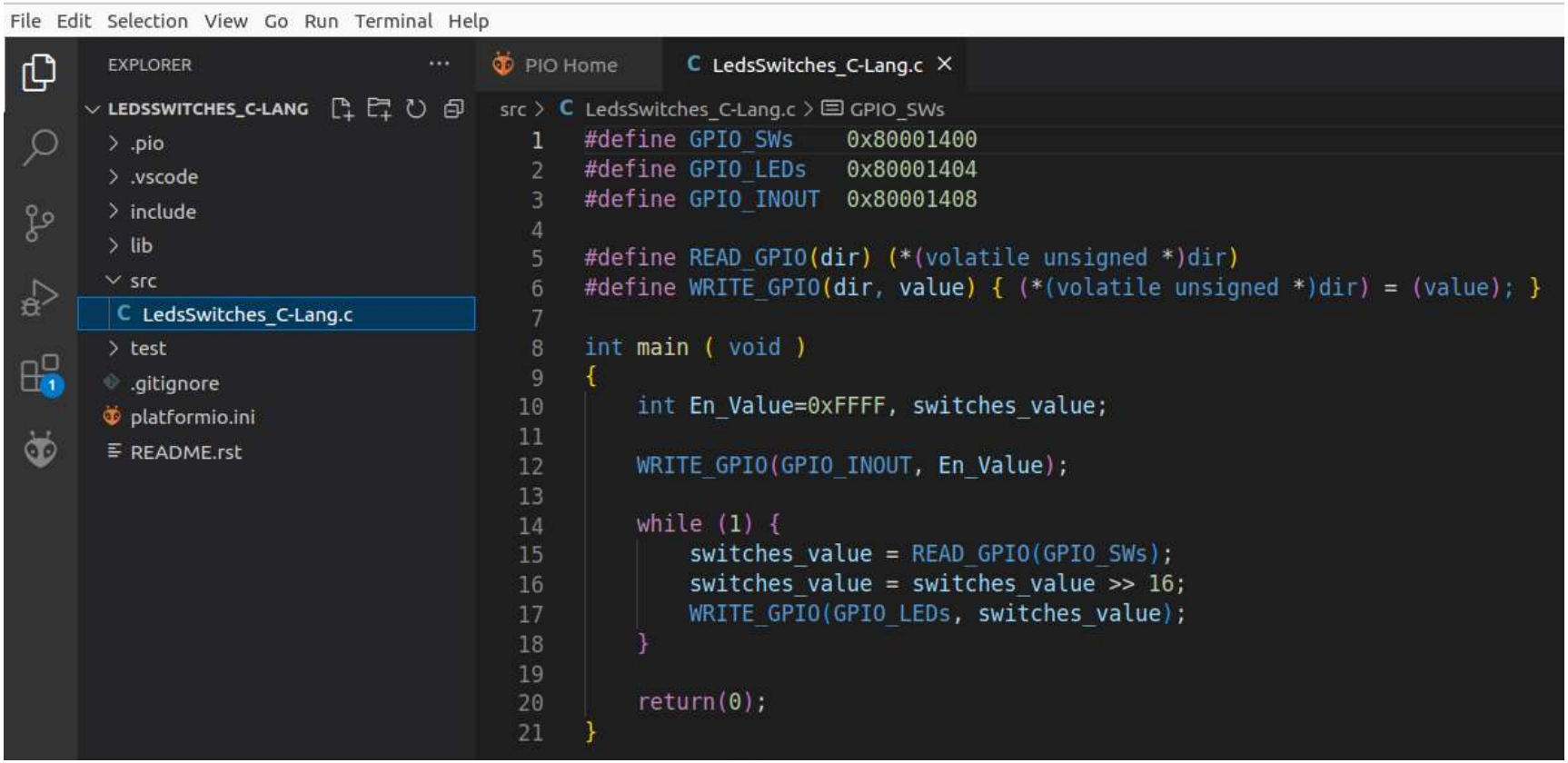
- RVfpga SoC Simulation and Development
- TinyML Basics and Model Training
- Applied AI Lab: Deep Learning for Computer Vision.
- Progress Summary Tracker
- References

RVfpga Overview

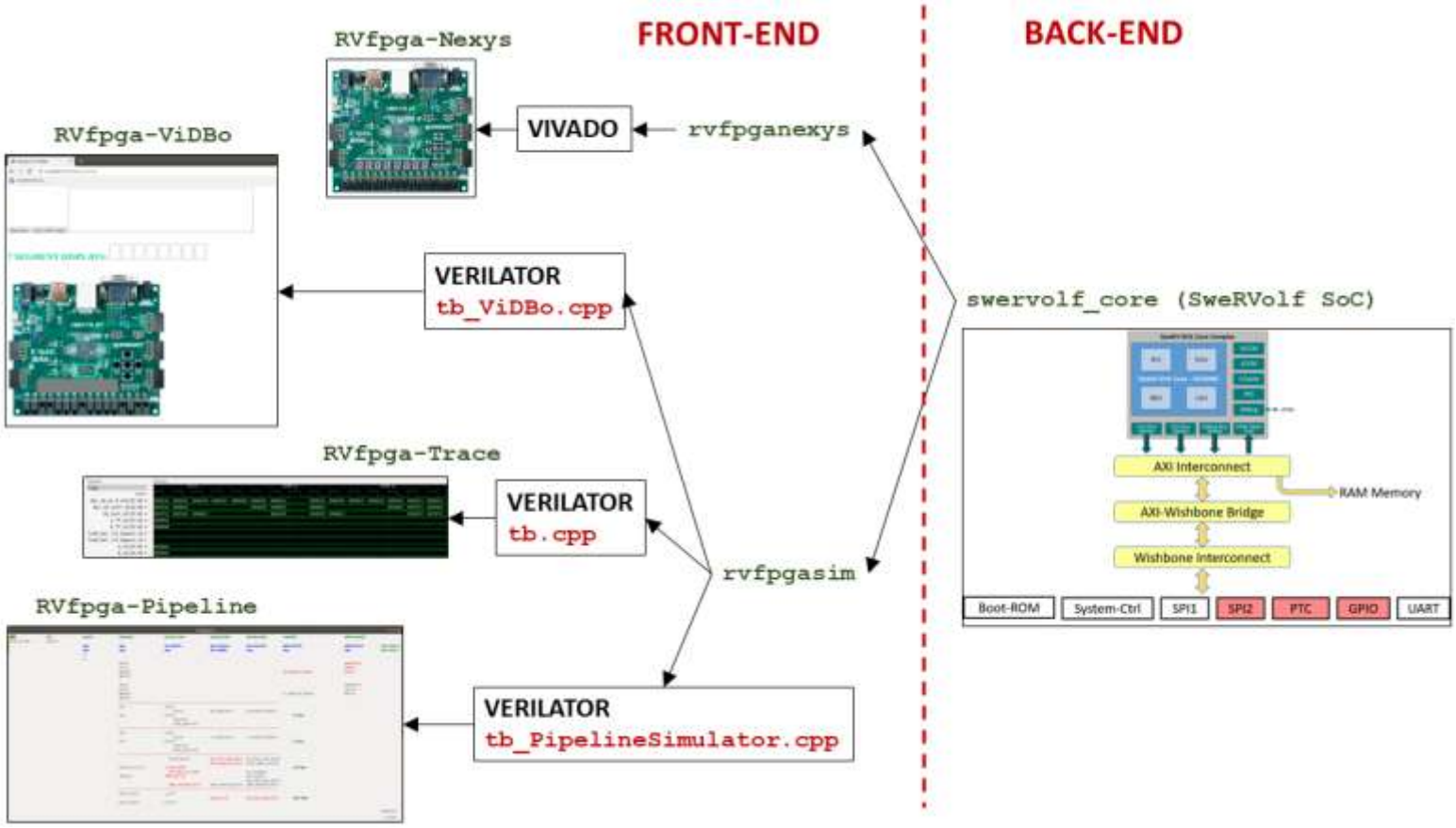
- RVfpga: RISC-V FPGA platform with **SweRVolf SoC** on **Nexys A7 virtual board.**
- Simulators: **ViDBo** (main), **Pipeline**, **Trace**, **Whisper** using **Verilator.**
- Virtual board mimics real hardware I/O (LEDs, switches, UART, displays)
- Course taken on **HarvardX edX** platform

RVfpga Simulation Steps (Platform IO)

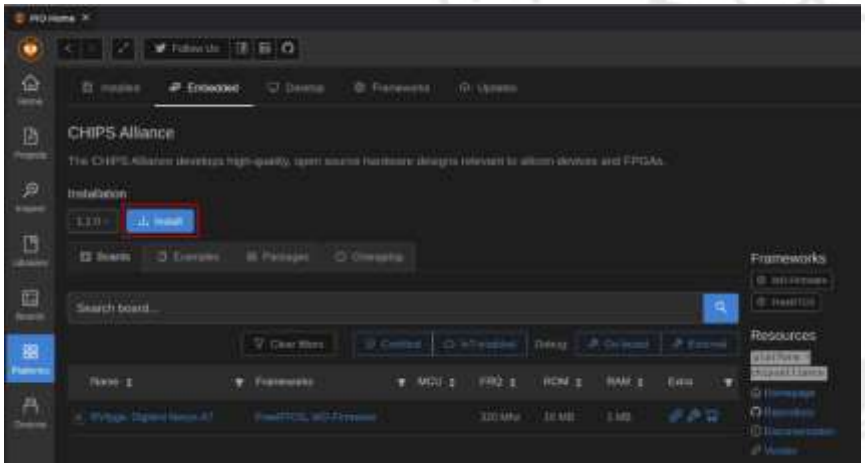
- Setup **RVfpga simulation** environment on **PlatformIO.**
- Built and ran Verilator-based simulators targeting **SweRVolf SoC**
- Used **rvfpgasim.v top module** and **tb_ViDBo.cpp** for I/O bridging
- Interacted with **virtual Nexys A7 board to test SoC outputs and inputs.**
- Explored C programming on simulated SoC and **started assembly and I/O chapters**



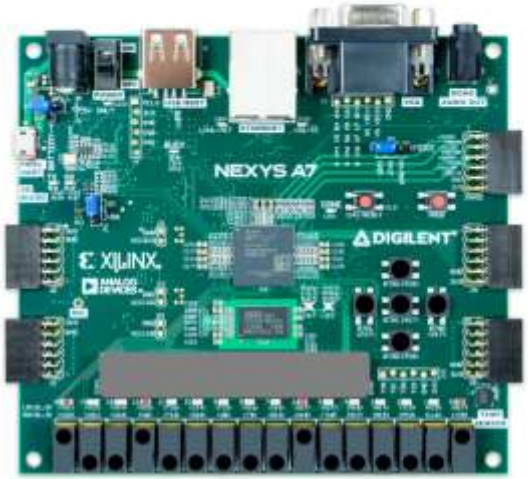
1.LedsSwitches_C-Lang.c Program


RVfpga-Nexys
FRONT-END
BACK-END
VIVADO
rvfpganexys
RVfpga-ViDBo
VERILATOR
tb_ViDBo.cpp
swervolf_core (SweRVolf SoC)
RVfpga-Trace
VERILATOR
tb.cpp
rvfpgasim
AXI Interconnect
RAM Memory
AXI-Wishbone Bridge
Wishbone Interconnect
Boot-ROM
System-Ctrl
SPI1
SPI2
PTC
GPIO
UART
RVfpga-Pipeline
VERILATOR
tb_PipelineSimulator.cpp

Simulation



7 SEGMENT DISPLAYS:



Progress Tracker

Completed:

- RVfpga Setup and ViDBo Simulation
- C Programming Basics on RVfpga SoC
- RISC-V Assembly Programming
- RISC-V Function Calls

Partial Progress on:

- Mixing C and Assembly Functions
- Introduction to Peripherals and Input/Output
- More I/O: 7-Segment Displays
- More I/O: Timers
- Interrupts

TinyML Basics

- **TinyML**: Running machine learning on low-power edge devices and microcontrollers
- **Covered fundamentals**: ML workflow, deep learning blocks, ML scenarios on mobile/IoT
- Explored keyword spotting, visual wake words, anomaly detection, responsible AI
- **Course content** from TinyML community and TensorFlow Lite Micro(**Edx Harvard**)

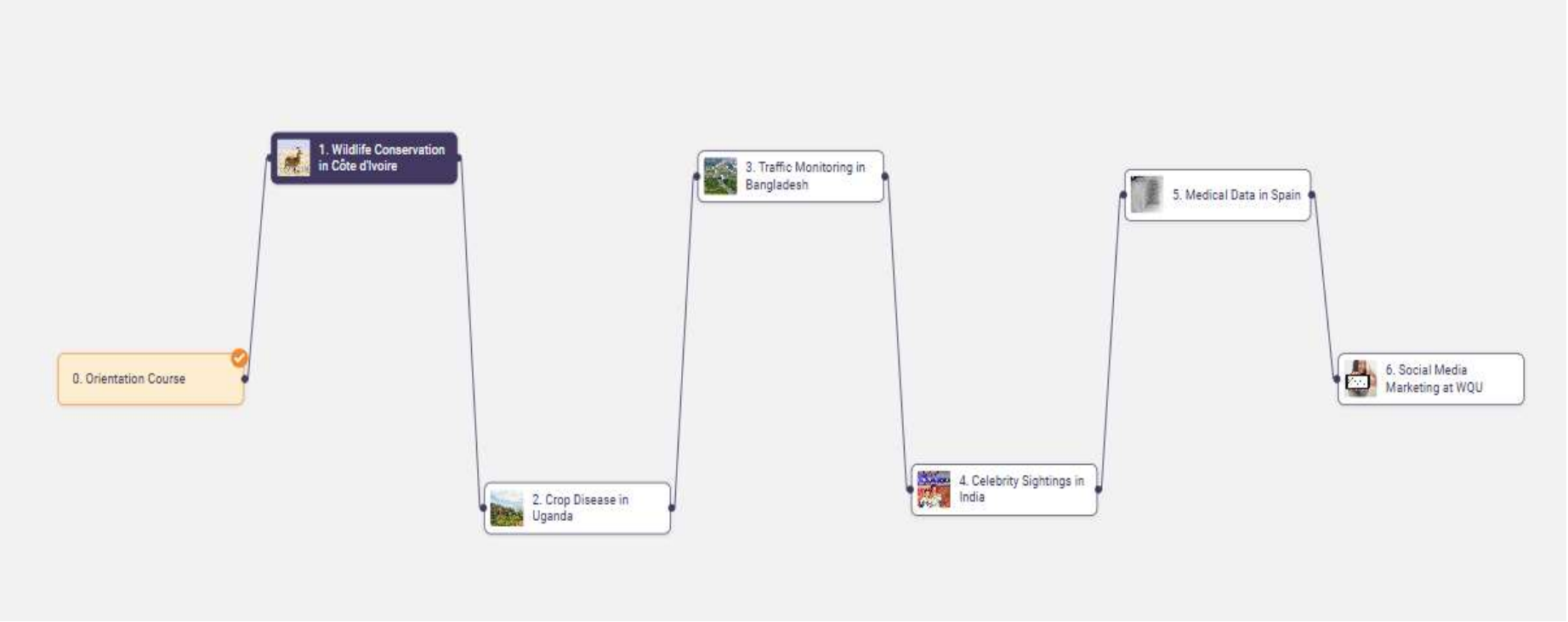
TinyML Projects and Training

- Worked on **Arduino-based TinyML projects** integrating **sensor data and ML models.**
- Trained **TinyML models** for real-time keyword spotting and **anomaly detection.**
- Learned **data preparation** and **model optimization** for **resource-constrained devices.**

WorldQuant University Lab Overview

- **Applied AI Lab**: Deep Learning for Computer Vision
- Hands-on practice with **CNN architectures**, **data augmentation**, and **classification tasks.**
- Bridging theory with practical AI implementation skills.
- Focus on **enhancing model accuracy and robustness.**

Map of Course



Summary

Task	Status	Comments	Next Steps
RVfpga Course (HarvardX edX)	Partially Completed	Setup, simulation, basic C programming done	Complete assembly and peripherals chapters
TinyML Course	Partially Completed	Covered basics and Arduino projects	Expand to complex TinyML models
WorldQuant Applied AI Lab	Ongoing	CNNs and image classification in progress	Complete lab and project submissions

Challenges Faced

- **No physical Nexys A7 FPGA board** available for real-time hardware deployment and testing
- **Complex Verilator simulation** environment requires careful setup and debugging
- **TinyML model constraints** due to limited memory and compute on microcontrollers
- **Vivado toolchain requirements** (high RAM, storage, and compute) exceed current PC capabilities.

References

- RVfpga HarvardX edX course materials and RVfpga source code
- TinyML foundational courses and TensorFlow Lite Micro documentation
- WorldQuant University Applied AI Lab content
- RVfpga simulators documentation and virtual board resources
- Report 1 and internal study materials