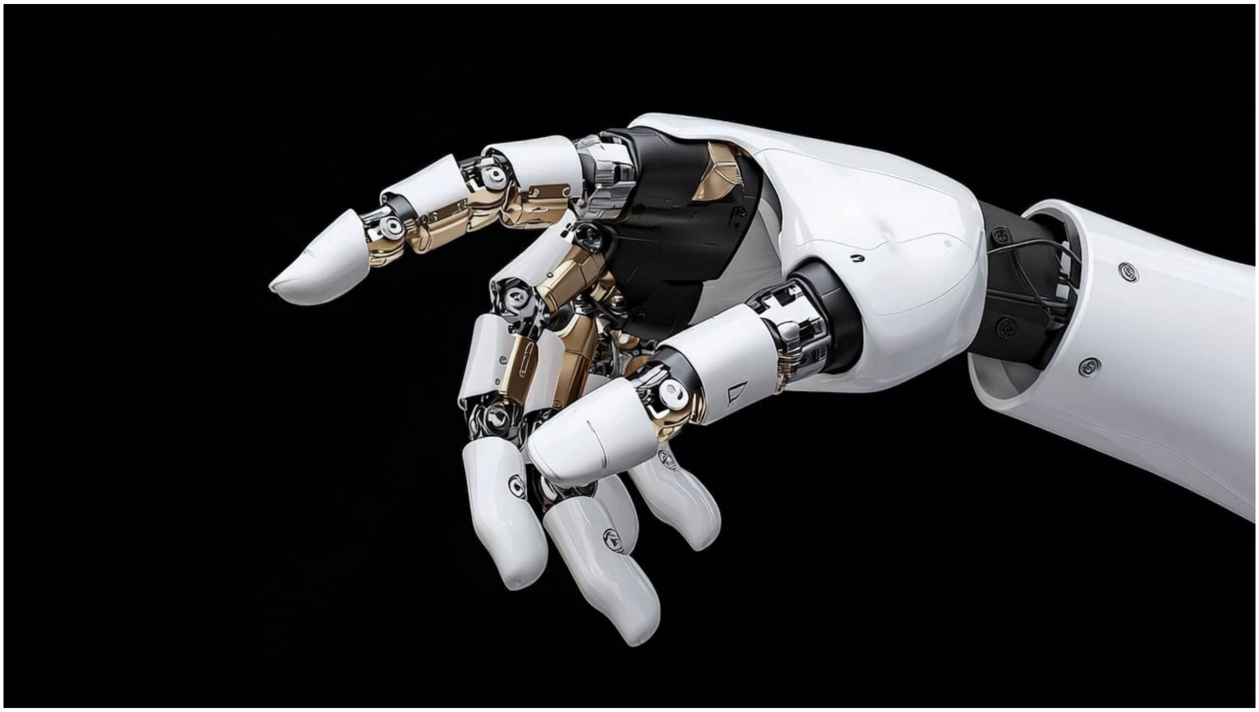
Humanoid Manipulation at the Edge of Physical Interaction

Architecture Shifts in Joints and Dexterous Hands

Lei Jin, *Senior Staff Engineer, TAA/UX*

Yahui Liu, *Principal Expert, SOL/UX*



Abstract

As humanoid robots move from laboratory demonstrations toward practical physical work, subsystem design at the point of physical interaction is becoming a new source of competitive differentiation. Joints and dexterous hands are no longer defined only by actuation, but increasingly by how they combine deterministic control, communication, sensing, and selective local intelligence within severe constraints on size, thermal budget, reliability, and cost. This shift is reshaping humanoid manipulation architecture and creating clear opportunities for solution platforms spanning motor control, industrial networking, sensing, and edge processing.

Humanoid Manipulation Is Becoming a New System Boundary

Humanoid robotics is moving beyond proof-of-concept demonstration toward real-world deployment, where the key question is no longer what these systems can show, but what they can deliver reliably, economically, and at scale. In this transition, more differentiation is now happening at the point of physical interaction—especially in joints and dexterous hands—because practical value depends not only on centralized intelligence, but on the ability to generate controlled motion, stable contact, and reliable object handling under severe constraints on size, power, and integration [1].

Recent research points in the same direction. Dexterous manipulation increasingly depends on tactile–visual fusion, because stable grasping requires access to pressure, texture, temperature, and slip information directly at the point of contact [2]. The same shift applies to humanoid joints, where torque awareness, multimodal sensing, and low-latency estimation are increasingly required to support precise control and safe interaction under dynamic conditions [3].

Taken together, these developments change how joints and dexterous hands should be understood. They are no longer just downstream actuator implementations. They are increasingly distributed mechatronic subsystems in which actuation, sensing, and low-latency local response must be coordinated close to the point of interaction.

Joints as Deterministic Servo Nodes

Humanoid joints are where subsystem architecture becomes product reality. They are no longer just mechanical pivots driven by a remote controller, but compact servo nodes positioned directly on the path from computation to actuation. Each joint must carry load, stabilize balance, absorb disturbance, and translate control decisions into smooth and reliable motion under severe constraints on size, weight, thermal budget, and cost. Deterministic control is therefore a defining property, because fast current regulation, stable position and velocity control, torque response, and disturbance handling must remain local and tightly timed if the robot is to react safely and predictably in the physical world [3][4].

The communication role of the joint is evolving in the same direction. Legacy links such as RS485 and traditional CAN are giving way to more time-sensitive options, including EtherCAT and CAN-FD combined with time synchronization. In more advanced architectures, customers are also showing interest in TSN-capable approaches. A joint is no longer expected merely to execute basic drive commands; it must increasingly operate as a synchronized and network-aware servo node within a larger distributed robotic system. Synchronization matters because deterministic motion depends not only on fast local loops, but also on predictable timing across multiple joints [1].

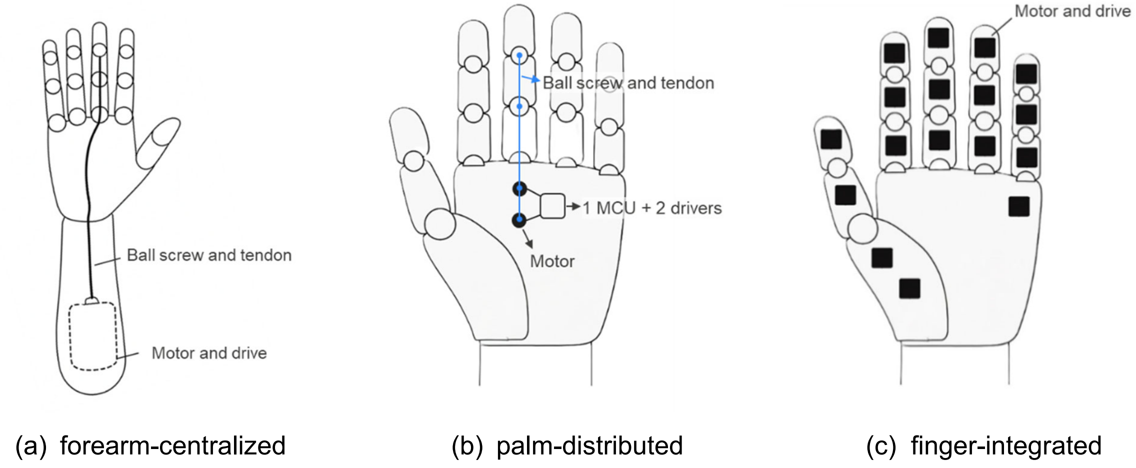
Deterministic control also depends on deterministic state awareness. Position feedback remains fundamental, but it is no longer sufficient on its own. A humanoid joint increasingly needs torque-related observation, current monitoring, temperature awareness, and contact-related estimation so that local control can respond predictably to load variation, external disturbance, and thermal limits [3][4]. Recent research shows that multimodal measurements such as motor currents, accelerations, and external-contact effects can be fused to estimate joint torque online, producing state information that is directly usable in humanoid control architectures even when dedicated torque sensors are absent [3].

Taken together, a humanoid joint should no longer be understood as a simple embedded actuator. It is better understood as a deterministic servo node: a compact mechatronic subsystem that executes motion locally, participates in synchronized networked control, and maintains enough real-time awareness of its own state to support stable and reliable physical interaction.

Dexterous Hands as Integrated Control-and-Perception Subsystems

If the joint is the first mature deterministic node, the dexterous hand is where subsystem integration becomes far more challenging. A hand must deliver more degrees of freedom in less space, while also accommodating a broader sensing mix and, in some designs, local visual perception.

Three representative actuation topologies are shown in Figure 1. In a “forearm-centralized” architecture, motors are placed upstream and motion is transferred mechanically to the fingers. In a “palm-distributed” architecture, multiple motors and drives are packaged inside the palm. In a “finger-integrated” architecture, actuation and drive electronics move closer to, or into, the fingers themselves. Each topology reflects a different trade-off. Forearm-centralized designs generally favor serviceability, thermal management, and cost, but often at the expense of wiring and transmission complexity, achievable dexterity, and control precision at the fingertips. Palm-distributed designs offer a more practical balance between dexterity and integration complexity. Finger-integrated designs provide the strongest path toward high degree of freedom and anthropomorphic motion, but also place the greatest pressure on package size, wiring density, manufacturability, and thermal handling.

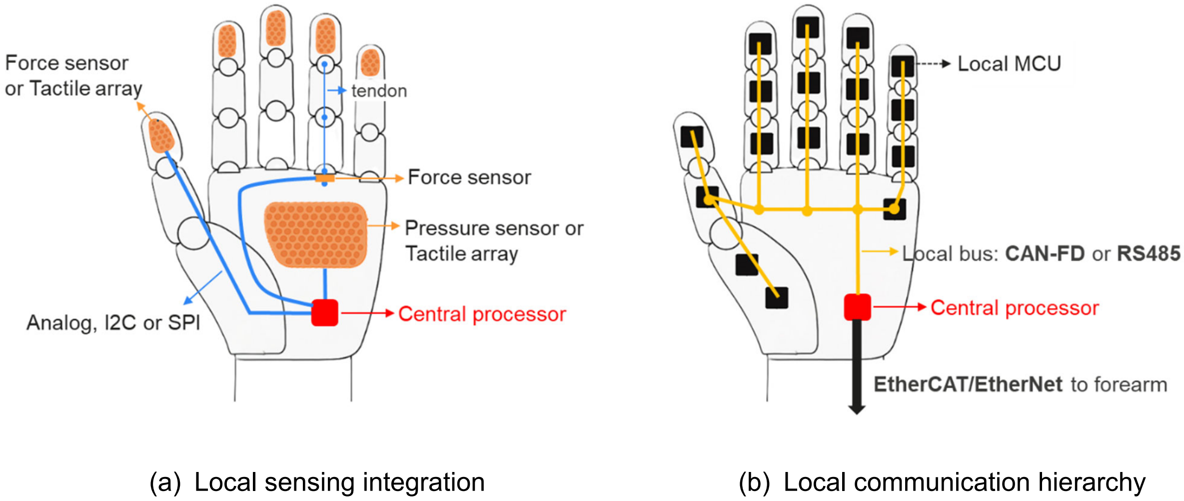


(a) forearm-centralized (b) palm-distributed (c) finger-integrated

Figure 1. Three representative dexterous-hand actuation topologies.

The same integration pressure appears in the sensing stack. Even without local vision, a dexterous hand may already need to combine encoder feedback, force sensing, touch sensing, tactile arrays, and actuator-related status signals across a space-constrained structure. This is not just a matter of adding more sensors. It changes routing, packaging, calibration, signal integrity, and local data aggregation inside the hand. As shown in Figure 2(a), these non-vision sensing signals are typically gathered at a central processor through I2C, SPI or an analog interface.

As actuation and sensing density increase with the degree of freedom, communication inside the hand also becomes layered. As shown in Figure 2(b), each finger-level actuator with a local MCU handles local motor control, while the central processor serves as the aggregation and coordination hub for the hand. From there, upstream connectivity links the hand to the forearm or main controller. In practice, the local coordination typically adopts RS485 or CAN-FD, while upstream integration shifts toward EtherCAT or Ethernet-class networking. The dexterous hand is therefore not only a dense electromechanical assembly, but also a subsystem with an internal communication hierarchy of its own.

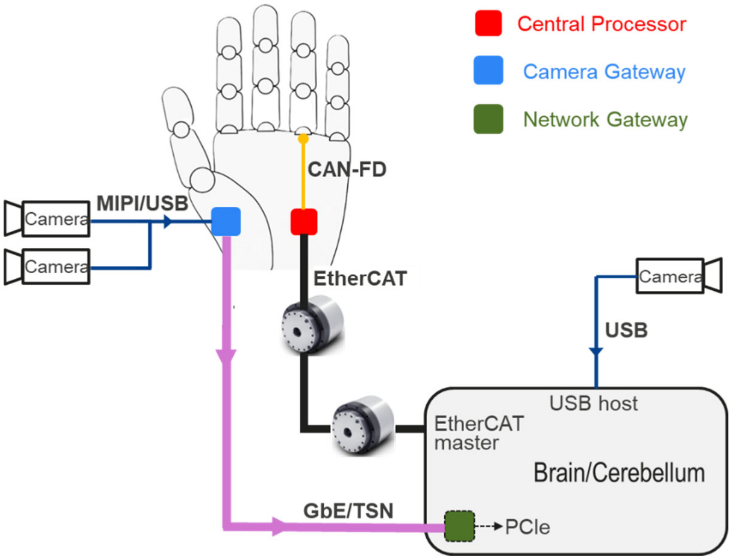


(a) Local sensing integration

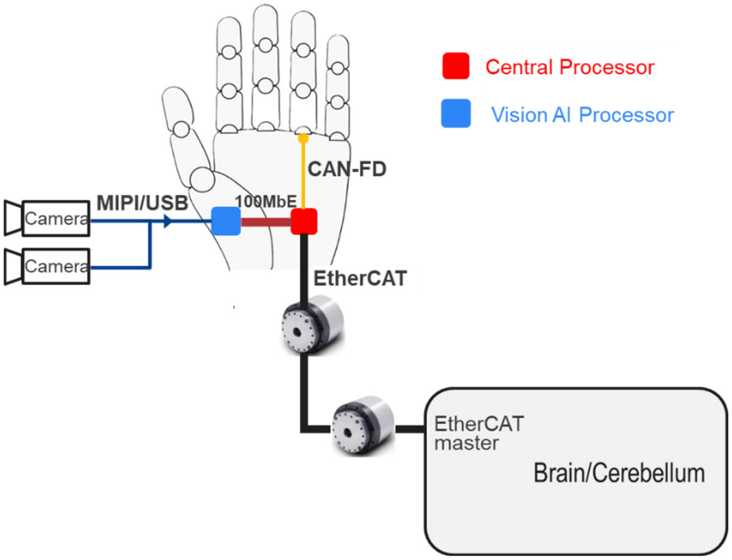
(b) Local communication hierarchy

Figure 2. Local subsystem integration inside the dexterous hand.

A further architectural shift appears when hand-mounted cameras are introduced, because now designers must determine how to handle high-bandwidth visual data streams alongside control-critical traffic and the dense local sensing traffic already present inside the hand. This creates a new partitioning problem. In a centralized architecture shown in Figure 3(a), most visual data are sent upstream to a higher-level compute domain for interpretation. In some cases, a camera gateway can be used to convert local camera interfaces such as MIPI or USB to Gigabit Ethernet (GbE) for better EMC performance and easier upstream integration. In a decentralized architecture shown in Figure 3(b), selected visual data are filtered, compressed, or partially interpreted closer to the hand by a vision AI processor before transmission. As a result, the upstream bandwidth requirement can be reduced significantly, and in some designs the remaining data flow can be merged into the central processor through a 100 Mbps Ethernet link. The choice between these architectures leads to different trade-offs in communication topology, memory movement, latency, cost, and subsystem complexity. Taken together, the dexterous hand should no longer be understood as a simple downstream actuator cluster. It is better understood as an integrated control-and-perception subsystem, where packaging, communication, sensing, and processing must be co-designed at the edge of physical interaction.



(a) Centralized architecture



(a) Decentralized architecture

Figure 3. Centralized vs decentralized processing partition in dexterous-hand architectures.

What This Shift Means for System Design

The move from centralized robotics toward distributed manipulation subsystems is already changing practical system design. The first consequence is that deterministic control and richer data flows can no longer be treated as the same architectural problem. Control-loop execution, fault response, and synchronized behavior depend on tightly timed local control paths, while dense tactile, force, and visual sensing introduce very different requirements in bandwidth, buffering, and transport. As a result, humanoid architectures increasingly need a clearer separation between control-critical traffic and data-rich sensing or perception flows, rather than forcing both into a single communication and compute model.

The second consequence is a new balance between centralized compute and local intelligence. Centralized processing remains attractive because it simplifies subsystem design and enables global fusion, but that advantage weakens as local sensing becomes richer and communication pressure increases. Designers must therefore decide which functions need to remain close to the point of interaction for timing, robustness, or data-efficiency reasons, and which functions benefit from being handled upstream. In joints, that usually favors local real-time servo execution and state-aware response. In dexterous hands, it often leads to palm-level aggregation, layered communication, and more selective decisions about what should be processed locally and what should be sent onward.

The third consequence is that miniaturization is no longer just a packaging challenge. It is a system constraint that shapes architecture. Smaller modules increase pressure on thermal dissipation, power delivery, EMI control, wiring density, manufacturability, serviceability, and reliability. They also make conventional functional-safety partitioning harder to transfer directly into highly compact joints and dexterous-hand nodes. For this reason, the next generation of humanoid manipulation systems will be defined not only by better algorithms, but by better partitioning of control, communication, sensing, and local processing under real physical constraints.

Implications for Renesas

These shifts create several clear alignment points for Renesas. For humanoid joints, Renesas aligns most clearly with the deterministic servo-node layer: RA8T2 and the RZ/T series address real-time motor control and industrial Ethernet with TSN capability, while RAA2P3226 and the ZSSC32xx family strengthen the sensing side through inductive position sensing (IPS) and resistive bridge-sensor signal conditioning for torque and force feedback [5]-[8]. In particular, IPS offers a compact, magnet-free alternative to magnetic encoders, with strong stray-field immunity and accuracy levels suited to high-performance robotic actuation. Beyond the component level, Renesas is increasingly presenting subsystem-oriented solutions and reference designs rather than only a collection of discrete devices [9].

In dexterous hands, a practical subsystem stack can be framed around RA6T2 and HVPAK for compact actuation-side integration, ZSSC32xx or RAA2S42xx family for force-sensor signal conditioning, and RZ/T2 or RA8T2 for the communication and control hub [5]-[8], [10], [11]. Beyond the component level, public solutions are already visible for micro-ROS implementation on Renesas MCUs and for hand-side vision AI on RZ/V2H [13]-[15], showing that Renesas is already extending beyond traditional embedded control into vision processing and the robotics software ecosystem around ROS 2 and micro-ROS. Additional opportunities may also emerge in impedance-based human-contact detection and touch-aware interaction sensing, where devices such as RAA2S4704 [16] point to a possible future path.

Looking ahead, smaller-package processors and tighter subsystem-level integration could further strengthen Renesas' fit with emerging humanoid architectures. As control electronics move deeper into increasingly space-constrained subsystem nodes, combinations such as GaN/MOSFET plus pre-driver, MCU plus PHY, or more compact control-and-power tiles integrating MCU, pre-driver, and MOSFET become increasingly relevant. These directions would further improve Renesas' alignment with the next stage of humanoid subsystem integration.

Summary

Humanoid robotics is increasingly being defined at the edge of physical interaction. In that shift, joints and dexterous hands should no longer be viewed as downstream actuator assemblies, but as subsystem boundaries where control, communication, sensing, and selective local processing must be co-designed under real physical constraints. Joints are becoming deterministic servo nodes. Dexterous hands are becoming integrated control-and-perception subsystems. This is why the next stage of humanoid manipulation will be shaped not only by better algorithms, but by better subsystem partitioning and integration. Renesas is already aligned with several parts of this transition. The next opportunity is to turn that component strength into more compact, deployable subsystem solutions for emerging humanoid architectures.

Glossary

CAN — Controller Area Network; a widely used fieldbus for embedded control communication.

CAN-FD — Controller Area Network with Flexible Data-Rate; an extended CAN standard with higher data payload and data rate than classical CAN.

RS485 — A differential serial communication standard widely used for robust industrial device interconnection over relatively long cables.

EtherCAT — Ethernet for Control Automation Technology; a deterministic industrial Ethernet protocol widely used for synchronized multi-axis motion control.

TSN — Time-Sensitive Networking; a set of Ethernet standards for bounded latency, traffic scheduling, and time synchronization.

I2C — Inter-Integrated Circuit; a two-wire serial interface commonly used for short-range communication with sensors and peripheral devices.

SPI — Serial Peripheral Interface; a synchronous serial interface commonly used for high-speed local communication between processors and peripheral devices.

MCU — Microcontroller Unit; a highly integrated processor device typically used for embedded control tasks.

MPU — Microprocessor Unit; a higher-performance processor device typically used where more compute capability, memory, or software complexity is required.

PHY — Physical-layer transceiver; the hardware interface that implements the physical signaling layer of a communication link such as Ethernet.

ROS 2 — Robot Operating System 2; a widely used robotics software framework for modular communication, coordination, and software integration.

micro-ROS — A microcontroller-oriented implementation of ROS 2 concepts for resource-constrained embedded nodes.

GbE — Gigabit Ethernet; Ethernet communication operating at 1 Gbit/s.

MIPI — Mobile Industry Processor Interface; a family of high-speed chip-to-chip interfaces widely used for camera and display connections.

USB — Universal Serial Bus; a widely used serial interface for device connectivity and data transfer.

EMI — Electromagnetic Interference; unwanted electromagnetic disturbance that can degrade signal integrity or disrupt circuit and system operation.

EMC — Electromagnetic Compatibility; the ability of a device or system to operate correctly in its electromagnetic environment without causing or suffering unacceptable interference.

References

[1] McKinsey & Company, “Turning humanoid supply chain constraints into billion-dollar wins,” April 17, 2026. Available: https://www.mckinsey.com/industries/industrials/our-insights/turning-humanoid-supply-chain-constraints-into-billion-dollar-wins

[2] Q. Mao, Z. Liao, J. Yuan, and R. Zhu, “Multimodal tactile sensing fused with vision for dexterous robotic housekeeping,” Nature Communications, vol. 15, Article 6871, Aug. 11, 2024. Available: https://www.nature.com/articles/s41467-024-51261-5

[3] I. Sorrentino, G. Romualdi, and D. Pucci, “UKF-Based Sensor Fusion for Joint-Torque Sensorless Humanoid Robots,” arXiv:2402.18380 / ICRA 2024. Available: https://arxiv.org/abs/2402.18380

[4] C. Semasinghe, D. Taylor, and S. Rezazadeh, “Design of Actuators for a Humanoid Robot with Anthropomorphic Characteristics and Running Capability,” Actuators, vol. 14, no. 5, 243, May 2025. Available: https://www.mdpi.com/2076-0825/14/5/243

[5] Renesas Electronics, “RA8T2 – 1GHz Arm Cortex-M85 Microcontroller Optimized for Motor Control Applications,” product page. Available: https://www.renesas.com/en/products/ra8t2

[6] Renesas Electronics, “RZ/T Series MPU,” product family page. Available: https://www.renesas.com/en/products/microcontrollers-microprocessors/rz-mpus/rzt-series-mpu

[7] Renesas Electronics, “RAA2P3226 – High-Speed Inductive Position Sensor with Dual Coil Sensing Element Featuring I2C, ABI, Step/Dir, and UART Interfaces,” product page. Available: https://www.renesas.com/en/products/raa2p3226

[8] Renesas Electronics, “Sensor Signal Conditioners (SSC/AFE),” product family page. Available: https://www.renesas.com/en/products/sensor-products/sensor-signal-conditioners-ssc-afe

[9] Renesas Electronics, “Robotics Solutions – AI Vision & Precision Motor Control,” application page. Available: https://www.renesas.com/en/applications/industrial/robotics

[10]Renesas Electronics, “RA6T2 – 240MHz Arm Cortex-M33 TrustZone, High Real-time Engine for Motor Control,” product page. Available: https://www.renesas.com/en/products/ra6t2

[11]Renesas Electronics, “HVPAK: High Voltage Programmable Mixed-Signal Matrix,” product family page. Available: https://www.renesas.com/en/products/programmable-logic/greenpak-programmable-mixed-signal-products/hvpak

[12]Renesas Electronics, “RZ/N2L – Integrated TSN-Compliant 3-Port Gigabit Ethernet Switch Enables Various Industrial Applications to Easily Implement Industrial Ethernet and TSN,” product page. Available: https://www.renesas.com/en/products/rz-n2l

[13]Renesas Electronics, “How to control motors with micro-ROS and MCK-RA6T2 via CAN FD,” video. Available: https://www.renesas.com/en/video/how-control-motors-micro-ros-and-mck-ra6t2-can-fd

[14]Renesas Electronics, “RZ/V2H – Quad-core Vision AI MPU with DRP-AI3 Accelerator and High-Performance Real-time Processor,” product page. Available: https://www.renesas.com/en/products/rz-v2h

[15]Renesas Electronics, “Dexterous Hand,” application page. Available: https://www.renesas.com/en/applications/industrial/robotics/dexterous-hand

[16]Renesas Electronics, “RAA2S4704 – Multi-Channel Impedance Measurement IC,” product page. Available: https://www.renesas.com/en/products/raa2s4704

RENESAS ELECTRONICS CORPORATION AND ITS SUBSIDIARIES (“RENESAS”) PROVIDES TECHNICAL SPECIFICATIONS AND RELIABILITY DATA (INCLUDING DATASHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for developers skilled in the art designing with Renesas products. You are solely responsible for (1) selecting the appropriate products for your application, (2) designing, validating, and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, or other requirements. These resources are subject to change without notice. Renesas grants you permission to use these resources only for development of an application that uses Renesas products. Other reproduction or use of these resources is strictly prohibited. No license is granted to any other Renesas intellectual property or to any third party intellectual property. Renesas disclaims responsibility for, and you will fully indemnify Renesas and its representatives against, any claims, damages, costs, losses, or liabilities arising out of your use of these resources. Renesas' products are provided only subject to Renesas' Terms and Conditions of Sale or other applicable terms agreed to in writing. No use of any Renesas resources expands or otherwise alters any applicable warranties or warranty disclaimers for these products.

(Rev.1.0 Mar 2020)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu, Koto-ku, Tokyo 135-0061, Japan
https://www.renesas.com

Contact Information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit:
https://www.renesas.com/contact-us

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.

© 2026 Renesas Electronics Corporation. All rights reserved.

Doc Number: R30WP0016EU0100