

Die ARMv8-R Architektur erweitert die ARM® 32-bit Echtzeit-Prozessoren mit neuen Funktionen für den schnell wachsenden Markt sicherheitsrelevanter Applikationen beispielsweise in der Automobilindustrie oder im Industrieanlagenbau. Die wichtigste Innovation der ARMv8-R Architektur ist der „Bare-Metal“-Hypervisor-Modus, bei dem verschiedene Betriebssysteme sowie Software-Echtzeit-Aufgaben auf einem einzigen Prozessor ausgeführt werden, die einzelnen Teile dieser Software-Systeme aber durch Hardware isoliert sind und sich dadurch nicht beeinflussen können.

In der Automobilindustrie wird derzeit typischerweise ein ARM® Cortex®-A Prozessor im Infotainment-System sowie eine große Anzahl von Mikrocontrollern in diversen Echtzeit-Steuerungen verbaut. Die neue ARMv8-R Architektur erlaubt es, das Infotainment-System mit solchen Echtzeit-Steuerungen zu kombinieren und auf einem einzigen Prozessor auszuführen. Weil die unterschiedlichen Teile der System-Software auf einem einzigen Prozessor konsolidiert sind, können die Stückkosten erheblich reduziert werden. Automobil typische, sicherheitskritische Teile der Gesamtapplikation sind komplett durch Hardware isoliert. Diese Sicherheitsfunktion, verbunden mit höherer Leistung, macht Prozessoren auf Basis der ARMv8-R Architektur ideal für innovative Anwendungen im Automobil, wie beispielsweise Fahrerassistenzsysteme oder Fahrzeug-zu-Fahrzeug-Kommunikation.

Auch Industriesteuerungen und Automatisierungssysteme sind wesentliche Einsatzgebiete. Mit Prozessoren auf Basis der ARMv8-R Architektur wird es möglich sein, ein Linux- Betriebssystem für Grafik- und Netzwerk-Funktionen mit Echtzeit-Maschinensteuerungen auf einem einzigen Prozessor zu kombinieren, mit Kosten- und Integrationsvorteilen beispielsweise in der Bedienoberfläche einer Maschinensteuerung.

Zu den weiteren Innovationen der ARMv8-R Architektur gehören: ein Speicherschutz-System für kürzeren Kontextwechsel im RTOS (Real-Time Operating System), erweiterte ARM NEON™ SIMD-Befehle für Algorithmen in der Radar- und Bildverarbeitung sowie zusätzliche Befehle aus der ARMv8-A Architektur wie beispielsweise CRC (Cyclic Redundancy Check) zur Erkennung und Korrektur von Datenfehlern.