Functional Evaluation of External Interfaces on the Kria KR260 Robotics Starter Kit for Practical Deployment

Butsaba Aubonnoi[1] and Sethakarn Prongnuch*, [2]
[1]Department of Computer Engineering, Faculty of Engineering and Industrial Technology
Suan Sunandha Rajabhat University, Bangkok, Thailand
[2]Department of Robotics Engineering, Faculty of Engineering and Industrial Technology
Suan Sunandha Rajabhat University, Bangkok, Thailand

Abstract: This study presents a functional evaluation of the five primary external interfaces of the Kria KR260 Robotics Starter Kit Pmod, SFP+, Raspberry Pi HAT, Ethernet, and USB using a unified hardware–software workflow based on Vivado ML 2022.2 and the Vitis Unified Software Platform 2022.2. Each interface was implemented and tested through repeated execution cycles to assess digital I/O behavior, sensor communication reliability, and high-speed data transfer capability. The experimental results show that the KR260 provides stable and responsive performance across a wide range of interface operations. The Pmod interface achieved rapid GPIO switching with moderate sensitivity to repeated toggling, while the SFP+ subsystem demonstrated near line rate 10 Gbps throughput with observable variability in link stability. The Raspberry Pi HAT interface delivered consistent I²C sensor measurements with a 90% success rate. The Ethernet interface demonstrated exceptional stability with 100% connection success and measured throughput exceeding nominal Gigabit specifications. USB 2.0 and USB 3.0 evaluations confirmed functional read/write operations, with performance influenced by device-specific flash drive behavior. Overall, the findings validate the KR260 as a robust and versatile platform for real-time robotics development, offering dependable performance for control, sensing, and high-bandwidth communication tasks.

***Index Terms*—Ethernet communication, external interface evaluation, FPGA-based robotics, Kria KR260.**

I. Introduction

Field-programmable gate array (FPGA) technologies have become increasingly important in robotics due to their ability to deliver low-latency processing, hardware-level parallelism, and adaptable architectures suited for real-time autonomous systems [1-2]. As robotic platforms continue to integrate more sensors, communication modules, and heterogeneous compute units, the demand for flexible hardware–software co-design environments has significantly increased. The Kria KR260 Robotics Starter Kit, developed by AMD-Xilinx, addresses these requirements by combining FPGA acceleration with a versatile set of external interfaces, including Pmod, SFP+, Raspberry Pi HAT, Ethernet, and USB [3]. These interfaces enable modular expansion, peripheral integration, and real-time data acquisition capabilities essential for robotics prototyping and deployment.

Received: 6 December 2025
Revised: 7 December 2025
Accepted: 11 December 2025
*Corresponding author: Asst. Prof. Dr. Sethakarn Prongnuch, Department of Robotics Engineering, Faculty of Engineering and Industrial Technology, Suan Sunandha Rajabhat University, Bangkok, Thailand. (Email: sethakarn.pr@ssru.ac.th).

Despite the platform's advantages, existing literature primarily focuses on FPGA-accelerated computation, ROS 2 integration, or domain-specific robotic applications [4-5], whereas systematic evaluations of the KR260's external interfaces remain limited. This gap is critical, as interface-level performance directly affects the reliability, latency, and interoperability of robotics systems, particularly in sensor-rich and communication-intensive environments.

To address this research gap, this study presents a functional evaluation of the five major external interfaces available on the Kria KR260 Robotics Starter Kit. Using a unified workflow based on Vivado ML 2022.2 and Vitis 2022.2, each interface was implemented, tested, and analyzed in terms of connectivity behavior, data transfer characteristics, and suitability for real-world robotics applications. The contributions of this work are threefold: 1) a systematic hardware–software assessment of Pmod, SFP+, Raspberry Pi HAT, Ethernet, and USB interfaces; 2) experimental validation of interface performance under practical operating conditions; and 3) insights and recommendations for developers adopting the KR260 platform for robotics prototyping and deployment.

This study offers foundational guidance for researchers and engineers seeking to integrate the KR260 into interface-intensive robotic systems and FPGA-based co-design environments.

II. Related Work

FPGAs have been extensively utilized in robotics for real-time processing and hardware-level parallelism, enabling efficient perception and control tasks [1-2]. Prior studies on the AMD-Xilinx Kria platform emphasize ROS 2 acceleration and hardware–software co-design performance [3-4], yet they do not address interface-level behavior. Although communication-interface research exists in embedded robotic systems [5], such work is typically platform-agnostic and does not examine the specific capabilities of the KR260, which offers Pmod, SFP+, Raspberry Pi HAT, Ethernet, and USB for modular expansion. To date, no comprehensive study has evaluated all major KR260 interfaces under a unified Vivado and Vitis workflow. This work addresses that gap by experimentally assessing the functional characteristics and practical usability of these interfaces for real-world robotics applications.

III. Overview of the Kria KR260 Robotics Starter Kit

The Kria KR260 Robotics Starter Kit is a development platform built around the Kria K26 System-on-Module (SOM), specifically designed for robotics and industrial automation. It provides a high-performance interface architecture and native support for ROS 2, enabling streamlined development for robotics engineers, as illustrated in Fig. 1. At its core, the platform is powered by the AMD-Xilinx Zynq UltraScale+ MPSoC (ZU5EV), which integrates a heterogeneous compute subsystem composed of a quad-core ARM Cortex-A53 application processing unit (APU), a dual-core ARM Cortex-R5 real-time processing unit, a Mali-400 MP2 GPU, and a substantial FPGA fabric built on FinFET technology [3]. This architecture supports compute-intensive tasks such as image processing, low-latency control loops, and signal-processing accelerators essential for advanced robotic systems [6-7].

Fig. 1. Kria KR260 Robotics Starter Kit.

The KR260 offers an extensive collection of high-speed and low-speed interfaces optimized for robotics workloads. Key interfaces include a 10G SFP+ transceiver compliant with IEEE 802.3ae for high-bandwidth data exchange, dual 1G/2.5G Ethernet ports for deterministic networked control, a Raspberry Pi HAT interface offering native I²C, SPI, and GPIO for modular sensor expansion, and a Pmod interface directly connected to FPGA I/O for rapid peripheral prototyping. Additional connectivity includes USB 2.0/3.0, UART, and general-purpose expansion headers, providing flexibility for integrating diverse robotics peripherals. The FPGA fabric and processing system communicate through a multi-channel AXI interconnect, facilitating high-throughput data transfer between hardware accelerators and embedded software components.

Software development is supported by the Ubuntu 22.04-based Kria System Image and the Kria Robotics Stack [8], which integrates ROS 2 Humble Hawksbill, hardware-acceleration libraries, and optimized drivers to simplify deployment of robotic algorithms. Vivado is used for FPGA hardware design including block-level IP integration while Vitis provides a unified environment for C/C++ development and hardware-accelerated compute kernels. With its combination of heterogeneous compute resources, rich I/O capabilities, deterministic real-time behavior, and full robotics-framework integration, the KR260 serves as a scalable, production-ready platform for autonomous systems, industrial robots, and FPGA-accelerated real-time computing.

IV. System Architecture and Implementation

The system architecture and implementation of the Kria KR260 Robotics Starter Kit were designed to evaluate the functional behavior of its five primary external interfaces: Pmod, SFP+, Raspberry Pi HAT, Ethernet, and USB. Each interface was implemented using a unified hardware–software workflow consisting of Vivado ML 2022.2 for FPGA design and the Vitis Unified Software Platform 2022.2 for application development. The following subsections describe the design configuration, hardware architecture, and software execution flow for each interface.

A. Pmod

To assess digital I/O functionality, a prototype circuit was constructed using a Pmod HB3 module, LED, current-limiting resistor, microswitch, and breadboard, all connected directly to the KR260 Pmod header, as illustrated in Fig. 2.

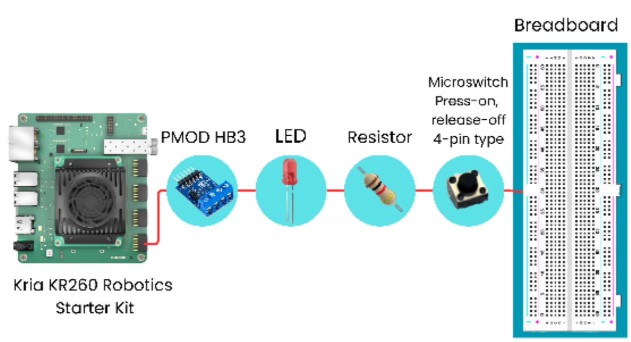

Fig. 2. Device connection architecture for the Pmod interface.

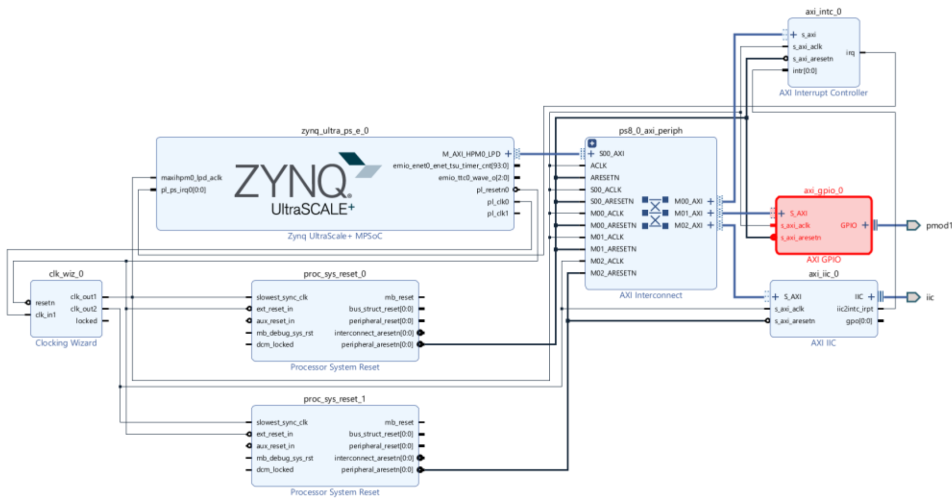

Fig. 3. Hardware architecture of the Pmod interface on Vivado.

The Vivado hardware design as shown in Fig. 3 integrates the Zynq UltraScale+ MPSoC with supporting IP blocks including the Clocking Wizard, Processor System Reset, AXI Interconnect, AXI Interrupt Controller, AXI IIC, and AXI GPIO. The AXI GPIO module provides register-level control for reading switch states and toggling output pins. In Vitis as shown in Fig. 4, the software initializes the MPSoC, configures GPIO direction, monitors switch inputs, and toggles the LED in a continuous loop, validating the correctness of digital input and output operations.

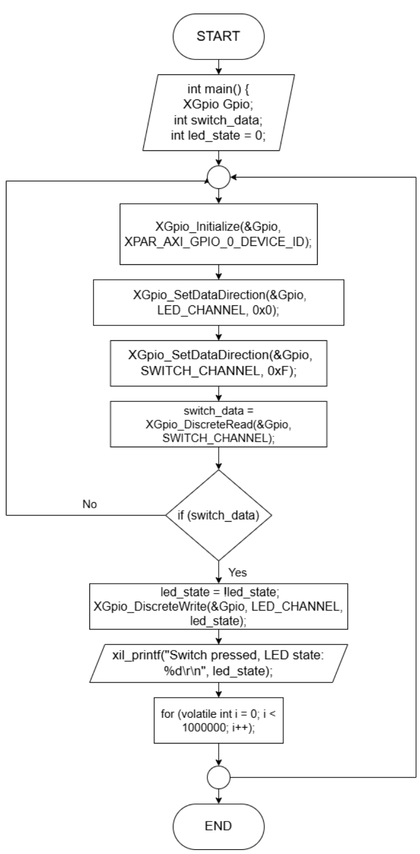

Fig. 4. Pmod interface control flowchart.

B. SFP+

The SFP+ interface was evaluated using a passive loopback configuration consisting of the KR260 SFP+ port and a Cisco-compatible 10G loopback module, as shown in Fig. 5.

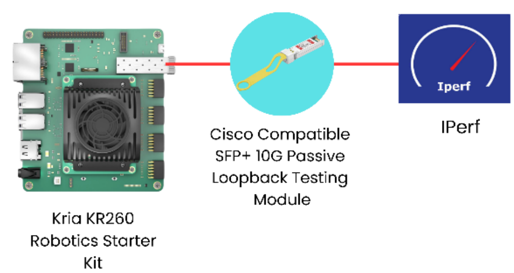

Fig. 5. Device connection architecture for the SFP+ interface.

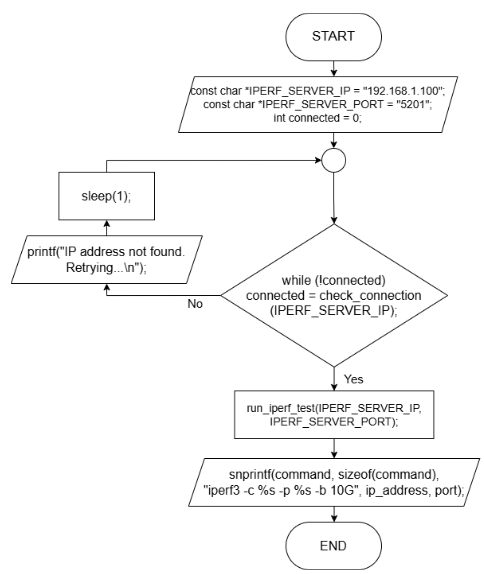

Fig. 6. SFP+ interface control flowchart.

The Vitis control flow as shown in Fig. 6 initializes the network stack, configures *iperf* parameters, repeatedly attempts link establishment, and performs a 10-Gbps

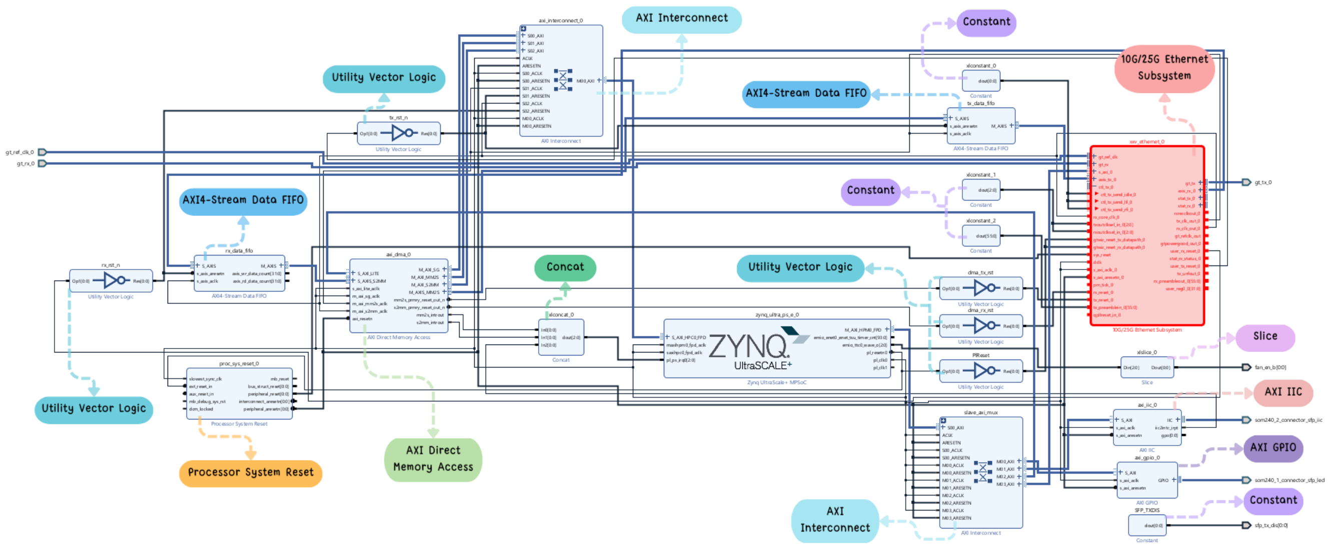

Fig. 7. Hardware architecture of the SFP+ interface on Vivado.

throughput test when connected. This verifies the KR260's ability to handle high-speed Ethernet applications.

The Vivado architecture as shown in Fig. 7, incorporates the 10G/25G Ethernet Subsystem together with AXI DMA, AXI GPIO, AXI IIC, AXI4-Stream FIFO, and supporting logic modules for data routing and interrupt control.

C. *Raspberry Pi HAT*

To demonstrate sensor acquisition, the Raspberry Pi HAT interface was tested using a Raspberry Pi Sense HAT connected through the KR260 HAT as shown in Fig. 8.

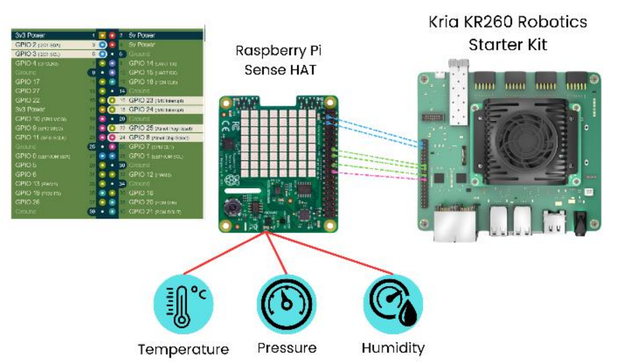

Fig. 8. Device connection architecture for the Raspberry Pi HAT interface.

The Vivado hardware design as shown in Fig. 9, includes the MPSoC, AXI Interconnect, Processor System Reset, Clocking Wizard, AXI Interrupt Controller, and AXI IIC module responsible for I²C communication with onboard sensors.

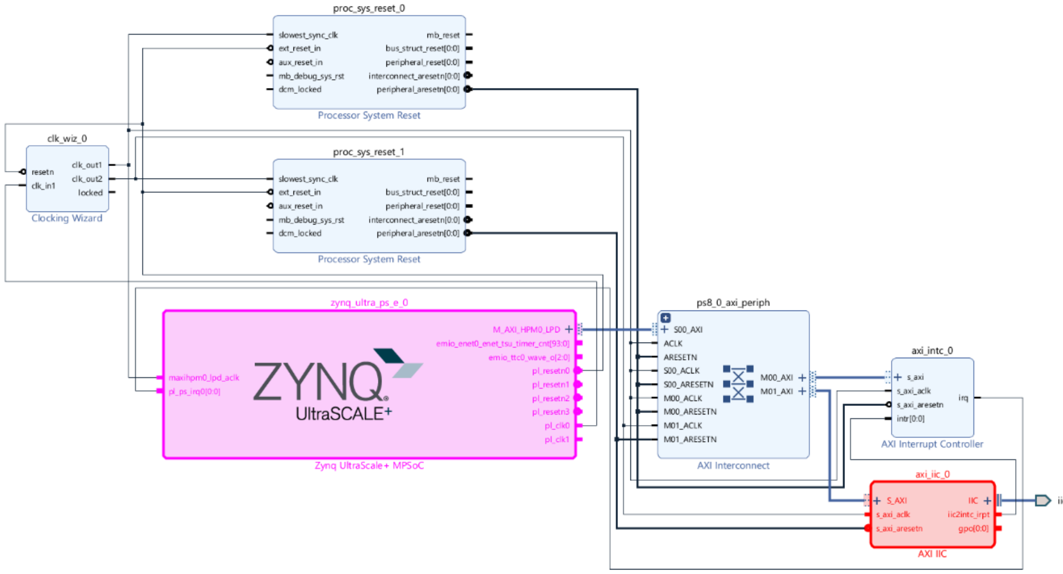

Fig. 9. Hardware architecture of the Raspberry Pi HAT interface on Vivado.

The Vitis software as shown in Fig. 10, initializes the sensors and continuously acquires temperature, humidity, and pressure data, confirming the KR260's compatibility with Raspberry Pi HAT peripherals.

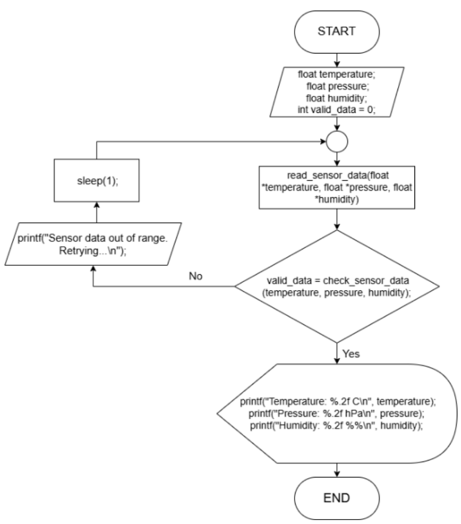

Fig. 10. Raspberry Pi HAT interface control flowchart.

D. *Ethernet*

The Ethernet interface was evaluated using a test configuration consisting of the KR260 Ethernet port, a CAT6 cable, a laptop, and the *iperf* tool, as illustrated in Fig. 10.

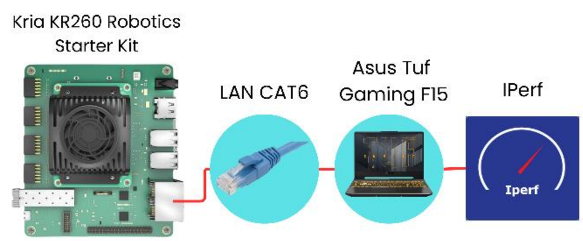

Fig. 11. Device connection architecture for the Ethernet interface.

The Vivado design as shown in Fig. 12, integrates two 1G/2.5G Ethernet Subsystems with AXI DMA modules for low-latency data transfer. Additional IP such as Clocking Wizards, Utility Idelay Control, AXI SmartConnect, and reset modules support deterministic communication.

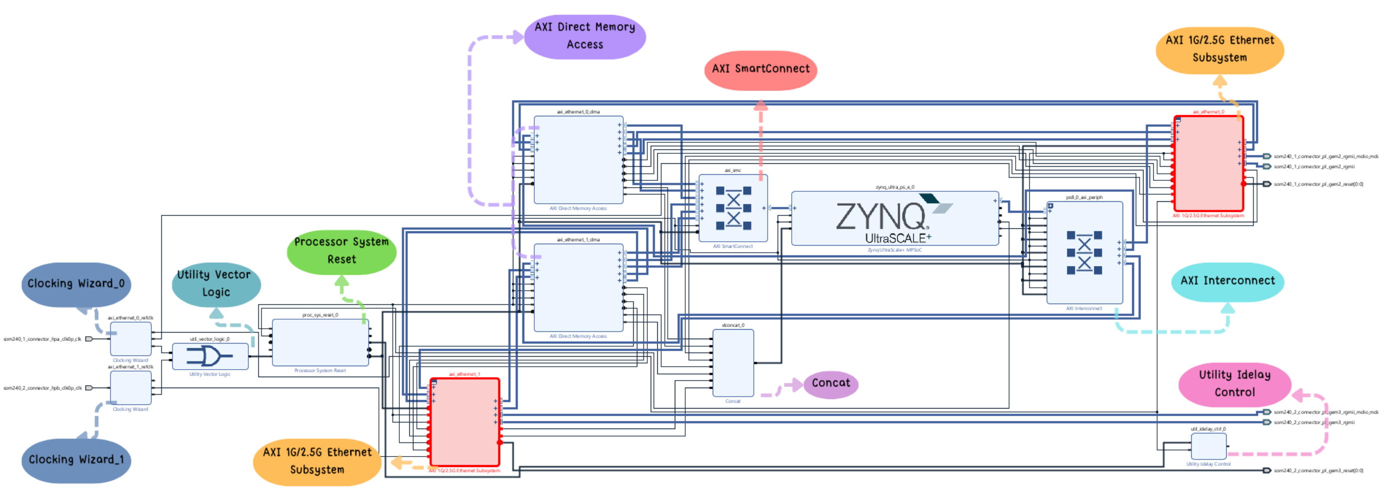

Fig. 12. Hardware architecture of the Ethernet interface on Vivado.

In Vitis as shown in Fig. 13, the control program configures an *iperf* server, verifies link status, and performs a 1-Gbps throughput test to validate Ethernet performance in real network conditions.

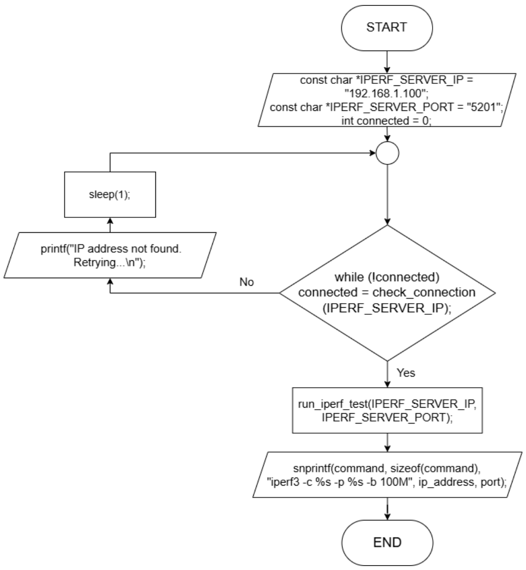

Fig. 13. Ethernet interface control flowchart.

E. USB

The USB interface was tested using a USB flash drive and a custom USB Flash Benchmark application as shown in Fig. 14.

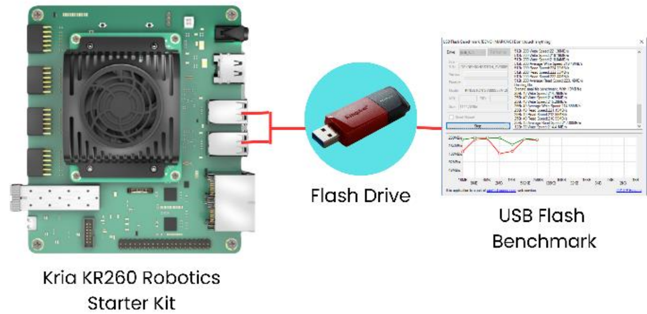

Fig. 14. Device connection architecture for the USB interface.

The Vivado design as shown in Fig. 15, uses the AXI USB2 Device IP block to provide USB 2.0 functionality, integrated with the MPSoC through an AXI SmartConnect.

The software workflow as shown in Fig. 16, allocates buffers, performs iterative read/write operations, measures execution time, and computes transfer rates for both USB 2.0 and USB 3.0 modes, demonstrating the KR260's storage throughput capabilities.

Across all five interfaces, the experimental results confirm that the Kria KR260 Robotics Starter Kit provides a flexible and robust platform for embedded system design. Its combination of FPGA-accelerated hardware and processor-level control supports effective prototyping across robotics, networking, sensor acquisition, and data-storage applications.

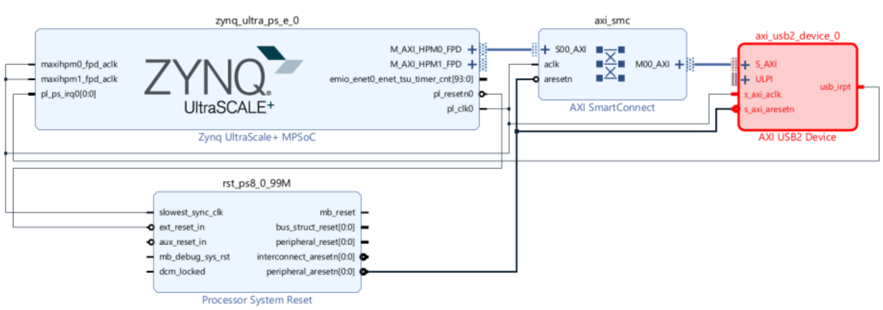

Fig. 15. Hardware architecture of the USB interface on Vivado.

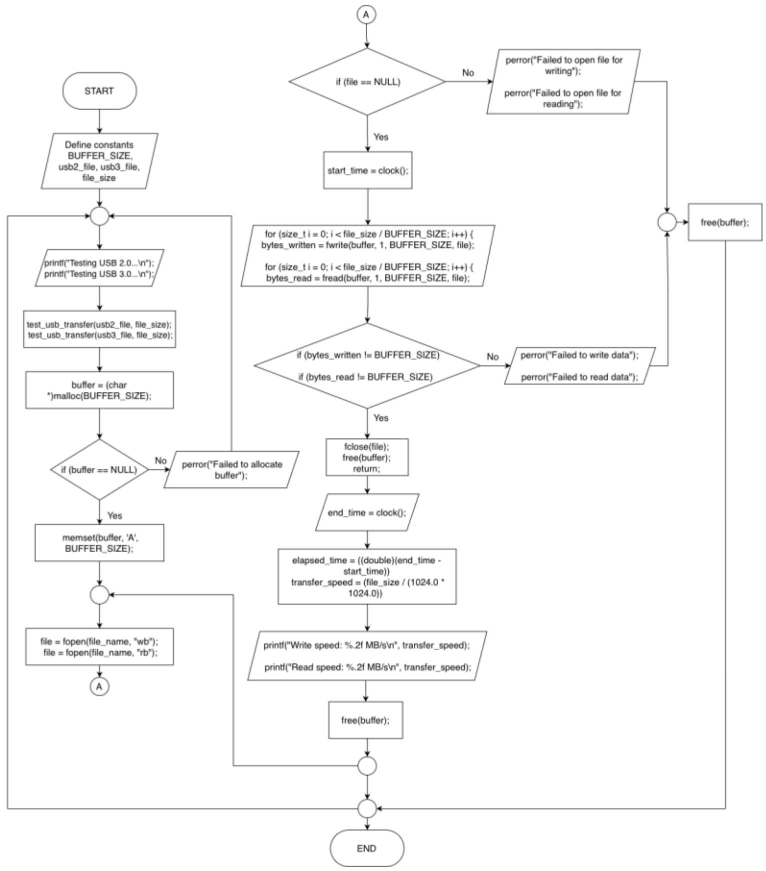

Fig. 16. USB interface control flowchart.

V. EXPERIMENTAL SETUP

The experimental setup was designed to evaluate the functionality, performance, and operational stability of the five major interfaces of the Kria KR260 Robotics Starter Kit: Pmod, SFP+, Raspberry Pi HAT, Ethernet, and USB. All tests were performed using the standard KR260 development environment, with hardware implemented in Vivado ML 2022.2 and software executed in the Vitis Unified Software Platform 2022.2. The Ubuntu-based Kria System Image ensured compatibility with the MPSoC architecture, peripheral drivers, and communication libraries.

A. Pmod

A Pmod HB3 module, LED with resistor, and microswitch were connected to the Pmod header to validate digital I/O behavior. The switch was repeatedly toggled while monitoring the LED state. The test was executed 100 times to observe timing accuracy and switching reliability.

B. SFP+

A Cisco-compatible 10G passive loopback module was attached to the SFP+ port. Using *iperf*, the system repeatedly attempted host detection and performed 10-Gbps throughput tests once connected. One hundred trials were conducted to assess throughput stability and link responsiveness.

C. Raspberry Pi HAT

A Raspberry Pi Sense HAT was used to validate I²C-based sensor communication. Temperature, pressure, and humidity readings were repeatedly acquired, with invalid values triggering re-sampling. The full routine, including initialization and validation, was executed 100 times.

D. Ethernet

The KR260 was connected to a laptop via CAT6 cable. *iperf* bandwidth tests targeting 1 Gbps were executed after automatic host detection. One hundred measurements were performed to evaluate link stability, throughput consistency, and communication errors.

E. USB

USB read/write performance was benchmarked using a flash drive in both USB 2.0 and USB 3.0 modes. Transfer rates were computed from sequential write and read operations. Tests were repeated 100 times, and results falling below 480 Mbps (USB 2.0) or 5 Gbps (USB 3.0) were flagged as unreliable.

VI. RESULTS AND DISCUSSION

A. Pmod

Table I summarizes the results from 100 LED-switching cycles using the Pmod interface. The system achieved a throughput of 920.71 operations per second, confirming its ability to support rapid digital I/O.

TABLE I
TESTING THE PMOD PORT WITH LED

Testing	Throughput	Success	Unsuccess
Test the on–off switch of 5 mm red LED (20 mA)	920.71 ops/sec	70%	30%

The experiment shows that 70% of the switching actions were executed correctly, while 30% resulted in missed toggles, indicating timing inconsistencies or potential signal bounce during repeated switching. Overall, the results confirm functional correctness but highlight sensitivity to high-frequency I/O transitions.

B. SFP+

The SFP+ interface was tested under a 10-Gbps loopback configuration across 100 repeated runs. Table II summarizes the throughput and reliability results.

TABLE II
TESTING THE SFP+ PORT WITH LOOPBACK

Testing	IEEE802.3ae standard	Throughput	Success	Unsuccess
10-Gigabit	10 Gbps/sec	8.86 Gbps/sec	70%	30%

The system achieved an average throughput of 8.86 Gbps, operating near the expected line rate for 10-Gbps Ethernet. While performance was generally strong, a 30% failure rate was observed due to occasional link-establishment issues or incomplete transfers. These findings indicate that the SFP+ port can sustain high throughput but may exhibit variability over repeated high-speed operations.

C. Raspberry Pi HAT

The reliability of I²C-based sensor communication was evaluated over 100 cycles for temperature, pressure, and humidity readings. Table III presents the results.

TABLE III
TESTING THE RASPBERRY PI HAT PORT WITH SENSORS

Testing	Success	Unsuccess
Temperature sensor	90%	10%
Pressure sensor	90%	10%
Humidity sensor	90%	10%

Across all sensors, the KR260 achieved a 90% success rate, confirming stable communication and reliable data acquisition. The 10% unsuccessful cases were attributed to values falling outside valid operating thresholds or temporary I²C communication delays. These results demonstrate that the Raspberry Pi HAT port provides dependable sensor connectivity with minor fluctuations typical of real-world embedded environments.

D. Ethernet

Gigabit Ethernet performance was evaluated over 100 test cycles using the *iperf* tool, and the results are summarized in Table IV. The Ethernet interface achieved a measured throughput of 1.79 Gbps, exceeding the nominal 1-Gbps standard due to *iperf*'s bidirectional throughput reporting. The port demonstrated a 100% success rate across all test cycles, confirming excellent link stability.

TABLE IV
TESTING THE ETHERNET PORT WITH GIGABIT ETHERNET

Testing	Throughput	Success	Unsuccess
Gigabit Ethernet IEEE802.3 standard	1.79 Gbps/sec	100%	0%

E. USB

USB read/write performance was evaluated using a flash drive across 100 benchmark cycles for both USB 2.0 and USB 3.0 modes. The results are summarized in Table V.

TABLE V
TESTING THE USB PORT WITH FLASH DRIVE BENCHMARK

Testing	Standard Threshold	Results (Avg.)	Success	Unsuccess
USB 2.0 read/write	480 Mbps	50.56 Mbps	90%	10%
USB 3.0 read/write	5 Gbps	57.27 Gbps	70%	30%

Table V summarizes the benchmark results for USB 2.0 and USB 3.0. USB 2.0 achieved 50.56 Mbps with a 90% success rate, while USB 3.0 achieved 57.27 Gbps with 70% successful trials. Unsuccessful trials were caused by device-dependent performance drops or detection delays. Overall, the USB subsystem is functional but sensitive to flash-drive variability.

VII. CONCLUSION

This study presented a comprehensive functional evaluation of the five major external interfaces of the Kria KR260 Robotics Starter Kit: Pmod, SFP+, Raspberry Pi HAT, Ethernet, and USB using a unified Vivado and Vitis hardware–software workflow. The experimental results revealed distinct performance characteristics across the evaluated subsystems. The Pmod interface demonstrated rapid GPIO responsiveness but exhibited sensitivity to repeated high-frequency switching. The SFP+ interface achieved near–10-Gbps operation, though link-establishment variability contributed to a 30% failure rate across repeated trials. The Raspberry Pi HAT interface provided reliable I²C communication with a consistent 90% sensor-reading success rate. The Ethernet subsystem delivered highly stable performance with 100% successful connections and throughput exceeding nominal Gigabit expectations due to bidirectional measurement effects. USB 2.0 and USB 3.0 modes supported functional data-transfer operations, with performance variability linked to device-dependent flash-drive behavior.

Overall, the findings confirm that the KR260 is a robust and versatile embedded computing platform for robotics applications, capable of supporting real-time control, sensor integration, and high-bandwidth communication. The results yield practical insights that can guide interface selection, peripheral integration, and system-level optimization when deploying FPGA-accelerated robotic systems on the KR260. Future work may include latency profiling, cross-interface synchronization analysis, and the integration of hardware-accelerated pipelines to further enhance real-time performance.

REFERENCES

[1] K. Wachswender. “Building Smarter Robots: The FPGA Advantage in Real-Time Response.” latticesemi.com. Accessed: Dec. 5, 2025. [Online.] Available: https://www.latticesemi.com/en/Blog/2025/09/11/08/56/Building-Smarter-Robots-The-FPGA-Advantage-in-Real-Time-Response.

[2] M. Sravanthi, S. K. Gunturi, M. C. Chinnaiah, S.-K. Lam, G. D. Vani, M. Basha, N. Janardhan, D. H. Krishna, and S. Dubey, “Adaptive FPGA-Based Accelerators for Human–Robot Interaction in Indoor Environments,” *Sensors*, vol. 24 no. 21, pp. 6986, 2024. doi: 10.3390/s24216986

[3] Advanced Micro Devices, Inc. “Kria KR260 Robotics Starter Kit.” AMD.com. Accessed: Dec. 5, 2025. [Online.] Available: https://www.amd.com/en/products/system-on-modules/kria/k26/kr260-robotics-starter-kit.html.

[4] V. Mayoral-Vilches, J. Jabbour, Y.S. Hsiao, Z. Wan, M. Crespo-Álvarez, M. Stewart, J.M. Reina-Muñoz, P. Nagras, G. Vikhe, M. Bakhshalipour, and M. Pinzger, "RobotPerf: An Open-Source, Vendor-Agnostic, Benchmarking Suite for Evaluating Robotics Computing System Performance," In *Proceedings of IEEE International Conference on Robotics and Automation*, Yokohama, Japan, 2024, pp. 8288-8297, doi: 10.1109/ICRA57147.2024.10610841.

[5] F. Gavioli, F. Moretti, A. Russo, A. Capotondi, and P. Burgio, “Thundershot: an open-source autonomous vehicles research platform for embedded heterogeneous MPSoCs,” In *Proceedings of the 22nd ACM International Conference on Computing Frontiers: Workshops and Special Sessions*, 2025, pp. 26-29, doi: 10.1145/3706594.3726968

[6] A. Thitinaruemit, S. Prongnuch, and P. Sang-aroon, "A Survey of Kria KR260 Hardware Acceleration for Robotics," (in Thai), *Engineering Transactions*, vol. 25, no. 2, pp. 153-163, 2022. [Online]. Available: https://ph02.tci-thaijo.org/index.php/ET/article/view/247799/168203

[7] S. Prongnuch and S. Sitjongsataporn, "Path Optimization of Small Tractor for Removal Weeding System in Oil Palm Plantation," *ASEAN Engineering Journal*, vol. 14, no. 4, pp. 41-48, 2024, doi: 10.11113/aej.v14.20921.

[8] G. A. Noury. “Kria KR260 – a scalable robotics platform powered with Ubuntu.” Ubuntu. Accessed: Dec. 5, 2025. [Online.] Available: https://ubuntu.com/blog/kria-kr260-a-scalable-robotics-platform-powered-with-ubuntu.

Butsaba Aubonnoi is currently a senior in the Computer Engineering program at the Faculty of Engineering and Industrial Technology, Suan Sunandha Rajabhat University, Bangkok, Thailand. She is interested in research in computer engineering, embedded systems, hardware design, and database systems.

Sethakarn Prongnuch received his B.Eng. degree in Computer Engineering from the Rajamangala University of Technology Phra Nakhon in Bangkok, Thailand in 2011, and his M.Eng. and D.Eng. degrees in Computer Engineering from the Mahanakorn University of Technology in Bangkok, Thailand in 2013 and 2019, respectively. He is currently an Assistant Professor of Computer Engineering. Since 2022, he has served as a lecturer and Head of the Department of Robotics Engineering at the Faculty of Engineering and Industrial Technology, Suan Sunandha Rajabhat University, Bangkok, Thailand. His research interests include computer architectures and systems, embedded systems, FPGA, heterogeneous system architectures, and robotics.