

GANDIKOTA YADAGIRI

Senior Software Engineer | C++ | Adaptive AUTOSAR | SIL | QNX/Linux
+91 8977878755
yadagiri.gandikota95@gmail.com

Professional Summary:

Embedded Software Engineer with 2+ years of experience in automotive systems, SIL (Software-in-the-Loop) frameworks, and Adaptive AUTOSAR using C++ on QNX/Linux platforms. Proficient in CAN communication, DBC parsing, Bluetooth protocol stack (BlueZ), and debugging tools like Valgrind, Wireshark, and QNX Momentics. Passionate about delivering robust solutions for next-gen automotive ECUs and infotainment systems.

Technical skills with overall experience :

Programming Languages: C++, C, STL, Multithreading

RTOS/OS: QNX 7.1, Embedded Linux

Embedded Tools: Vector CANoe, DaVinci Configurator, QNX Momentics, QEMU

Protocols: CAN, UART, Ethernet, Socket Programming

Standards: AUTOSAR (Adaptive)

Debugging & Profiling: Valgrind, Wireshark, QNX Tracelogger, GDB

Platforms: Qualcomm 8295P/8155P, Renesas RH850, Raspberry Pi 4, TI ARM64

DevOps & CI/CD: Jenkins, Plastic SCM, Gerrit-Git

Agile Tools: Jira, Confluence

Professional experience :

Actevia Technology Services Pvt Ltd — Senior Software Engineer

PROJECT 2: SIL (Software-in-the-Loop) Testing for Automotive Software

Client: Aptiv **Role:** Software Developer

Team Size: 11 **Duration:** Nov 2024 – Till Date

Technologies: C++, STL, Vector CANoe, XML/CSV/HDF5, Linux, Aptiv internal tools, Plastic SCM, Valgrind

Description:

Developed a SIL framework to validate ECU software in virtualized environments without physical hardware. Enabled high-fidelity simulations of vehicle behavior and early bug detection.

Responsibilities:

- Designed a SIL architecture using C++ to simulate ECU signals and behaviors.
- Created a DBC decoder for CAN logs conversion into XML, CSV, HDF5 formats.
- Developed the RESIM stream decoder with 98% signal simulation accuracy.

- Integrated CANoe with SIL for automated validation of test cases.
- Identified high-risk software issues via edge-case scenario testing.

PROJECT 1: Adaptive AUTOSAR HPC Cluster (Qualcomm 8295P/8155P) – Mahindra XUV BE 6/9E

Client: Aptiv **Role:** C++ Developer

Team Size: 24 **Duration:** Mar 2023 – Nov 2024

Technologies: C++, Adaptive AUTOSAR, Vector CANoe, Davinci Developer Adaptive, QNX Momentics IDE, Gerrit-Git, QEMU, Wireshark, GDB, Valgrind, QFIL, ADB, Renesas

Description:

Contributed to the development and integration of Adaptive AUTOSAR applications, focusing on signal communication, persistence services, and system emulation for the Mahindra XUV BE 6/9E platform powered by Qualcomm HPC (8295P/8155P) SoCs.

Responsibilities:

- Developed Adaptive AUTOSAR applications and custom bridge components using C++ to facilitate communication between the Signal Manager and bridge services via the PPS (Persistent Publish/Subscribe) mechanism.
- Implemented and optimized the Persistence Cluster module, ensuring reliable storage and retrieval of signal data across reboots and over-the-air updates.
- Modified ARXML service interface definitions to comply with Vector DaVinci Developer Adaptive tooling standards, supporting seamless configuration and code generation.
- Generated Skeleton and Proxy code for Adaptive AUTOSAR services, enabling service communication within the application cluster.
- Performed in-depth software debugging using QNX Momentics IDE, GDB, and QNX tracelogger (.kev) files to identify bottlenecks and improve system stability.
- Configured QEMU-based emulation environment to test Adaptive AUTOSAR applications on Linux before deployment to QNX-based target hardware.
- Automated build processes and resolved integration issues using Gerrit, build scripts, and CI/CD pipelines to streamline development workflows.

Internal R&D Projects | Demonstrated to Multiple Automotive Clients

PROJECT: Bare Metal ESXi Hypervisor on Raspberry Pi 4

Technologies: VMware ESXi Hypervisor, ARM64 Architecture, UEFI, GParted

Project Overview:

Led the successful porting of VMware ESXi to Raspberry Pi 4 (ARM64), enabling bare-metal virtualization for running multiple guest operating systems simultaneously with efficient resource allocation and hypervisor-level control.

Key Contributions:

- Modified the Raspberry Pi's default boot sequence and UEFI firmware configuration to prioritize ESXi hypervisor boot, replacing the standard OS boot process.
- Ported ESXi to the ARM64 architecture, resolving hardware compatibility and driver-level challenges for stable execution on ARM-based systems.
- Implemented CPU core pinning and memory partitioning strategies to optimize resource allocation for multiple guest virtual machines, ensuring consistent performance.
- Configured vSwitch-based virtual networking within ESXi to enable robust inter-VM communication and external network access.
- Validated hypervisor stability and VM performance through functional testing and benchmarking, demonstrating the feasibility of ARM-based virtualization on low-cost hardware platforms.

PROJECT: Multi-Device Bluetooth Audio Streaming on Raspberry Pi 4

Technologies: BlueZ (Bluetooth Stack), A2DP, HFP, FTP, PulseAudio, ALSA, Bluetooth 5.0, Wireshark

Project Overview:

Engineered a Bluetooth 5.0 multi-device audio streaming system on Raspberry Pi 4 using a custom-built BlueZ stack with PulseAudio and ALSA, supporting real-time synchronized playback across multiple devices along with call and file transfer capabilities.

Key Contributions:

- Compiled and integrated the latest BlueZ stack from source, enabling support for A2DP, HFP, and FTP profiles with full Bluetooth 5.0 compatibility and stable multi-device connectivity.
- Configured PulseAudio and ALSA to mirror audio streams simultaneously to multiple Bluetooth audio devices, minimizing latency and ensuring playback synchronization.
- Implemented support for hands-free calls (HFP) and wireless file transfers (FTP), allowing the Raspberry Pi to serve as a central multimedia and communication hub.
- Conducted protocol-level analysis using Wireshark and BTSnoop to trace HCI, L2CAP, and AVDTP packets, diagnosing issues such as audio lag and packet drops.
- Developed automation scripts using bluetoothctl for seamless device pairing, trusted device management, and session handling across reboots.
- Tuned kernel-level Bluetooth configurations to enhance connection stability and reduce latency during high-load streaming scenarios.

Education:

Bachelor of Technology: Avanthi Institute of Engineering & Technology, JNTU Hyderabad

Year of Graduation: 2021

Intermediate: Sri Raghavendra Junior College Year of Completion: 2014

Secondary School Certificate: Sri Krishnaveni Talent High School Year of Completion: 2012