Comprehensive Deployment Roadmap: Multi-Modal Gunshot Detection on FPGA

This document outlines a structured, 6-week learning and implementation plan for deploying the CNN14 + BiLSTM + Attention gunshot detection (GSD) model onto a Field-Programmable Gate Array (FPGA), integrated with a 6-axis microphone array and a thermal camera for multi-modal localization.

The roadmap is divided into three major phases: Foundational Preparation, Core ML/DSP Acceleration, and System Integration, ensuring a systematic build-up of knowledge and hardware functionality.

Phase I: Foundational Preparation (Weeks 1–2)

This phase establishes proficiency in Vitis High-Level Synthesis (HLS), the core tool for custom hardware development, and initiates the preparation of the CNN component using the Vitis AI framework.

Week 1: Vitis HLS Fundamentals and CNN Preparation

The primary focus is mastering the HLS workflow, optimization pragmas, and preparing the CNN14 feature extractor for the Deep Learning Processor Unit (DPU).

Day	Topic Focus	Key Learning Objectives	Free Resource / Tutorial

Day 1	**Vitis HLS Introduction**	Understand the HLS flow (C/C++ to RTL), C-simulation, and C-synthesis.[1] Learn when and why to use HLS for hardware acceleration.[4]	BYU Computing Bootcamp Vitis HLS [3], Webinar: HLS - What Is It and When Do You Use It? [4], Vitis HLS Tutorials (Getting Started).[5]
Day 2	**HLS Data Types & Pragma Basics**	Implement **fixed-point arithmetic** (ap_fixed) for efficiency, essential for all ML and DSP kernels.[6] Learn and apply primary optimization pragmas (PIPELINE, UNROLL).	Vitis HLS User Guide Tutorials: Data Types, Loops Primer.[6]
Day 3	**Array Management in HLS**	Map software arrays to hardware Block RAMs (BRAMs). Implement **Array Partitioning** (complete, block) for parallel weight access, critical for BiLSTM.[6]	Vitis HLS Introductory Examples: Array Partitioning.[7]
Day 4	**Vitis AI Framework & DPU**	Overview of the Vitis AI framework, DPU architecture, and the flow for deploying neural networks (CNN14).[8]	Introduction to Vitis AI (Video) [9], Vitis AI Playlist.[8]
Day 5	**Vitis AI Model Quantization**	Start the process of **quantizing** the CNN14 model to	Vitis AI Playlist: Quantization/Deep Dive.[8]

		the fixed-point format required by the DPU.[11] Understand how Vitis AI Optimizer prepares the model for compilation.[11]	

Week 2: Acoustic Signal Chain (PDM Demodulation)

This phase focuses on the first custom hardware kernel: processing the raw audio data from the 6-axis microphone array using highly parallelized PDM demodulation.

Day	Topic Focus	Key Learning Objectives	Free Resource / Tutorial
Day 6	**PDM Demodulation Theory**	Understand Pulse Density Modulation (PDM) and the necessity of the Cascaded Integrator-Comb (CIC) filter chain for conversion to PCM audio.[12]	Research on CIC Decimation Filter Architecture.[14]
Day 7	**HLS CIC Filter Implementation**	Implement a single-channel CIC filter C++ function using **fixed-point data types**. Use the #pragma HLS PIPELINE directive for high throughput.[16]	HLS Matched Filter/Decimation Filter Tutorials.[16]
Day 8	**Parallel Dataflow**	Implement the	Vitis HLS User

		architecture for **six parallel CIC filters** concurrently using the #pragma HLS DATAFLOW pragma. This is essential for phase coherence in the 6-axis array.[6]	Guide Tutorials: Dataflow Paradigm.[6]
Day 9	**C/RTL Co-simulation**	Verify the CIC filter bank's functional correctness using C-simulation and then confirm the generated RTL's timing and behavior using Co-simulation.[18]	Vitis HLS User Guide Tutorials (RTL Co-simulation).[6]
Day 10	**IP Packaging & AXI Interfaces**	Package the CIC filter bank as a reusable IP core for Vivado.[19] Define the I/O using the **AXI4-Stream** protocol for high-speed data flow and **AXI4-Lite** for control.[20]	Vitis HLS IP Integration Tutorial.[20]

Phase II: Core ML and DSP Acceleration (Weeks 3–4)

This phase accelerates the complex ML classification logic (BiLSTM-Attention) and the core DSP localization engine (DOA).

Week 3: BiLSTM-Attention Layer Acceleration

Optimize the recurrent neural network component using custom HLS techniques to minimize the inherent latency of sequential models.

Day	Topic Focus	Key Learning Objectives	Free Resource / Tutorial
Day 11	**BiLSTM Structure in C++**	Review the mathematical structure of the LSTM cell (gates, recurrence). Begin writing the unoptimized BiLSTM C++ implementation.[22]	Open-source FPGA-based LSTM HLS C++ Example.[22]
Day 12	**BiLSTM Optimization: Pipelining**	Apply aggressive #pragma HLS PIPELINE II=1 to the loops inside the LSTM gate computations to maximize parallelism within each sequential time step.[6]	Vitis HLS User Guide: Pipelining Loops.[6]
Day 13	**BiLSTM Optimization: Memory**	Apply Array Partitioning directives to the BiLSTM's weight and bias matrices (e.g., W_{ih}, W_{hh}) to enable parallel reading from BRAMs, supporting pipelined computations.[6]	Vitis HLS User Guide: Array Partitioning.[6]

Day 14	**Attention Mechanism: Matrix Math**	Implement the core matrix-vector multiplications required for the attention mechanism's scoring function.[24] Optimize these operations using loop unrolling and pipelining for GEMM acceleration.[25]	HLS Matrix Multiplication/GEMM tutorials.[25]
Day 15	**Attention Mechanism: Softmax/Output**	Implement the **Softmax** function using optimized fixed-point methods (e.g., CORDIC/LUTs).[27] Package the combined BiLSTM-Attention as an HLS IP core.[20]	Vitis HLS Tutorials (Beamformer Analysis for DSP techniques).[24]

Week 4: Spatial Processing and DPU Compilation

Develop the Direction of Arrival (DOA) kernel and finalize the preparation of the CNN component for the target FPGA.

Day	Topic Focus	Key Learning Objectives	Free Resource / Tutorial
Day 16	**DOA Algorithm Selection & Theory**	Select a suitable real-time algorithm like Beamscan or GCC-PHAT.[28]	Research: Beamscan and GCC-PHAT algorithms.[28]

		Understand the core DSP requirements (FFT, matrix operations).[30]	
Day 17	**DOA HLS: FFT Implementation**	Implement a fixed-point Fast Fourier Transform (FFT) kernel in HLS, a core component of both DOA algorithms. Focus on achieving optimal pipeline Initiation Interval (II).[6]	Vitis HLS Introductory Examples: FFT.[7]
Day 18	**DOA HLS: Matrix Operations**	Implement the high-level matrix/vector mathematics (e.g., spatial covariance estimation) required by the DOA algorithm, applying optimization techniques from the Attention layer.[25]	HLS Matrix Multiplication Tutorials.[25]
Day 19	**Vitis AI Compilation (VAI_C)**	Use the Vitis AI Compiler (VAI_C) to compile the quantized CNN14 model into the DPU executable binary (XMODEL) for the target platform.[32]	Vitis AI Tutorials: DPU Integration.[32]
Day 20	**Final IP Core**	Perform final	Vitis HLS IP

	Generation	synthesis and IP packaging for the DOA Engine HLS core. Ensure all custom IP cores (CIC, BiLSTM, DOA) have robust AXI4-Stream data and AXI4-Lite control interfaces.[20]	Packaging Review.[20]

Phase III: System Integration and Finalization (Weeks 5–6)

The final phase brings all the accelerated kernels together into a single hardware design, followed by software development for control, and multi-modal sensor fusion.

Week 5: Vivado System Integration

Assemble all hardware components in the Vivado Block Design and generate the final hardware file (bitstream).

Day	Topic Focus	Key Learning Objectives	Free Resource / Tutorial
Day 21	**Vivado Project Setup**	Create the Vivado project, instantiate the target Zynq Processing System (PS), and configure the memory systems.	Implementing a Vitis HLS RTL IP in Xilinx Vivado.[21]

Day 22	**PL Integration: DPU and Custom Cores**	Integrate the Vitis AI DPU IP core and the custom HLS IP cores (CIC, BiLSTM, DOA) into the Programmable Logic (PL).[32]	Integrate DPU-TRD PL Acceleration Kernel.[32]
Day 23	**AXI Interconnection**	Connect the data path using **AXI4-Stream** (e.g., CIC $\rightarrow$ DOA $\rightarrow$ BiLSTM/DPU). Use the Vivado **Run Connection Automation** tool to link AXI4-Lite control to the PS.[20]	Vitis HLS IP Integration Tutorial (Connecting AXI Interfaces).[20]
Day 24	**Thermal Camera Interface**	Implement the simpler **SPI/I2C** controller logic in the PL for the thermal camera, minimizing interface complexity to focus on core audio processing.[33]	I2C and SPI communication on FPGA example.[33]
Day 25	**Hardware Bitstream Generation**	Run the full synthesis, place, and route process in Vivado to generate the final hardware bitstream (.bit file) and the XSA hardware definition file. Understand the HLS	Vivado Implementation (Vitis HLS Documentation).[34]

		implementation report for resource utilization and timing.[34]	

Week 6: Software, Verification, and Fusion

Develop the host application on the Processing System (PS) to manage the accelerators and perform the multi-modal fusion logic.

Day	Topic Focus	Key Learning Objectives	Free Resource / Tutorial
Day 26	**PS Software Development Setup**	Set up the Vitis Software Platform project using the generated XSA file.[35] Access the auto-generated driver files for the custom AXI-Lite IP cores.[20]	Vitis Tutorials: Getting Started.[35]
Day 27	**Hardware Control Software**	Write the C/C++ application on the PS to control the flow: reading sensor data and using AXI4-Lite control registers to start and manage the DPU and HLS kernels.[20]	Vitis HLS IP Integration Tutorial (Creating a Software Application).[20]
Day 28	**Multi-Modal Fusion Logic**	Implement the sensor fusion algorithm in the PS	Custom C/C++ application development using

		software. Correlate the acoustic classification (BiLSTM output) with the DOA localization estimate and the thermal camera confirmation to confirm the source location.[6]	PS resources.[6]
Day 29	**Real-Time Simulation & Debug**	Load the final application onto the target FPGA/SoC. Debug the data flow and verify the low-latency inference performance (target: 100-200ms per segment) using the timer code.[20]	Measuring the run time on the hardware.[20]
Day 30	**Final Testing and Documentation**	Conduct comprehensive end-to-end testing with audio and thermal inputs. Document the system performance, resource utilization, and lessons learned.	Project Finalization and Reporting.

Works cited

1. Vitis High-Level Synthesis User Guide - AMD, accessed October 25, 2025, https://www.xilinx.com/support/documents/sw_manuals/xilinx2022_2/ug1399-vitis-hls.pdf

2. Vivado Design Suite User Guide: High-Level Synthesis, accessed October 25, 2025, https://www.xilinx.com/support/documents/sw_manuals/xilinx2020_1/ug902-vivado-high-level-synthesis.pdf
3. Vitis: HLS • BYU Computing Bootcamp, accessed October 25, 2025, https://byu-cpe.github.io/ComputingBootCamp/tutorials/vitis_hls/
4. Webinar: HLS - What Is It and When Do You Use It? - YouTube, accessed October 25, 2025, https://www.youtube.com/watch?v=RLCpw7RyhZM
5. Vitis HLS — Vitis™ Tutorials 2021.1 documentation - GitHub Pages, accessed October 25, 2025, https://xilinx.github.io/Vitis-Tutorials/2021-1/build/html/docs/Getting_Started/Vitis_HLS/Getting_Started_Vitis_HLS.html
6. Tutorials and Examples - 2025.1 English - UG1399, accessed October 25, 2025, https://docs.amd.com/r/en-US/ug1399-vitis-hls/Tutorials-and-Examples
7. Xilinx/Vitis-HLS-Introductory-Examples - GitHub, accessed October 25, 2025, https://github.com/Xilinx/Vitis-HLS-Introductory-Examples
8. Vitis AI - YouTube, accessed October 25, 2025, https://www.youtube.com/playlist?list=PLx15eYqzJifffUiCaTxzFFwJSKT7eKnfq
9. Introduction to Vitis AI - YouTube, accessed October 25, 2025, https://www.youtube.com/watch?v=Z6jkUz0Re7Q
10. Xilinx/Vitis-AI-Tutorials - GitHub, accessed October 25, 2025, https://github.com/Xilinx/Vitis-AI-Tutorials
11. [2401.17544] Trainable Fixed-Point Quantization for Deep Learning Acceleration on FPGAs, accessed October 25, 2025, https://arxiv.org/abs/2401.17544
12. HIGH-ACCURACY ACOUSTIC SENSING SYSTEM WITH A 2D TRANSCEIVER ARRAY: AN FPGA-BASED DESIGN, accessed October 25, 2025, https://hammer.purdue.edu/ndownloader/files/44937940
13. M3-AC: A Multi-Mode Multithread SoC FPGA Based Acoustic Camera - MDPI, accessed October 25, 2025, https://www.mdpi.com/2079-9292/10/3/317
14. Design and Implementation of a Decimation Filter For High Performance Audio Applications, accessed October 25, 2025, https://www.researchgate.net/publication/224311738_Design_and_Implementation_of_a_Decimation_Filter_For_High_Performance_Audio_Applications
15. Master theses CIC filter design with HLS - Webthesis - Politecnico di Torino, accessed October 25, 2025, https://webthesis.biblio.polito.it/11004/1/tesi.pdf
16. Matched Filter: Vitis HLS Implementation - YouTube, accessed October 25, 2025, https://www.youtube.com/watch?v=hvtVcxLI-xE
17. HLS Programmers Guide - 2025.1 English - UG1399, accessed October 25, 2025, https://docs.amd.com/r/en-US/ug1399-vitis-hls/HLS-Programmers-Guide
18. Introduction to High-Level Synthesis with Vivado HLS, accessed October 25, 2025, https://users.ece.utexas.edu/~gerstl/ee382v_f14/soc/vivado_hls/VivadoHLS_Overview.pdf
19. AMD Vitis™ HLS, accessed October 25, 2025, https://www.amd.com/en/products/software/adaptive-socs-and-fpgas/vitis/vitis-

hls.html
20. Vitis HLS Integration Tutorial • ECEn 625 - GitHub Pages, accessed October 25, 2025, https://byu-cpe.github.io/ecen625/hls-integration-tutorial/
21. From Xilinx Vitis HLS to FPGA IP - YouTube, accessed October 25, 2025, https://www.youtube.com/watch?v=bHig4zQpq2o
22. ntampouratzis/FPGA-based-LSTM - GitHub, accessed October 25, 2025, https://github.com/ntampouratzis/FPGA-based-LSTM
23. The community version of HLS_BLSTM (A BLSTM FPGA accelerator of an OCR appilcation, using CAPI/SNAP) - GitHub, accessed October 25, 2025, https://github.com/oprecomp/HLS_BLSTM
24. These tutorials offer a broader introduction to the Vitis Unified IDE, in addition to briefly describing the most simple HLS flows and use cases. - XD261, accessed October 25, 2025, https://docs.amd.com/r/en-US/Vitis-Tutorials-Vitis-HLS
25. spcl/gemm_hls: Scalable systolic array-based matrix-matrix multiplication implemented in Vivado HLS for Xilinx FPGAs. - GitHub, accessed October 25, 2025, https://github.com/spcl/gemm_hls
26. Matrix-vector multiplication - SLING user documentation, accessed October 25, 2025, https://doc.sling.si/en/workshops/hls-for-fpga/04-programming/02-matv_mult/
27. HLS FPGA Design Guide - Siemens HLS Academy, accessed October 25, 2025, https://hls.academy/topics/hls-fpga/
28. Beamscan Direction of Arrival Estimation Using FPGA - MATLAB & Simulink - MathWorks, accessed October 25, 2025, https://www.mathworks.com/help/phased/ug/direction-of-arrival-beamformer.html
29. Beamforming & DOA | PySDR: A Guide to SDR and DSP using Python, accessed October 25, 2025, https://pysdr.org/content/doa.html
30. Digital Beamforming Implementation on an FPGA Platform - UPCommons, accessed October 25, 2025, https://upcommons.upc.edu/bitstreams/df9e1453-da34-4d71-a712-0d0d444cb25c/download
31. FPGA-Based Architectures for Acoustic Beamforming with Microphone Arrays: Trends, Challenges and Research Opportunities - MDPI, accessed October 25, 2025, https://www.mdpi.com/2073-431X/7/3/41
32. Vitis AI Tutorial – Part 4 - Octavo Systems, accessed October 25, 2025, https://octavosystems.com/app_notes/vitis-ai-tutorial-part-4/
33. I2C and SPI communication on FPGA, accessed October 25, 2025, https://forums.ni.com/t5/Example-Code/I2C-and-SPI-communication-on-FPGA/ta-p/3823804
34. Running Implementation - 2025.1 English - UG1399, accessed October 25, 2025, https://docs.amd.com/r/en-US/ug1399-vitis-hls/Running-Implementation
35. Xilinx/Vitis-Tutorials: Vitis In-Depth Tutorials - GitHub, accessed October 25, 2025, https://github.com/Xilinx/Vitis-Tutorials