EasyChair Preprint

№ 15965

Design-for-Test Strategies for Chiplets and Multi-Chip Module Integrations

Peter Wang

June 27, 2025

Design-for-Test Strategies for Chiplets and Multi-Chip Module Integrations

Peter Shun Shen Wang
JTAG Technologies
Singapore
IEEE1149@jtag.com.sg

Peter ven den Eijnden
JTAG Technologies B.V.
Eindhoven, the Netherlands
Peter@jtag.nl

Abstract*—This paper presents two advanced methodologies and three diagnostic approaches that significantly enhance Design-for-Test (DfT) capabilities in chiplet-based Multi-Chip Module (MCM) architectures. By leveraging machine intelligence and JTAG-compliant boundary scan infrastructure, the proposed strategies enable efficient identification and diagnosis of interconnect and functional faults in unknown dies. The combination of a twin-MCM test architecture, Connection Test Mode (CTM), and Wishbone-JTAG interfaces notably improves adaptive fault coverage and functional validation, particularly for modern chiplet-based automotive electronics.

Keywords—*MCM, KGD, JTAG, Boundary Scan, Machine Intelligence, Deep Learning, Chiplets, IEEE1149, DFT, FCT, AI*

I. INTRODUCTION

As chiplet integration and advanced packaging technologies become more prevalent**,** especially in automotive electronics**,** new DfT strategies are essential for ensuring the quality and reliability of complex Multi-Chip Modules (MCMs). This paper introduces two diagnostic methodologies supported by three practical approaches as follows:

1. **Machine Intelligence for Interconnect Testing**– Utilizes adaptive diagnostics and predictive algorithms to detect anomalies such as signal integrity issues and unexpected IC-related behaviors.
2. **Boundary Scan-Based Diagnostics**– Leverages the Provision System to identify interconnect and functional issues through JTAG-compliant boundary scan standards.
3. **Integrated Functional Testing (IFT)**– Merges JTAG and Wishbone Bus interfaces via the Core-Commander platform for both structural and functional validation in a unified test environment.

Method I-- introduces a twin-MCM-based architecture [3] [4] designed for validating unknown dies [1], also solves an internal disruption within the IC during boundary scan test [16]. Unlike the traditional single-MCM setups, the twin-MCM configuration enables robust infrastructure and interconnect testing between paired dies via a conduit board [2]. It enhances rigorous Ball Grid Array (BGA) pin validation and die recognition (ID) to be revealed during chiplet bumping and flip-chip process.

Method II-- provides a comprehensive Design-for-Testing (DfT) framework demonstrated across three case studies using JTAG-Netlists (JTN) derived from schematics and integrated with Boundary Scan ICs. The testing process includes [5] [6]:

- Infrastructure Testing: Validates presence and continuity of boundary scan ICs within the JTAG daisy-chain.
- Interconnection Testing: Ensures pin-to-pin connectivity and IEEE 1149.1.6 compliance.
- Cluster Testing: Extends test coverage to non-boundary scan devices, such as memory chips.
- In-System Programming (ISP): Prepares embedded memories by programming firmware for CPUs, MCUs, and FPGAs for functional testing.

An advanced net-alternating strategy toggles boundary-scan cells between high (“1”) and low (“0”) logic levels to verify bidirectional signal integrity. Each I/O pin typically includes two boundary-scan cells—drive and sense—defined in the Boundary Scan Description Language (BSDL) file [8]. Pins such as power, ground, no-connects, or analog interfaces are categorized as “linkage bits,” meaning they cannot be directly driven or sensed and therefore fall outside the boundary-scan test domain [16]. Mixed-function channels, configured as transmit-only or receive-only, behave according to their BSDL designations [10]. These asynchronous configurations are summarized in Table 1 [16].

PCIE_A_REFCLK_P	out bit	P channel (Boundary Scan cells)
PCIE_A_REFCLK_M	out bit	N channel (linkage bit)
PCIE_B_REFCLK_P	out bit	P channel (Boundary Scan cells)
PCIE_B_REFCLK_M	out bit	N channel (linkage bit)
QLINK_CLK_P	inout bit	P channel (Boundary Scan cells)
QLINK_CLK_M	inout bit	N channel (linkage bit)
QLINK_UDL1_P	inout bit	P channel (Boundary Scan cells)
QLINK_UDL1_M	inout bit	N channel (linkage bit)
QLINK_UDL2_P	inout bit	P channel (Boundary Scan cells)
QLINK_UDL2_M	inout bit	N channel (linkage bit)

Table 1. P-N channel-based boundary scan cells with altered linkage bits.

In practical applications, one-way communication paths are widely employed in autonomous driving and monitoring systems—for instance, in AI-enabled road-mapping that receives uplinked commands and transmits downlinked image or video data for analysis [7]. This proposal emulates such communication patterns, focusing on the receiving operation, and utilizes a twin-MCM-based platform to simulate ground-to-air interactions between a host system and a peripheral unit.

By emulating this communication flow in a controlled test environment, the approach minimizes hardware and software overhead during prototyping, which is then applied to mass production, as illustrated in Chart 1.

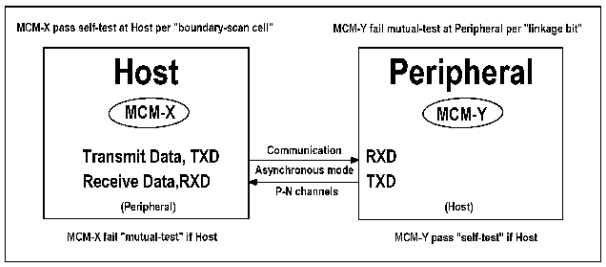


Chart 1. Uplink and downlink communication between the host and peripheral unit.

The infrastructure test accurately retrieves each die's unique 32-bit binary ID, ensuring IEEE 1149.1 compliance. Unlike conventional approaches that rely on generic designators (e.g., U1, U2, U3), this method uses silicon-embedded identifiers to address and manage each die within the system. During both infrastructure (Infra) and interconnection (Inter) testing phases, the JTAG-Netlist (JTN) queries each die's silicon ID to verify its active presence and proper alignment within the JTAG daisy chain. This process ensures traceability and eliminates ambiguity in die identification. The Test Access Port (TAP) interface is central to this operation, enabling the serial chaining of CPUs, MCUs, and FPGAs, among other ICs, while maintaining synchronization of key control signals, including TDI, TDO, TCK, TMS, and TRST, in boundary scan mode, as illustrated in Figure 1.

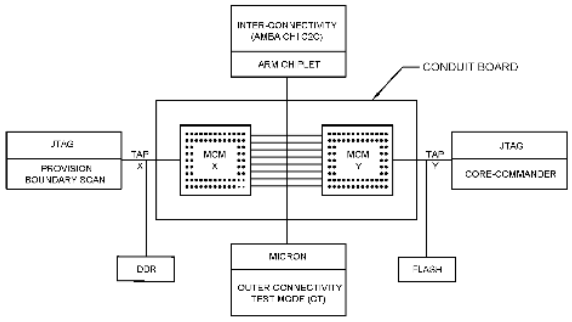


Fig. 1. Twin-MCM-based platform with a serially connected daisy chain over parallel buses.

To address linkage pins, this method reconfigures mismatched pins during interconnection testing; the JTAG-based tests perform adaptive reconfiguration of boundary-scan cells. This includes dynamically alternating pins between "drive-a" and "drive-b" or "sense-a" and "sense-b" modes. The system also toggles logic levels high ("1") or low ("0") in real time, depending on whether the pin functions as an "out" or "inout" node. This adaptability is managed by the JTAG-based Provision System, which incorporates machine intelligence to align each pin's role with its corresponding JTAG-Netlist (JTN) definition. The system seamlessly coordinates "drive" and "sense" cells to generate the appropriate logic levels in real-time, based on functional and topological context.

This method improves performance on asynchronous transmission lines, particularly within P-N channel configurations:

- N-channels serve as dedicated "out" paths utilizing linkage bits, which are untestable via boundary-scan.
- P-channels, in contrast, support both drive and sense operations and are fully testable within the scan domain.

To further optimize testing efficiency and prevent signal contention at linkage bits, the system allows intelligent pin swapping among devices sharing the same chip ID. This pin interleaving strategy avoids signal conflicts by alternating linkage assignments, resulting in a more adaptable bus structure. As a result, the interconnected nets exhibit characteristics similar to neural networks, with braided bus paths between paired modules (MCM-x and MCM-y) overlapping to enhance signal distribution. This architectural behavior is illustrated in Figure 2.

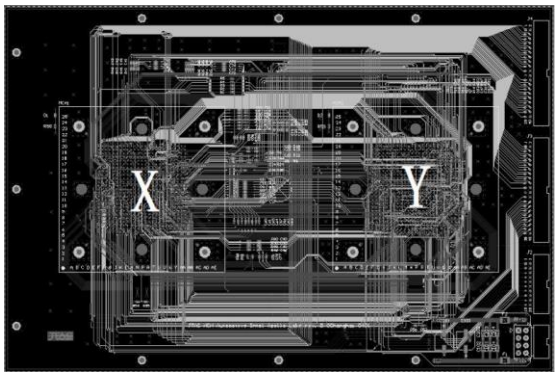


Fig. 2. JTAG's machine intelligence provisions known good dies (KGD)

A successful test outcome in this scenario is validated using an array of look-up tables, where columns represent test vectors and rows correspond to the combined netlist derived from JTAG Test Netlist (JTN) files of both MCMs. This structure enables systematic verification of signal integrity across all interconnections. The results of the interconnection test are captured and analyzed through a Truth Table Report (TTR)**,** which reflects the expected versus observed logic states for each test condition. The Truth Table displays each net (connection between pins), along with the status of each boundary-scan driver/sensor at different vector levels. Any sensed errors, indicating faulty connections, are highlighted in the table. BSD is an optional, advanced software module that works as a post-processor for the results data derived from the JTAG-based Technologies for test execution. This representation offers a clear and concise method for identifying discrepancies and verifying connectivity. The TTR structures are illustrated in Figure 3.

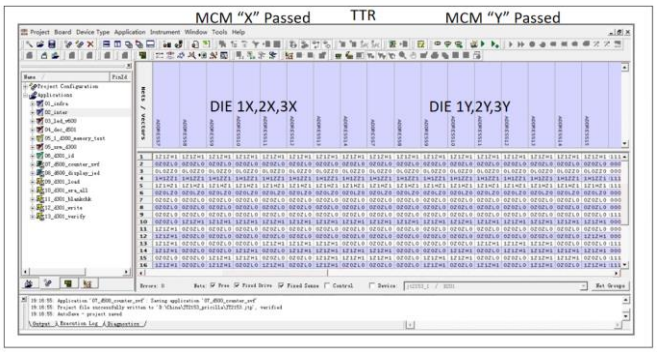


Fig. 3. Interconnection Test validates dies between MCM-x & y

However, violations may still occur, particularly at linkage bits or across unintended paths such as I2C buses. To address these issues, the JTAG-based Boundary Scan Diagnosis (BSD) program is employed. BSD automatically detects and analyzes network-related faults, including open circuits, shorts, and bridging conditions between signal lines. When an anomaly is identified, the system traces the fault to its root cause by analyzing pin locations**,** involved ICs**,** and the associated net connections. Defective points are marked—typically in red—within the Truth Table Report (TTR), providing immediate visual feedback for debugging. This diagnostic output is illustrated in Figure 4.

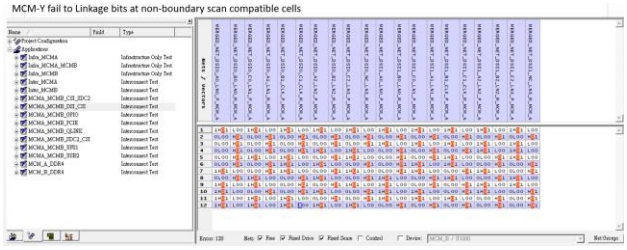


Fig. 4. Inadequate linkage bits and unwanted I2C bus detected.

The testing framework integrates two MCUs (MCU-x and MCU-y), evaluated across three operational modes: Connection Test Mode (CTM) for interconnect integrity, debug mode for at-speed functional verification, and emulation mode for simulating real-world communication scenarios.

For fully coherent processors, such as Marvell's CN9130 (MCU-x) and Infineon's XMC413x (MCU-y), the ARM AMBA CHI specification defines the protocol and transport layers [13], facilitating modular and scalable system design. The expanding ecosystem of open-source ARM-based MCUs has further propelled the integration of the JTAG-based Wishbone bus architectures to improve Design-for-Test (DfT) capabilities. The introduction of ARM Flexible Access has accelerated the adoption of a broad range of ARM-based IP cores that are compliant with JTAG standards, simplifying the integration of these cores into testable chiplets and MCM systems [15], as follows in Figure 5.

- JTAG Component**:** Essential for infrastructure testing, particularly in bare-die chiplets where traditional test points may be limited.
- Wishbone Component: Used for validating core logic and embedded memory subsystems, providing a complementary functional test path to JTAG.

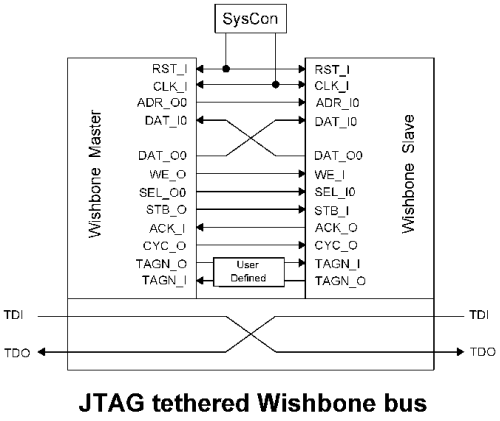


Fig. 5. JTAG-based daisy chain integrated with the Wishbone bus.

The JTAG component originated as a simple two-wire serial interface, using TDI (Test Data In) and TDO (Test Data Out) for data transmission. It was initially developed through the collaborative efforts of engineers from companies such as Philips, BT, GEC, Texas Instruments, among others. Over time, the interface was expanded to include essential control signals—TMS (Test Mode Select), TCK (Test Clock), and TRST (Test Reset)—leading to the formation of the Joint Test Action Group (JTAG) [5]. This collaborative initiative culminated in the creation of IEEE Standard 1149.1, also known as the Test Access Port (TAP) standard, which defines the boundary-scan architecture for integrated circuits. Compliance with this standard requires a valid Boundary Scan Description Language (BSDL) file. Without a valid BSDL file, an MCU or FPGA cannot be considered compliant with IEEE 1149.1 [5].

A) Connection Test Mode (CTM) – Approach 1

According to Micron Technologies, Connectivity Test (CT) Mode [13] functions similarly to boundary-scan testing, specifically optimized for rapid verification of electrical continuity between the memory device and the memory controller on a printed circuit board (PCB). In CT mode, the memory device operates synchronously with the external controller, enabling real-time assessment of data and control line integrity. While DDR4 memory systems [12] retain some architectural similarities with DDR3, several critical design updates in DDR4 have significantly influenced both PCB layout and signal behavior. These enhancements are reflected in the reference design shown in Figure 6.

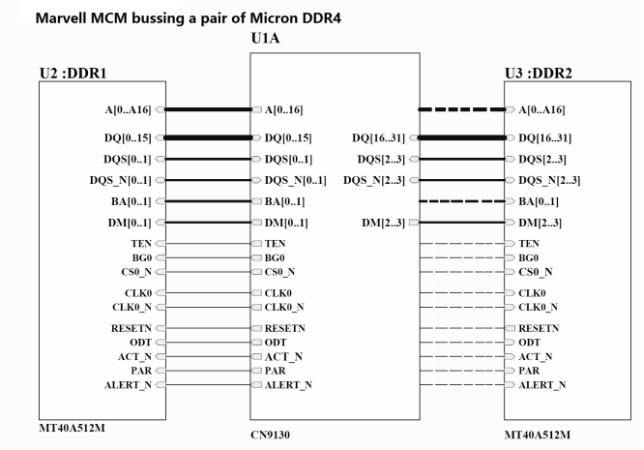


Fig. 6. Schematics of Marvell MCU targeting two Micron DDR4s.

If the drive signal registers a high logic level ("1") while the corresponding sense signal remains low (“0”), this logic mismatch may indicate the presence of an interconnection fault**.** Such discrepancies often result from issues like incorrect test vector sequencing, signal path degradation, or faulty pin connectivity. This detection technique is particularly effective for diagnosing a variety of physical defects, including open circuits, shorts, and bridging faults across pins and nets. The diagnostic result is visually represented in Figure 7.

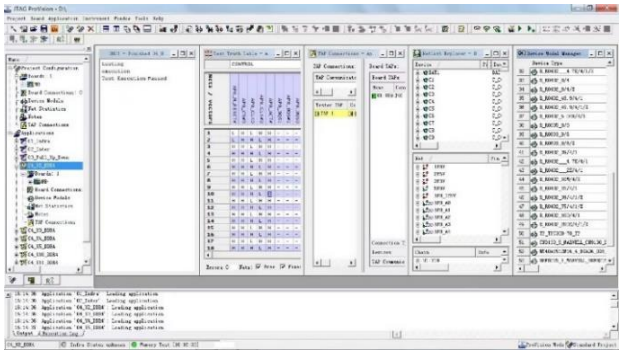


Fig. 7. CTM-based testing of 2 ×DDR4 using the Provision System.

The following Truth Tables define the test vectors used for two DDR4 chips, based on the netlist connecting MCU-x to a dual DDR4 configuration. This automated interconnection testing approach is especially effective for system-level validation, particularly in scenarios where an MCU interfaces with DDR memory. In this case study, both DDR4 devices are implemented in bare-die form, underscoring the need for robust interconnection testing structures and careful test-path scheduling through silicon interposers. These interposers, fabricated from either silicon or organic substrates, act as high-speed electrical conduits in 2.5D integration schemes, enabling efficient and reliable signal transmission between heterogeneous components. The interconnection test verifies correct electrical connectivity among DDR-a, DDR-b, and the MCU. When no faults are detected, the Provision system confirms the interconnect integrity as error-free. The corresponding test results are presented in Figures 8 and 9.

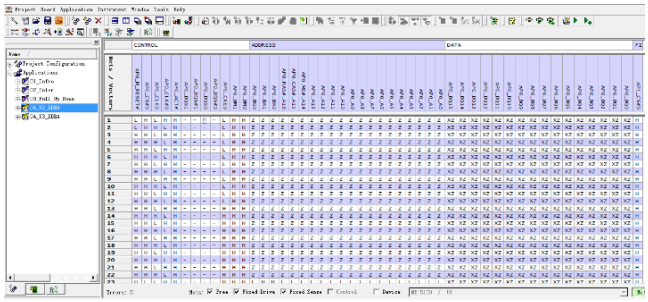


Fig. 8. DDR4-a tested using the CTM-based interconnection test.

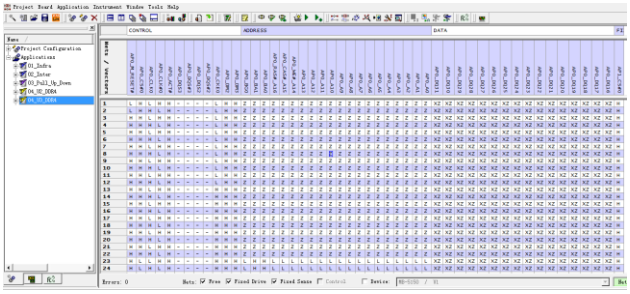
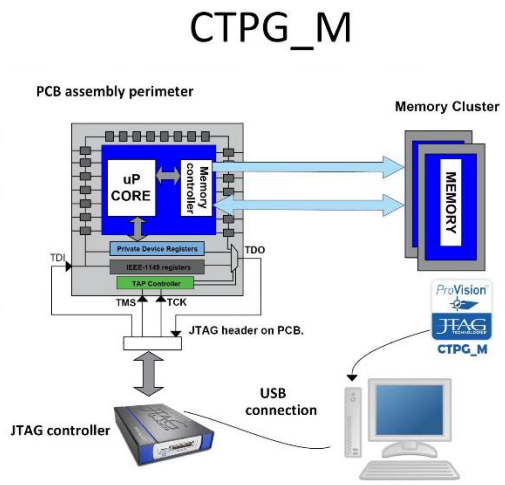


Fig. 9. DDR4-b tested using the CTM-based interconnection test.

B) Debug Mode Testing – Approach 2

JTAG Technologies developed the Core-Commander Test Program Generator (CTPG_M) as an integral component of the Provision platform. CTPG_M enables the automated generation of test programs for accessing and debugging embedded processor cores, streamlining functional verification in complex SoC and MCM environments. The following illustration demonstrates the operational flow of CTPG_M within this debug environment, as depicted in picture 1 [15].



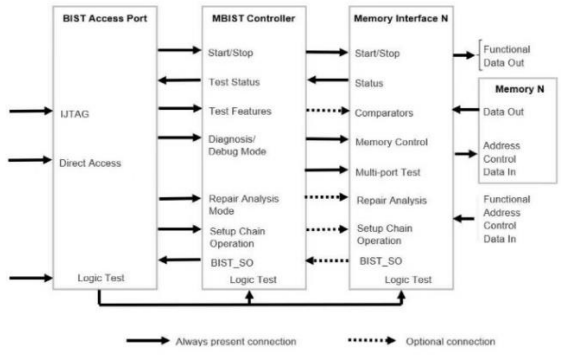
Picture 1. Test Program Generator (CTPG_M)

JTAG Technologies' Emulative Test & Programming (ETP) technology provides access to an MCU's emulation modes or an FPGA's internal IP bus, enabling advanced functional test and programming capabilities. Within the JTAG Provision environment, the Core-Commander options can be used either interactively or via scripted command sequences, offering flexibility for various testing workflows. This approach is especially valuable for conducting at-speed connectivity tests on memory devices in situations where:

- A boundary-scan register is not present, limiting structural test access
- Full signal-line visibility to the memory device is constrained
- Conventional write/read cycle-based test methods are infeasible due to timing or design limitations

By leveraging embedded emulation and debug logic in conjunction with the on-chip memory controller, this method provides direct access to memory and peripheral devices for at-speed testing. Once the starting address of the memory is defined, CTPG_M automatically generates the required command sequences for Core-Commander, allowing efficient debugging through scripted API control.

In the context of chiplet-based Multi-chip Modules (MCMs), this strategy offers a distinct advantage over traditional Memory Built-In Self-Test (MBIST) approaches typically used in System-on-Chip (SoC) design and verification. While Serial BIST techniques have been effective for monolithic-memory testing, they encounter challenges when applied to heterogeneous designs featuring multiple embedded RAM blocks with varying sizes, access widths**,** and port configurations. This limitation is particularly evident in modern telecom ICs**,** which often incorporate multiport memories on a single die. Such complexity reduces the effectiveness of conventional M-BIST methods, as discussed in the work of Neil H.E. Weste and David Harris, and illustrated in Picture 2.



Picture 2. SoC-based Memory Built-in Self-Test (BIST)

This testing process integrates key components, e.g., algorithms, clock signals, address and data generators, and read/write control logic, to formulate test instructions and evaluate the memory controller's responses. Execution is accomplished by downloading and running an SVF-based file (e.g., "*chain-ddr-test.svf"*), which is written into the MCU's internal registers [14] as depicted in Figure 10.

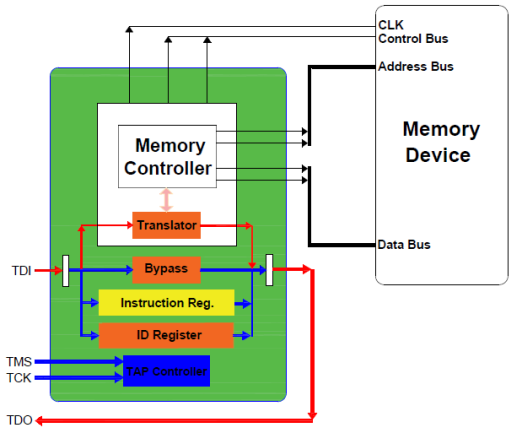


Fig. 10. Debug logic-based testing of DDR-X at speed using MCU-X.

The Serial Vector Format (SVF) is a vendor-neutral, ASCII-based file format used to deliver boundary-scan test vectors to electronic circuits through the JTAG interface. An SVF file consists of a sequence of structured test instructions and optional comment lines, providing precise control of JTAG operations, including signal toggling, scan chain configuration, and data verification. Depending on system architecture, processors may utilize on-chip memory for faster access and tighter integration, or off-chip (external) memory to support larger capacity and flexibility. These memory classifications, based on functional roles, are essential in both computing and embedded systems. Their integration and validation are demonstrated through the JTAG Function Test (JFT) workflow, as illustrated in Figure [14].

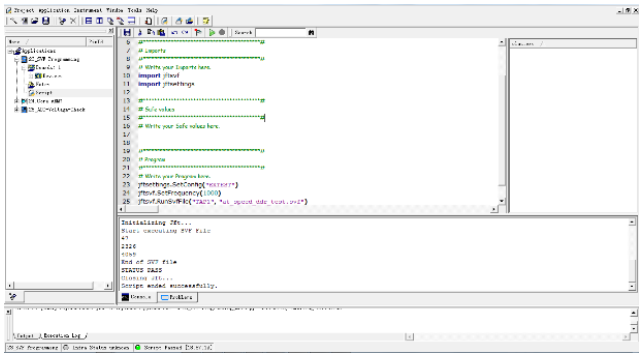


Fig. 11. SVF program applied for DDR-x at-speed testing.

The JTAG Functional Test System (JFT) is a DLL-based API that provides a suite of software modules for developing boundary-scan tests and performing in-system device programming. Test scripts are commonly authored in Python, enabling flexible and automated control over JTAG operations, as illustrated in Figure 12.

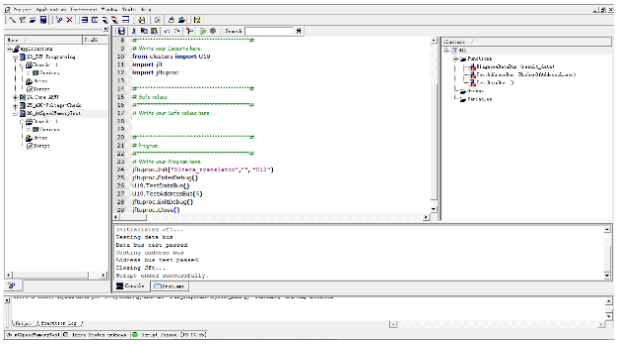


Fig. 12. Scripted Tests: to write SVF by JFT in debug mode

C) Emulation Mode Testing- Approach 3

The JTAG-based Functional Test (JFT) framework enables access to core logic processing at GPIO ports, facilitating instruction execution and real-time data transmission across the Wishbone bus. In this configuration, MCU-x functions as the command initiator**,** while MCU-y transmits data via shared GPIO pins, enabling coordinated communication between the two devices over the Wishbone bus. This case study highlights the functional control of LEDs—specifically (D1, D2, D3)—which are connected to pins F14, E22, and B2, respectively. The operational control of these LEDs is orchestrated by MCU-x, demonstrating the effectiveness of this emulation-based testing approach. The corresponding logic and pin mappings are detailed in Table 2.

Device MCU	Pin	Color	LED
MCU-X	F14	Red	D1
MCU-X	E22	Green	D2
MCU-X	B22	Blue	D3

Table 2. Emulation mode-based LED function test

The JTAG Functional Test (JFT) system controls LEDs using a DLL-based API, which provides essential software modules for boundary-scan test development and in-system device programming. Test scripts are typically written in Python. The table below maps the relevant Wishbone bus signals from two MCUs to their associated GPIO ports, which are utilized to test the operation of three LEDs [9]:

- P3-5 → G1 (Pin V22)
- P3-6 → J1 (Pin T18)
- P3-7 → L1 (Pin U20)

The JTAG Functional Test (JFT) is primarily used for validating logic devices or mixed-signal clusters. It can also be extended into reusable test modules by leveraging embedded peripherals, such as ADCs, DACs, and memory controllers. Reliable execution of this configuration depends on accurate handshaking protocols within the Wishbone-based interconnect, as detailed in Table 3.

P17	P2_27	P3	P2_27
M21	P2_28	P4	P2_28
N17	P2_29	N3	P2_29
M16	P2_30	L4	P2_30
U19	P2_31	N2	P2_31
		B4	P3_0
		B3	P3_1
		B1	P3_2
		E4	P3_3
		F2	P3_4
V22	**P3_5**	G1	**P3_5**
T18	**P3_6**	J1	**P3_6**
U20	**P3_7**	L1	**P3_7**
		D8	P3_8
		C5	P3_9
		B2	P3_10
		D5	P3_11
		D4	P3_12
		C1	P3_13
U21	P3_14	H2	P3_14
V21	P3_15	M1	P3_15

Table 3. Wishbone buses are shared between two MCUs via GPIO ports.

This case study utilizes MCU-x GPIO ports—P3-5, P3-6, and P3-7 (as specified in the respective datasheets)—to emulate LED functionality using switches SW2, SW3, and SW4. These switches control LED brightness levels by interacting with the programmed logic on the MCU. The system writes 32-bit binary data into specific functional memory blocks located at predefined memory addresses. It then monitors the resulting behavior via the PIO2-5, PIO2-6, and PIO2-7, which serve as the corresponding monitoring function blocks. The complete signal mapping and data paths are presented in Table 4 [16].

Address Location	32 bits DATA							
3FFF C054	F	F	F	F	F	F	7 B D	F
	1 1 1 1 1 1 1 1	1 1 1 1 1 1 1 1	1 1 1 1 1 1 1 1	1 1 1 1 1 1 1 1	1 1 1 1 1 1 1 1	1 1 1 1 1 1 1 1	0 1 1 1	1 1 1 1 1 1 1 1
Function Block							PIO2-7	PIO2-6 PIO2-5
Behavior with lead through configured in MCU-Y							SW4	SW3 SW2

Table 4. Memory map of function blocks

The selected functions regulate LED behavior by updating pin counters within the corresponding registers and latching these values to maintain state, as demonstrated in the scripted test sequence illustrated in Figure 13.

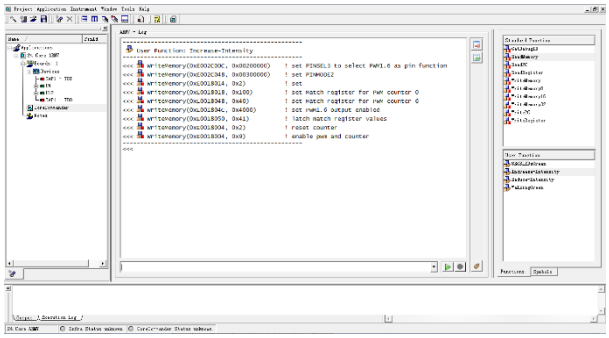


Fig. 13. Writing values to registers at respective function blocks.

II CONCLUSION

This paper introduces two cutting-edge testing methodologies and three diagnostic techniques that automate boundary–scan–based fault detection, significantly improving chiplet-level diagnostics. These innovations effectively address key limitations in 3D Multi-Chip Packaging (MCP) and heterogeneous chiplet integration [24]. The proposed solutions offer modular, scalable, and reusable testing strategies, particularly well-suited for automotive electronics, where reliability, safety [17], and rapid deployment are critical. Design-for-Test (DfT) techniques are now being tailored for chiplet-based Multi-Chip Module (MCM) applications, including anti-lock braking systems (ABS), airbags, and in-vehicle communication modules. Engine Control Units (ECUs), power steering, and chassis subsystems are already benefiting from these advances [18].

Leading Japanese automotive manufacturers [20]—such as Toyota and Honda—alongside technology partners like Denso and Renesas Electronics, are working to standardize chiplet communication within vehicles via the Advanced SoC Research for Automotive (ASRA) group [21]. They have submitted proposals to the Universal Chiplet Interconnect Express (UCIe) Consortium [23] to formalize communication standards. Renesas, for its part, has unveiled plans for next-generation System-on-Chips (SoCs) and microcontrollers (MCUs) targeting the full spectrum of automotive digital applications. Its fifth-generation R-Car SoC lineup will incorporate advanced "in-package chiplet integration", allowing custom AI accelerators [22], memory, and I/O components to be scalable in a single package [19].

Monolithic IC designs are reaching physical and economic limits, making chiplet-based modular architectures essential for integrating greater transistor counts and boosting performance. These architectures support process specialization and shared infrastructure, like power management modules. As Moore's Law slows, chiplet and MCM approaches offer a practical pathway to sustaining innovation in high-performance and safety-critical automotive computing [25].

Acknowledgment

The authors gratefully acknowledge JTAG Technologies for providing the essential hardware and software tools that made this research possible. Special thanks to the engineering teams involved in boundary-scan deployment and automotive chiplet research, whose insights and tools were critical to this work.

REFERENCES

[1] Wang Shun Shen Peter, etc., "Wafer Scale Burn-in Testing", U.S. Patent No. 6,121,065 assigned to IME, University of Singapore

[2] Wang, Shun Shen Peter, "Method of making a High-Density Multi-Layer Wiring Board", U.S. Patent No. 6,026,564.

[3] Wang, Shun Shen Peter, "Method of and an arrangement for analyzing manufacturing defects of multi-chip modules made without known good die', U.S. Patent No. US 2021/0116497 A1

[4] Wang Shun Shen Peter, etc., "Instrumentation of TWIN-MCM based Mutual Test", published by "Microelectronics Journal", Vol. 114 on August 21, 2021, http://doi.org.j.mejo.2021.105108.

[5] Boundary Scan Test: A Practical Approach, by Harry Bleeker and Peter van den Eijnden, published by Eluwer Academic, Netherlands, @1993.

[6] Boundary-Scan Solutions towards MCM and Chiplets, by Peter van den Eijnden and Peter Shun Shen Wang, Published by JTAG Technologies, ISBN 978-626-01-1751-1, Library of Congress Control Number: 2023918565

[7] Wang Shun Shen Peter, etc., "Thermal Design of an MCM with the Implanted Temperature Sensor", published by Electrical and Electronic Engineering (IEEE), published on Feb. 25, 2022. (ISSN Print: 2329-1613; ISSN On line: 2329-1605).

[8] IEEE Standard 1149.1-1990: IEEE Standard Test Access Port and Boundary-Scan Architecture. Published by the Institute of Electrical and Electronics Engineers, Inc., 345 East 47th Street, New York, NY 10017.USA.

[9] Altera®, IEEE1149.1 JTAG Boundary Scan Testing application notes, June 2005 Ver. 6.0

[10] Parker and S. Oresjo: A Language for Describing Boundary-Scan Devices. Journal of Electronic Testing: Theory and Applications, Vol. 2, No. 1, March 1991, pp. 43-74.

[11] Wang, Shun Shen Peter, "Switch-Mode based interposer Enabling Self-Testing of an MCM without Known-Good-Die, U.S. Patent Pending No. 17240956.

[12] Micron®, DDR4 SDRAM MT40A2G4 MT40A1G8 MT40A512M16

[13] Marvell® OCTEON TX2™ CN913X Four Core Arm®v8 Multi-Core SoC for Intelligent Networking, Security, control plane, and Edge Computing

[14] NXP® MCU Cortex ARM 7 core LPC2468FET208

[15] Altera® FPGA Cyclone III Cyclone III EP3C16F484C8N
Lattice® PLD: ISPLSI2064VE

[16] A new problem at Boundary-Scan testing: an internal disruption within the IC during interconnect testing. 電子情報通信学会技術研究報告. DC. ディペンダブルコンピューティング: IEICE technical report

[17] KAMEYAMA Shuichi, BABA Masayuki, HIGAMI Yoshinobu, TAKAHASHI Hiroshi Built-In Self-Diagnosis for Functional Safety Assurance Grants-in-Aid for Scientific Research, Grant-in-Aid for Scientific Research (C), Japan Society for the Promotion of Science, Apr, 2016 - Mar, 2019

[18] Chiplets- How New Semiconductor Architectures are Reshaping the Competitive Integration, Reina Ogawa, Industry Innovation Dept. Ryusuke Ishiguro (IP Analysis) Intellectual Property Dept. Technology & Innovation Studies Div. Mitsui & Co. Global Strategic Studies Institute

[19] Institute of Science Tokyo- Chiplet integration technology with the simplest scheme, Scalability of inter-chip bandwidth, and integration scale

[20] Major Japanese push for automotive chiplets for 2028. Business news December 28, 2023. By Nick Flaherty

[21] Challenges and Prospects of Chiplet Package Design, January 2025 Journal of The Japan Institute of Electronics Packaging 28(1):48-57 DOI:10.5104/jiep.28.48

[22] Japan charts a course to EV chiplets, Telcos &AI, Apr. 28, 2025 by Ian Scales

[23] The Rise of Chiplets and Simplified Interconnectivity, IDTechEx, Mar 10, 2025, by Lily-Rose Schuett

[24] APCS 2024 Conference Programs in Advanced Semiconductor and 3D Packaging

[25] IMEC, why are chiplets attracting the attention of the automotive industry?