

## TECHNICAL EXPERIENCE DETAILS

### DV Project

Title	Intent	Outcome and Learnings
Synchronous FIFO	Implement FIFO design and UVM based test bench to validate the design	<ul style="list-style-type: none"><li>Using SV learned to implement below<ol style="list-style-type: none"><li>FIFO design</li><li>UVM test bench components like Scoreboard, Monitor, Driver, Sequencer, Environment, Agent</li></ol></li><li>Github link: <a href="#">nandu161096/FIFO UVM TB AND DESIGN</a></li></ul>
8-bit cipher (Encryption & Decryption)	Implement 8-bit cipher design and UVM based test bench to validate the design	<ul style="list-style-type: none"><li>Using SV learned to implement below<ol style="list-style-type: none"><li>Pseudo random generator (PRNG)</li><li>Encryption &amp; decryption</li><li>UVM test bench components like Scoreboard, Monitor, Driver, Sequencer, Environment, Agent</li></ol></li><li>Github link: <a href="#">nandu161096/8_BIT_CIPHER</a></li></ul>

### WLAN FIRMWARE - VI team | April 2024 – Present (Qualcomm Chennai)

General roles and responsibilities	Details of Feature/Project		
<ul style="list-style-type: none"><li>Debugged the HW issues raised by SW team and resolved them quickly.</li><li>Reviewing the code changes raised by peer engineers</li><li>Proposed improvements in the VI testing infra to resolve the issues quickly.</li><li>Mentored junior engineers</li><li>Prepared the mid-level design document, test plan for the feature and present it to wider team.</li></ul>	Feature/Project	Intent	Outcome and Learnings
	Mixed Mode testing - Feature	Stress the WLAN HW by <ol style="list-style-type: none"><li>Enabling multiple features</li><li>Randomly changing the Wi-Fi parameters like BW, NSS, MCS.</li></ol>	<ul style="list-style-type: none"><li>Gained experience using the emulation environment</li><li>Understood the control path and data path in VI's uFW</li><li>Learned to use rand APIs to choose Wi-Fi parameters randomly.</li></ul>
	Themisto 1.0 and 2.0 - Project	Validate the WLAN features in both pre/post Silicon stage and Silicon stage	<ul style="list-style-type: none"><li>In both Themisto 1.0 and 2.0, brought up the WLAN features like rate walk, encryption, protection, aggregation via SW for each emulation release.</li><li><b>Implemented automation infrastructure to execute the test cases automatically and simulation infra to RCA the HW issues.</b></li><li><b>Identified 2 HW bugs (Decrypt failure &amp; Decode failure for long packets) in Pre-Silicon stage</b></li></ul>
	Trestles 1.0 - Project		<ul style="list-style-type: none"><li>Brought up basic rate walk via SW in E1P5 emulation build</li><li>Implemented recipe from DV team and validated the MAP feature.</li><li>As part of MAP feature validation, identified a HW bug related to it.</li></ul>

General roles and responsibilities	Details of feature		
	Title	Intent	Outcome and Learnings
<ul style="list-style-type: none"> <li>Sustenance support in the modules like Linux core kernel, ethernet and PPE driver.</li> <li>Reviewing the code changes raised by peer engineers</li> <li>Proposed suggestions to improve the performance of the AP</li> <li>Prepared the design document, test plan for the feature and present it to wider team</li> </ul>	Energy related products (ERP) regulations – phase 2	Enter low power mode where there is no activity for 20 minutes in the DUT	<ul style="list-style-type: none"> <li>Learned to use interface stats, conntrack details and CPU utilization present in kernel to detect the idle state and enter low power mode</li> </ul>
	Energy related products (ERP) regulations – phase 1	<ul style="list-style-type: none"> <li>CLI commands in the QSDK to enter and exit low power mode by making the interfaces to up and down respectively</li> <li>In the low power mode, the energy consumption will be low</li> </ul>	<ul style="list-style-type: none"> <li>Mastered to implement the CLIs and parsed the input from user</li> <li>Implemented a user space layer interact with kernel via NL socket</li> </ul>
	Passing the traffic via PPE with VLAN over bridge topology (Requested by Customer)	With VLAN over bridge topology, pass the traffic via PPE (Packet processing engine) to improve the performance	<ul style="list-style-type: none"> <li>Explored the ECM (Enhanced connection manager) driver for flow creation and learned to configure PPE HW via PPE driver to accelerate the traffic with VLAN over bridge topology.</li> </ul>
	Direct Switch Feature	The traffic between ethernet and Wi-fi will flow via hardware to improve the egress rate.	<ul style="list-style-type: none"> <li>Executed the test cases suggested by leads and resolved the issues which popped up.</li> <li>Provided sustenance support for the feature.</li> </ul>

Radio driver /firmware sustenance Team | June 2018 – April 2021 (EmbedUR Systems Chennai)

Title & Aim	Roles and Responsibilities
<p><b>Title:</b> WNBU-Cheetah NXP</p> <p><b>AIM:</b> Provide radio driver /firmware sustenance in CISCO AP (8964 Wi-fi chip based)</p>	<ul style="list-style-type: none"> <li>Fixing issues in the wireless radio driver/firmware code based on <b>Linux OS</b></li> <li>Evaluating and assessing the new feature request</li> <li>Providing technical guidance, leading and reporting the status of the team to internal manager periodically</li> <li>Performed Unit testing and conducted code review internally</li> </ul> <p><b>Issue/Modules worked:</b> Beacon management, Jitter issue, Mesh, Radio driver initialization, Client connectivity issues, data path &amp; control path</p>

General roles and responsibilities	Details of feature		
	Title	Intent of the feature	Outcome and Learnings
<ul style="list-style-type: none"> <li>Sustenance support in the BSP modules</li> <li>Reviewing the code changes raised by peer engineers</li> </ul>	Board bring up of OLT based on ARM A-53	From platform team perspective, bring up the board and start the applications	<ul style="list-style-type: none"> <li>Learned how inter core communication works in a SOC by implementing mailbox driver using RPMsg-Lite infrastructure</li> <li>Compilation of all SW modules using gcc10 tool chain to successfully build the ELF image</li> <li>Implemented a watch dog timer feature to update master core about the status of slave core</li> </ul>
	CPU Load measurement	To compute the CPU load periodically and raise an alarm to the application layer when CPU load is high	<ul style="list-style-type: none"> <li>Learned to create a task to compute the CPU load periodically from PCB (process control block) structure</li> <li>Raise an alarm to the application layer when CPU load is high and clear it when CPU comes to normal</li> <li>Implemented trace and debug command to raise/clear the alarm for testing</li> </ul>
	40GBASE-KR4 support in 7250 Switch	Scope is to operate the switch port in 40G and provide sustenance	<ul style="list-style-type: none"> <li>Implemented a state machine between switch and line card to operate the port in 40G</li> <li>Provided sustenance support till the feature is delivered in main stream</li> </ul>