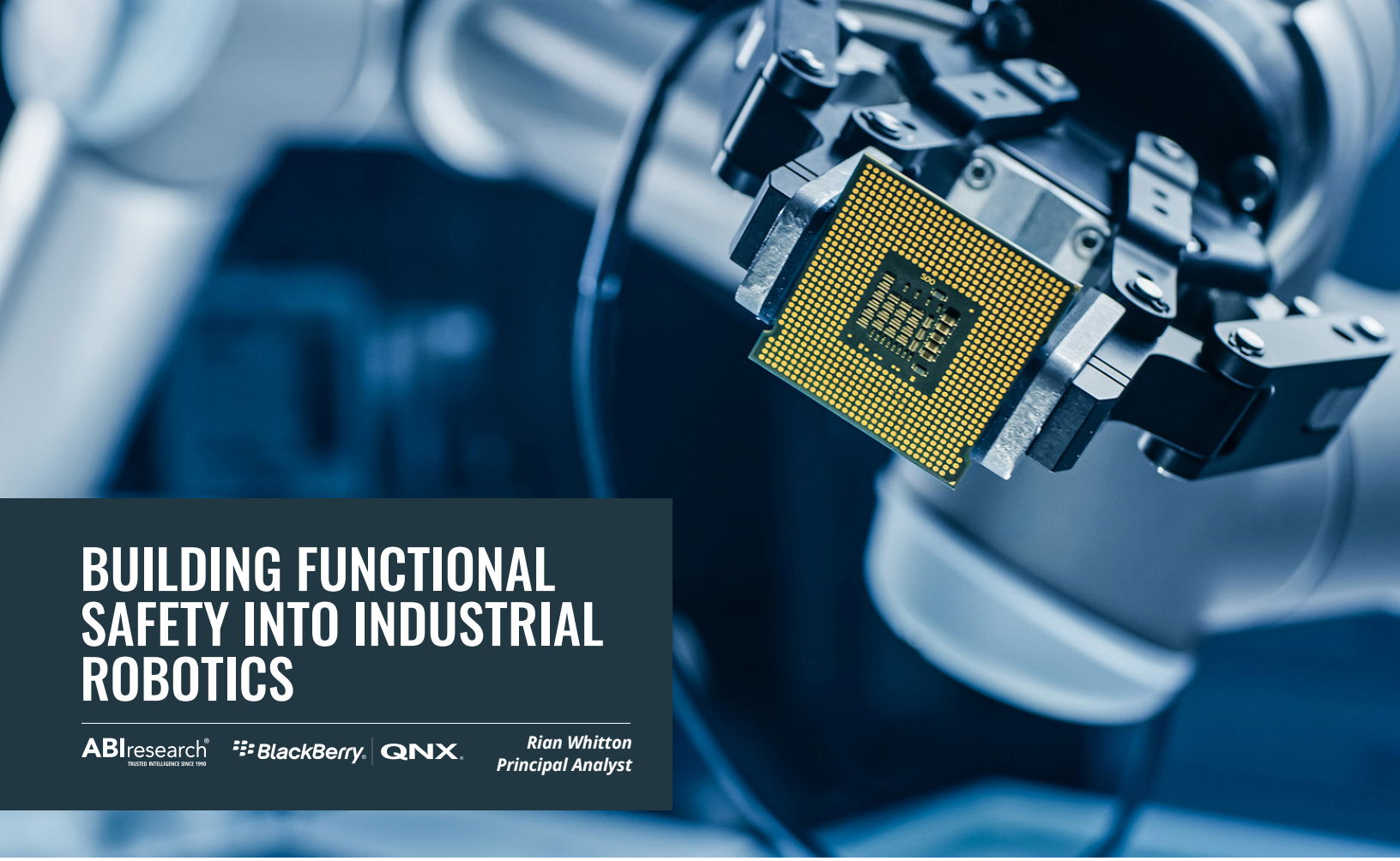


BUILDING FUNCTIONAL SAFETY INTO INDUSTRIAL ROBOTICS

ABIresearch® TRUSTED INTELLIGENCE SINCE 1990 | BlackBerry. | QNX.

Rian Whitton
Principal Analyst

TABLE OF CONTENTS

EXECUTIVE SUMMARY

Functional safety is critical to many modern robotic applications and will become more so. In the near future, many robots will run in areas where they interact directly and indirectly with workers, and the tasks they perform will disproportionately relate to production logistics and moving goods for the general public. If a robot malfunctions, there is a risk to the surrounding workers, to product safety, and to public safety. This risk amplifies as robots are increasingly required to handle multiple tasks and navigate crowded environments. So, mitigating safety risks under these conditions will become a fundamental requirement that cannot be addressed without considering security as an integral part of robot design.

This is where functional safety of the underlying software and hardware becomes an essential part of reducing, if not eliminating these risks. The need for more advanced software tools that orchestrate mixed criticality and ensure functional safety within a given system also becomes more apparent. Although robotics vendors want to use open-source solutions for many applications, they need certifiable and functional safety for mission-critical execution. Real-Time Operating Systems (RTOSs) and hypervisors from proprietary software developers can facilitate this, based on decades of providing certified software for the more developed automotive and medical device markets.

An RTOS is an Operating System (OS) that enables real-time communication between different embedded software through breaking up code into smaller blocks and tasks. An RTOS implements preemptive multi-tasking using a periodic interrupt routine that switches between running different tasks. This is known as task scheduling and most RTOSs use some kind of priority scheduling to determine which communication is most urgent. Many RTOS vendors have expanded their product portfolios by adding a hypervisor, which enables virtualization solutions without compromising the integrity of the overall system.

A hypervisor enables robotics system designers to consolidate systems from various suppliers and solution providers with different reliability, safety, and security requirements on a single System on Chip (SoC). A mobile robot, for example, is a complex system that features multiple functional domains, including motion, perception, localization, navigation, manipulation, and power. The main role of a hypervisor, also called a Virtual Machine Manager (VMM), is to orchestrate and run multiple workloads from multiple Virtual Machines (VMs) on a single abstracted hardware. The hypervisor controls the hardware, and VMs are isolated from the hypervisor and from each other, just as they would be if they were running on separate SoCs. By emulating the underlying hardware by way of virtual devices, the hypervisor ensures that software running in the VM runs as it would directly on the physical hardware. A good hypervisor ensures software executing in a VM must present no more than a minor decrease in speed compared to the same software running directly on the underlying hardware. This enables the robots to adhere to stringent safety and reliability requirements, despite their increasing complexity and functionalities.

This whitepaper explains the need for RTOSs and hypervisors for ensuring robotics' reliability, safety, and security requirements, the trends driving their adoption, and the advantages of proprietary and open-source RTOS vendors.

THE MARKET FOR INDUSTRIAL ROBOTICS

THE POTENTIAL OF THE ROBOTICS INDUSTRY

Following the initial crisis of the COVID-19 pandemic, the industrial robotics marketplace has seen expanded interest from investors, corporations, and even governments. High levels of investment and financial backing have made "automation" a hot topic, akin to the hype seen for Artificial Intelligence (AI) or quantum computing.

Despite many public automation vendors seeing some losses in 2020 due to the challenges of the pandemic, the results for 1Q 2021 point to a rebound that will likely precede a year of unprecedented revenue and growth for those in industrial robotics. As Table 1 illustrates, COVID-19 has failed to severely dent the bottom line of the robotics industry and has even allowed for accelerated optimism in 2021.

Table 1: Selected Company Performances 2020 and 1Q 2021

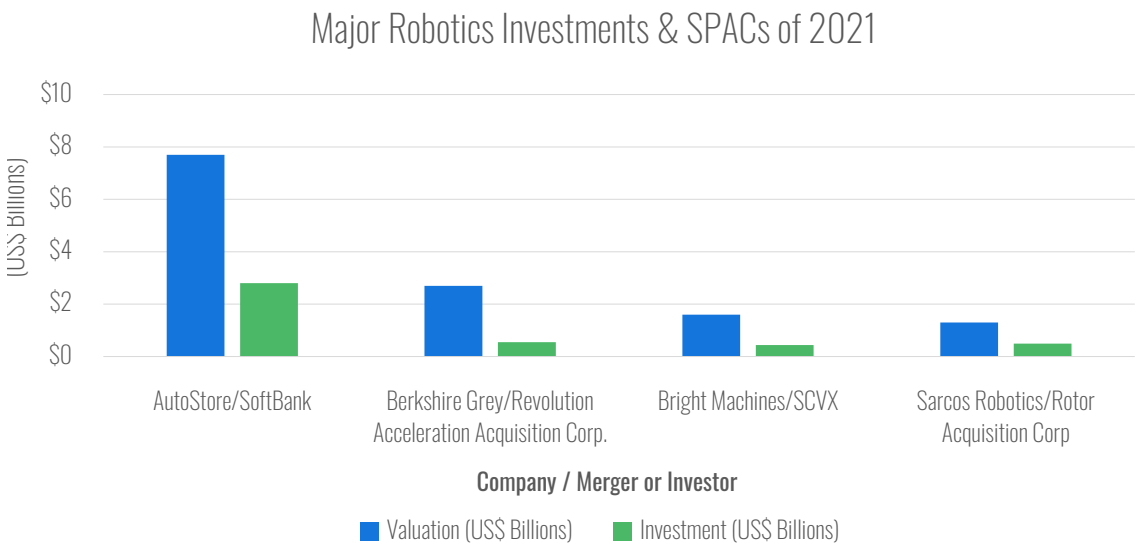
(Source: ABI Research)

COMPANY	2020 REVENUE (US$ AND PERCENTAGE CHANGE FROM 2019)	1Q 2021 REVENUE PERCENTAGE CHANGE FROM 1Q 2020 (%)
ABB Robotics	US$2.18 billion (-12%)	(+27%)
Kuka Robotics AG	US$1.19 billion (-22%)	(+3%)
Mobile Industrial Robots (Teradyne)	US$42 million (0%)	(+55%)
Universal Robots (Teradyne)	US$219 million (-12%)	(+ 32%)

This optimism is part of a broader scaling up of the industry, with investment being the key highlight. In 2011, venture capitalists and corporations spent US$194 million on robotics investments globally. This rocketed up to US$1.5 billion in 2015 and has since expanded to US$28.8 billion in 2019. While there was a minor drop in investments and acquisitions in 2020 to a value of US$20.2 billion, there have been many recent announcements of robotics vendors going public via Special-Purpose Acquisition Corporations (SPACs), as shown in Chart 1.

Chart 1: Notable Investments and SPACs 2021

(Source: ABI Research)



The companies receiving investments are increasingly focused on deploying fixed and mobile robots in closed, industrial spaces where there is no access to the public. These environments can include manufacturing facilities, warehouses, distribution centers, fulfillment centers, refineries, mines, and construction facilities.

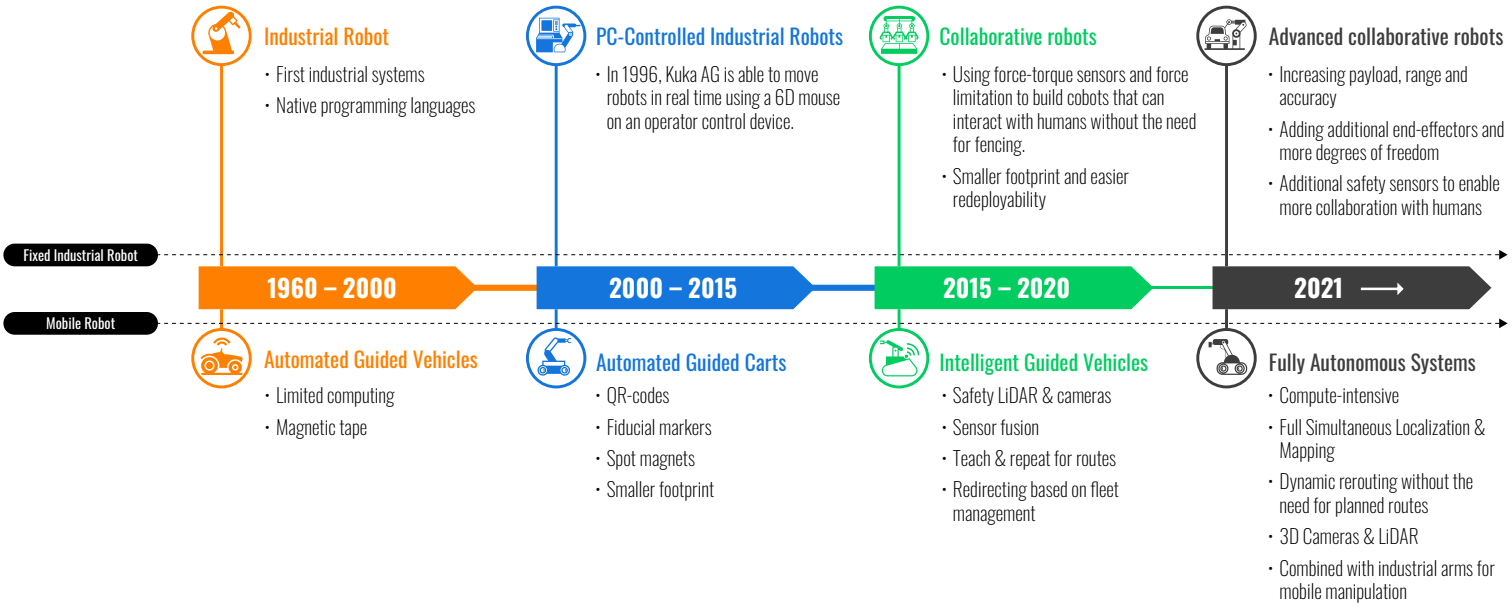
NEW ROBOTS, NEW DEMANDS

Until now, most industrial robots have been fixed, articulated, Selective Compliance Assembly Robot Arm (SCARA), parallel, and gantry robots used in the automotive and electronics manufacturing industries. They are fenced off from workers and other infrastructure, stationary, and powered through wired cabling. They are mechanically complex and computationally simple, with very little computational power being used beyond their set task.

Slowly, Automated Guided Vehicles (AGVs) have developed into highly autonomous systems that require far less "guidance" to move about. Figure 1 provides a short breakdown of the evolution of both traditional industrial robotic systems and mobile robotic systems. There is an irresistible trend toward more data consumption and higher demands for computing, in turn leading to the slow replacement of microcontrollers and individual electronic units with centralized SoCs.

Figure 1: Notable Investments and SPACs

(Source: ABI Research)



The trajectory of robotics runs in line with the greater goals of the fourth industrial revolution, defined by interconnectedness and interoperability between different products, cyber-physical systems, and extreme flexibility. Improvements in sensor and compute technology have greatly expanded the capacity of robots to perceive, map, localize within, and navigate industrial environments.

FUNCTIONAL SAFETY IS CRUCIAL FOR THE INDUSTRY TO MEET ITS POTENTIAL

As the number of robots in industrial applications grows, the unreasonable risk due to hazards caused by malfunctioning behavior increases. If uncontrolled, the risk will lead to production downtimes and operational losses, even compromising the safety and well-being of human employees. A functional safety system must be designed to prevent safety-critical errors at both the hardware and software levels. This is done via the safe and automatic management of software errors, operator errors, hardware failures, and challenges related to environmental changes, such as close proximity to workers or indoor/outdoor transitions. Functional safety is about ensuring design intent; and continuous operation of the safety-related system is a key intention. Any functional safety system has to actively detect and respond to errors or potential safety incidents. This distinguishes functional safety from passive safety systems, such as fences that cage an industrial robot, or limitations on a robot's payload or moving speed.

In addition to standards for hardware and software, functional safety is based on regulations, procedures, and auxiliary equipment needed to meet safety guidelines.

Table 2: Safety Approaches in Robotics

(Source: ABI Research)

CATEGORY	SAFETY THROUGH LIMITATIONS	SAFETY THROUGH TECHNOLOGY
Character	Passive	Active
Safety Features	Fencing, magnetic tape, clear demarcation between work cells and the rest of the factory	Onboard and external sensors to facilitate autonomous movement and limit force for collaboration with workers
Computing	Limited, microcontrollers	SoC, Application-Specific Integrated Circuit (ASIC), Graphics Processing Unit (GPU), Field-Programmable Gate Array (FPGA)
Connectivity	Wired, ethernet	Industrial, Wi-Fi, cellular
Pros	Reliable, safe (at the physical level), predictable, few autonomy exceptions	Enables greater autonomy and more flexibility in the workspace
Cons	Limiting, static	Autonomy exceptions, edge cases and need for certification, increased computing demands due to ingestion of data

So far, safety has been introduced as an add-on feature, often dealt with post manufacturing, not as an integral part of the robot design and value proposition. The risk is that once the robot is deployed based on the initial functionality requirements, it may not be the optimal topology for addressing critical safety properly. Critical and functional safety should be an integral part of the robot's deployment and not one that is added later. For any safety-critical robots, including industrial robots, AGVs, or fully autonomous systems, functional safety should be considered from the very early stages of building the robot architecture.

REGULATIONS

The regulations surrounding functional safety for electronic and electrical systems in industrial robotics are centered around three key sources. International Electrotechnical Commission (IEC) 61508 is the foundational set of regulations for supporting functional safety, while International Organization for Standardization (ISO) 13482 (robotics) and ISO 13849 (machinery) deal with more specific aspects of embedded systems within different products. While these regulations focus on functional safety, other regulations like ISO 102018 focus on passive safety measures like protective equipment and fencing.

Table 3: Robotics Regulations

(Source: ABI Research)

REGULATION	DESCRIPTION
IEC 61508	▪ IEC 61508 is the foundational source for good software methods, techniques and tools to support functional safety. ▪ The IEC 61508 series provides functional safety standards for the life cycle of Electrical, Electronic, or Programmable Electronic (E/E/PE) systems and products. It addresses those parts of a device or system that perform automated safety functions, including sensors, control logic, actuators, and micro-processors. ▪ IEC 61508 does not relate to protective equipment or procedures, which are covered by ISO 10218.
IEC 62061	▪ Safety of machinery – Functional safety of safety-related electrical, electronic, and programmable electronic control systems ▪ Defines safety integrity levels (SIL) for safety-related control systems
ISO 13482	▪ Specific to human-robot physical contact applications such as: ▪ Mobile servant robot ▪ Physical assistant robot ▪ Person carrier robot ▪ These are service robots that are growing in previously untapped markets like medical device, retail, and delivery.
ISO 10218	▪ Specific to industrial robots, which are generally fenced off from the public with physical infrastructure.
ISO 13849	▪ Specific to machinery control systems. ISO 13849 comes in two parts. Part 1 focuses on general principles for design and safety requirements, while Part 2 focuses on validation of safety certification through analysis. ▪ Many AGV and Autonomous Mobile Robot (AMR) vendors focus on this regulation.

Importantly, different robotics vendors may meet different levels of certifiability when it comes to these regulations. Pre-certified products pass an official audit and meet the standards of either the IEC or the ISO, while other vendors might merely say they are compliant. This can be as limited as saying the product was developed with regulations in mind, but it may not have passed through any third-party validation.

A further point of note for vendors and end users is the likely Safety Integrity Level (SIL) of a robot's application. There are four levels of SIL, defined by the acceptable probability of dangerous failure and the necessary risk reduction factor. These criteria are dependent on the force being used, as well as potential interactions with workers. For higher SIL (3 to 4) applications, special precautions like redundant data collection must be undertaken, and certified RTOSs become a necessity.

EMBEDDED SYSTEMS FOR ROBOTICS

Robots are steadily becoming more computationally complex, so there are shifts away from multiple microcontrollers toward more expensive computing platforms and SoCs. The best examples of these include the NVIDIA Jetson series, as well as the Qualcomm Robotics RB5 Platform and Intel's X86 product line.

The adoption of these systems is being driven by increased demands for more intelligent robotics platforms capable of handling the following functionalities:

- Collect and hold more data
- Sensor fusion
- AI training and inference

To address these functionalities, most robotics architectures are distributed between Arm/X86 processors, specialty systems like GPUs, and microcontrollers for motors, sensors, and *similar systems*. The software stack is divided between high functions like navigation and path planning, and low functions related to brakes, batteries, and gears, as described below:

- High functions involve communication to central fleet management systems and to cloud platforms, and are generally enabled by a Data Distribution Service (DDS), either sourced from Linux or RTI, the originator of DDS.
- Low functions related to the internal communication within the robot are generally managed by specialist software, often packaged in an RTOS. Embedded systems in robots increasingly require real-time behavior, and due to hardware resource constraints, performance and efficiency are top priorities. An RTOS provides the resource management and scheduling required to meet the demands of applications.

Table 4: Breakdown of Software Stack for Mobile Robot Vendors

(Source: ABI Research)

SEGMENT	FUNCTION	SOFTWARE	COMMUNICATION PROTOCOL
High-level systems	Simultaneous Location and Mapping (SLAM) Perception Path planning	Linux (Robot Operating System (ROS)/ROS 2), Ubuntu, proprietary software products DDS	Wi-Fi Cellular Ethernet
Low-level systems	Communication between motherboard and microcontrollers for navigation	RTOS, FreeRTOS, QNX Neutrino, Micro-Velocity	Ethernet / CANBus / Modbus
Low-level systems/ Mission Critical	Protected stop	Safety Programmable Logic Controllers (PLCs) from off-the-shelf, certified RTOS	Ethernet / CANBus / Modbus / RS232

For internal communication, there is further division between safety-critical elements and non-safety-critical elements. The non-safety-critical communications are generally orchestrated by open-source solutions like Linux and FreeRTOS. Safety-critical communications are isolated and based on safety PLCs anchored to individual certified sensors. This off-the-shelf solution for RTOS and internal communications is due to a range of factors.

Table 5: Comparing Safety Implementation, Requirements, and Addressable Market across Automotive and Industrial Robot Vendors

(Source: ABI Research)

VERTICAL	AUTOMOTIVE	INDUSTRIAL ROBOTICS (FIXED)	INDUSTRIAL ROBOTICS (MOBILE ROBOTS
Typical leading companies	Larger multi-billion-dollar entities	Largest vendors post more than US$1 billion a year	Smaller companies and startups
Safety requirements	▪ Vehicles must meet a very high bar for safety due to being in dynamic environments	▪ Many microcontrollers needed for articulated robotic systems	▪ Safety demands are strict, but confined largely to fixed environments; they can also be mitigated by force limitations; no robot has to go up to 40 miles per hour ▪ Bump sensors and Light Detection and Ranging (LiDAR) safety systems are used for AMR safety ▪ AGVs and Automated Storage and Retrieval Systems (AS/RSs) are mainly based on a General-Purpose Operating System (GPOS)
Typical safety implementation methods	▪ Develop safety solution through long-term planning and partnerships ▪ Have the resources and predictability	▪ Use proprietary RTOS from vendors like QNX Neutrino, Green Hills Integrity, VxWorks (Wind River), Micro-Velocity	▪ Limited resources, time constraints, and rush to market lead robotics companies to take the most "off-the-shelf" solution they can
Addressable market	▪ Limited growth outlook for annual shipments ▪ Millions of units	▪ Solid growth outlook ▪ Hundreds of thousands of units	▪ Very high-growth outlook ▪ Tens of thousands of units, but will grow to a million within the decade

RTOS AND THE BENEFITS OF MICROKERNEL ARCHITECTURE

An RTOS can either be based on a monolithic or microkernel architecture. Monolithic architectures benefit from faster performance due to a lower number of context switches, because process scheduling, memory management, and file management all run as a single large process in the same address space. However, their greater centrality may mean that system updates are more complex and that failures may require a reboot. Alongside greater maintenance challenges, highly complex machines like automobiles, medical devices, and robots are using microkernel architecture.

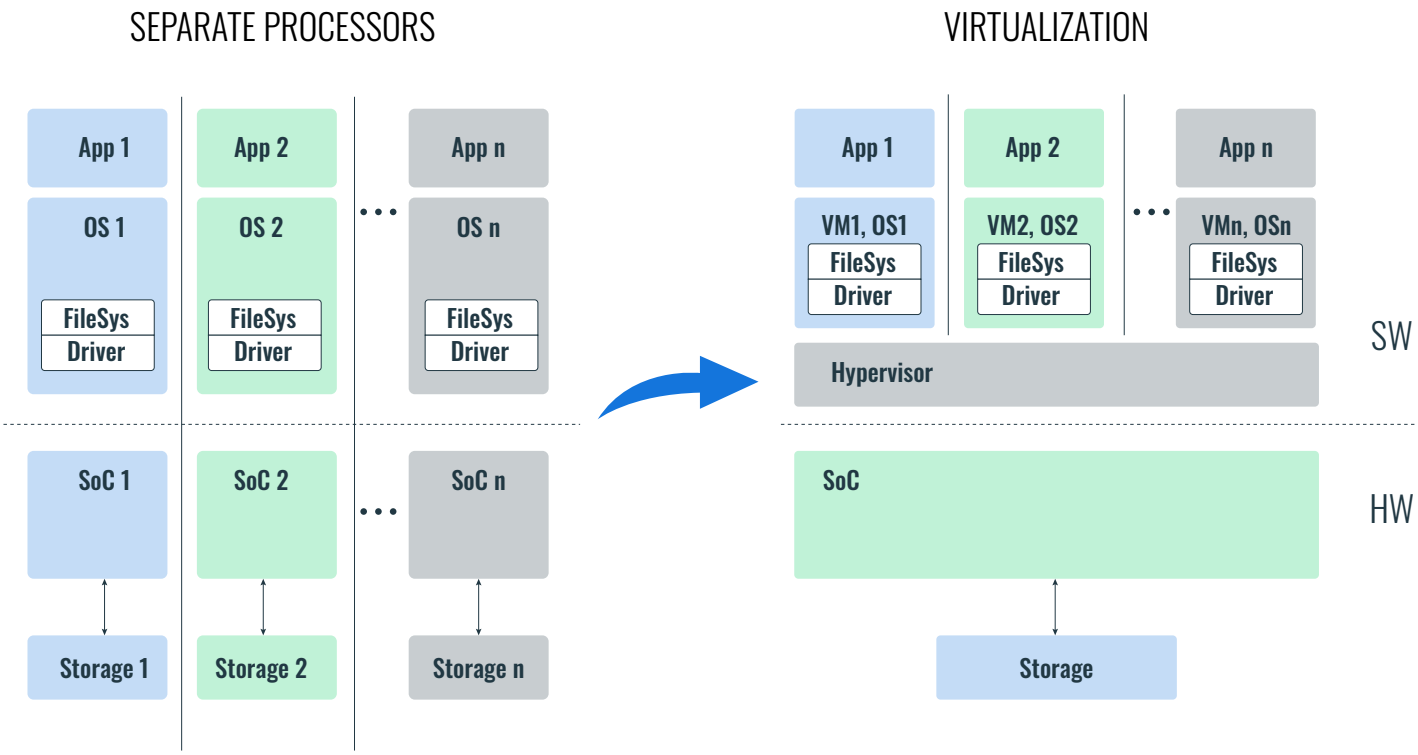
Microkernel architectures are becoming more popular due to their reliability, safety, and smaller footprint. The microkernel protects and allocates memory for other processes and provides task switching. All other components, including drivers and system-level components, are each contained within their own isolated process space. This allows for isolation of individual system faults, easy debugging and offering easy expansion through extensions with other OSs.

An aspect of functional safety is designing the system to isolate safety-critical functions from other functions and ensure they are free from interference. Hypervisors can add to this by providing spatial isolation, which means code or data from one partition cannot be altered by the data of another partition. Meanwhile, temporal isolation ensures that a virtual partition cannot affect the ability of the other virtual partitions to access a shared resource.

HYPERVISORS-AS-A-SOLUTION

A hypervisor, also called a VMM, is a software abstraction layer that sits on a single abstracted hardware (like an SoC), and is able to manage and orchestrate multiple workloads from various VMs. These VMs provide environments in which different OSs and their applications can run. An OS and its applications in a hypervisor VM are known as a guest, as illustrated in Figure 2.

Figure 2: Where Hypervisors Sit in the Operating Stack
(Source: BlackBerry)



A hypervisor manages separate VMs, making sure they share resources without conflict, and ensuring both spatial and temporal isolation. An embedded hypervisor allows designers to run separate and isolated guest OSs, such as Linux, Android, and QNX, on a single SoC. This is essential for ensuring safe mixed-criticality communications and allows more processing to be done at a higher level of efficiency on a single piece of hardware.

For example, the important warning displays on an automobile digital instrument cluster should be isolated from vehicle infotainment systems, even if they share display space on the dashboard. In an industrial context, an RTOS can be used for isolating non-critical elements like the Graphical User Interface (GUI) display software from automated control software within an industrial control unit, even as they are running on the same SoC.

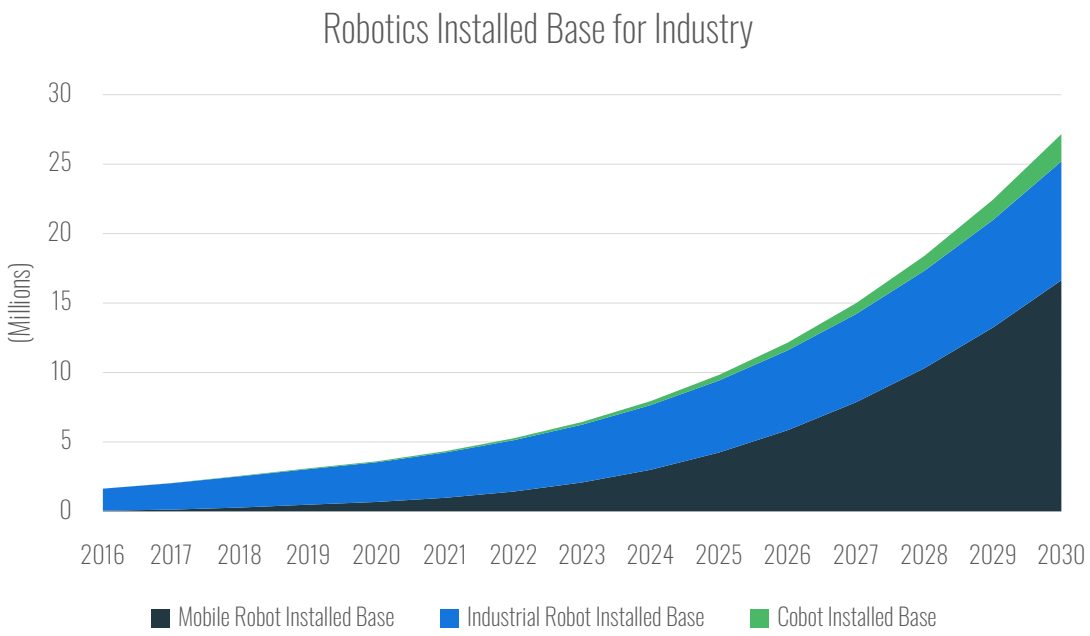
A microkernel architecture like the BlackBerry QNX RTOS can separate multiple functions, both spatially and temporally, and provide isolation and freedom from interference in embedded systems with mixed criticality.

MARKET SIZING AND THE RTOS OPPORTUNITY

The International Federation of Robotics (IFR) estimates that 2.7 million industrial robots were installed as of 2020 and, by the end of the decade, this will be supplemented by collaborative and industrial mobile robot shipments. By 2030, ABI Research projects there will be up to 27 million industrial, collaborative, and/or mobile robots in industrial and logistics environments, as seen in Chart 2. Of these, 3 million will be mobile robots operating in outdoor environments for applications like cleaning, maintenance, construction, agriculture, and delivery. They will require similar levels of certification as in the modern automotive sector. Meanwhile, another 8.5 million of these robots will be fixed industrial systems, which already use an RTOS to a high degree. Based on these projections, the demand for RTOSs and hypervisors to manage mixed-criticality communications is set to grow substantially over the decade.

Chart 2: Robotics Installed Base for Industry World Markets: 2016 to 2030

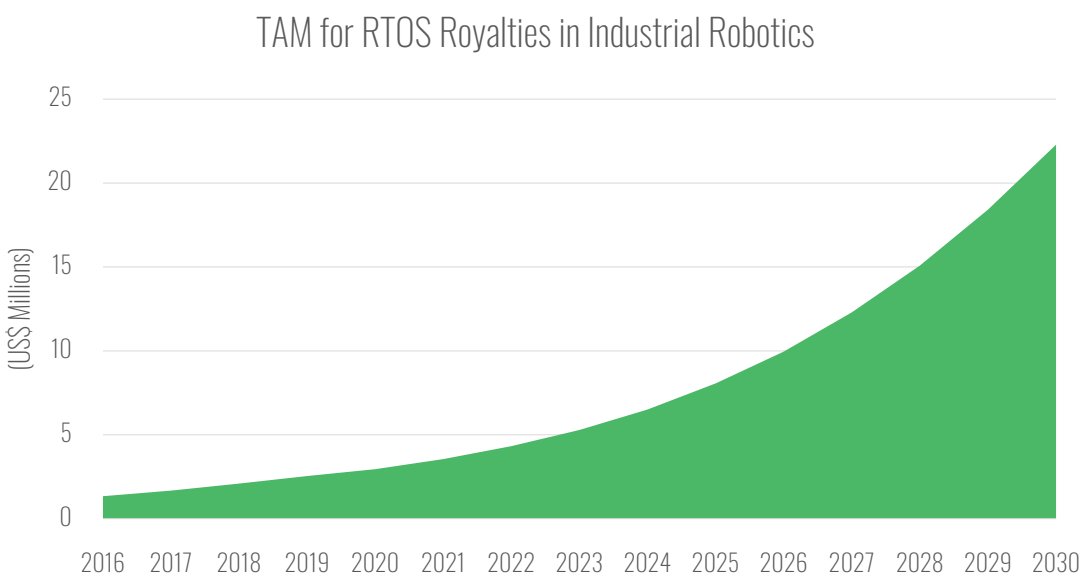
(Source: ABI Research)



Based on the projected installed base of the commercial robotics market over the next 10 years, the Total Addressable Market (TAM) for proprietary RTOSs in this market will amount to more than US$22.3 million in 2030 (as seen in Chart 3). This compares to US$3 million in 2020.

Chart 3: TAM for RTOS in Industrial Robotics World Markets: 2016 to 2030

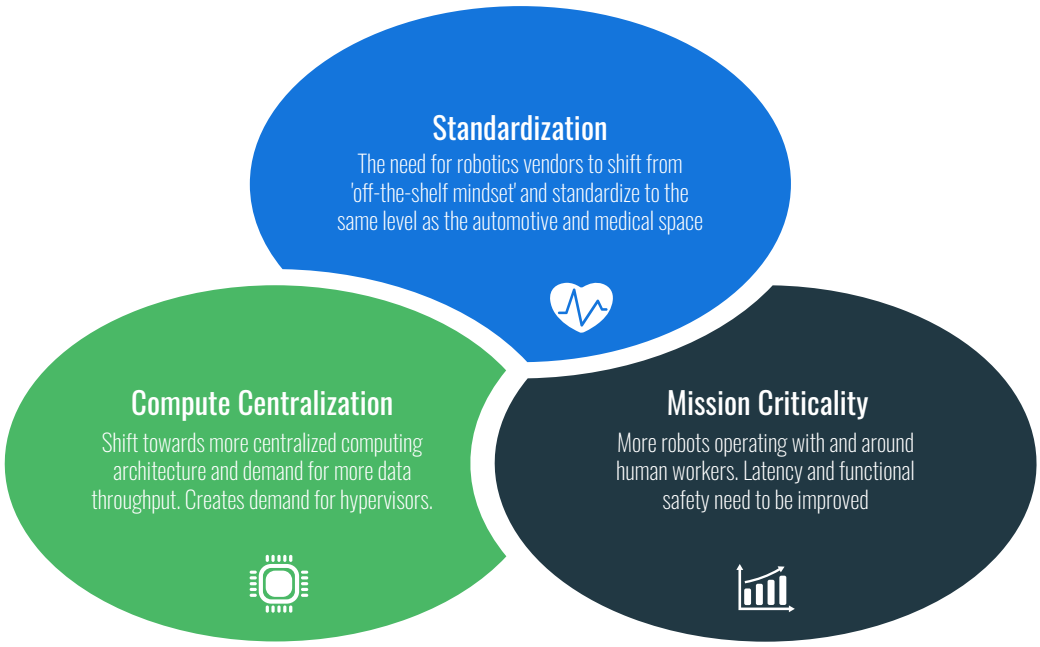
(Source: ABI Research)



SAFETY STANDARDIZATION, COMPUTATION, AND LATENCY: THE THREE TRENDS DRIVING RTOS TECHNOLOGIES IN INDUSTRIAL ROBOTICS

Figure 3: Three Trends Powering RTOS and Hypervisor Adoption in Robotics

(Source: ABI Research)



SAFETY STANDARDIZATION FOR RTOS AND HYPERVISORS TO MANAGE MISSION-CRITICAL AND NON-CRITICAL COMMUNICATION

The global robotics industry is currently fragmented. Given the sheer range of use cases, there is inevitable variation between functional standards. AGVs and AMRs operating at slow speeds in controlled environments have less demand on latency than outdoor robots or industrial arms.

But as fleets grow in size, regulatory demands are set to increase. There are also calls for greater interoperability between different robot OSs to allow for information sharing. This is exemplified by two initiatives: 1) the German VDA 5050 proposal, and 2) the interoperability working group being led by MassRobotics. If there is to be effective information sharing between different robotic systems, it is essential to develop common platforms that cover potentially multiple certifications. A proprietary RTOS could perform this role for robotics as it does in the automotive industry.

DEMAND FOR IMPROVED COMPUTING POWER AND COMPUTE CENTRALIZATION

As robots are expected to attain greater insight on their environment, microcontrollers must eventually give way to more sophisticated systems that can train and infer highly complex algorithms. Looking forward, a number of exciting techniques have yet to be popularized, as shown in Table 6.

Table 6: New Robotic Competencies for Mobile Industrial Robots

(Source: ABI Research)

INNOVATIONS	DESCRIPTION	VALUE
Panoptic Segmentation	Use AI/Machine Learning (ML) to categorize collections of pixels from camera feeds into recognizable "objects." For example, the millions of pixels representing a wall can be categorized as a single object.	▪ Consolidating Three-Dimensional (3D) points into simpler objects lowers overhead on processors. ▪ Easy categorization of individual objects.
Object Perception	Will allow a robot to learn to distinguish the walls and floors of a room from the furniture and other objects within it.	▪ Storing these elements as individual objects means that adding or removing a chair will not necessitate the complete redrawing of the map. ▪ Reduces the cost of calibration.
Natural Interaction	Robots that learn from multiple situations and combine that knowledge into a model that allows them to take on new, untrained tasks based on maps and objects preserved in memory. Creating those models and abstraction demands complete integration of all three layers of SLAM: perception, mapping, and semantics.	▪ Incremental improvements. ▪ Improved Human-Machine Interface (HMI).
Mobile Manipulation	Mobile manipulation systems require the integration of a large number of hardware components for sensing, manipulation, and locomotion, as well as the orchestration of algorithmic capabilities in perception, manipulation, learning, control, planning, etc.	▪ Burgeoning use cases, with an emphasis on high-value/low-volume projects. ▪ Combines the value of mobile and articulated robots onto one platform.

To deal with these new processing demands, robots increasingly must house and employ larger and all-encompassing SoCs, replacing distributed computing architectures full of microcontrollers with centralized computing architectures that use fewer inputs and can achieve greater efficiency. This gradual centralization means implementing multiple software systems on the same SoC, thereby reducing the overall number of SoCs needed in an embedded device. This can be seen through new releases like NVIDIA's Jetson and Jetson Nano, and the Qualcomm RB5 platform that delivers orchestration of mixed-criticality communications.

DEMANDS FOR ULTRA-LOW LATENCY DUE TO NEW USE CASES

As robots are expected to do more in the factory, safety-critical applications demand ultra-low latency. A review of different use cases and the necessary latency to make them feasible is provided in Table 7.

Table 7: Latency Demands for Nascent Robotics Applications
(Source: ABI Research)

USE CASE	RELIABILITY DEMANDS	LATENCY REQUIREMENT (MILLISECONDS (MS))	ROBOT APPLICATIONS	VENDORS
Cooperative motion control for robotic arms	>99.9999%	1 ms	Collaborative robots (cobots)	Universal Robots, ABB, FANUC, AUBO, Flexiv
Video-operated remote control for mobile robots	>99.9999%	10 ms to 100 ms	Outdoor field robotics	Re2 Robotics, Sarcos Robotics, ANYBotics
Mobile control of assembly robots	99.999%	4 ms to 8 ms	Automotive, electronics	ABB, Kuka AG, FANUC, Yaskawa
Autonomous navigation	95%	1,000 ms	Industrial material handling	Seegrid, Vecna, MiR, Outrider
Remote driving in public spaces	>99.9999%	5 ms to 10 ms	Autonomous haulage and last-mile delivery	Nuro AI, Starship Technologies
Cloud robotics platform	99%	10 ms to 100 ms	Industrial robotics	Ericsson, Amazon Web Services (AWS), Microsoft Azure

One of the challenges of deploying Ultra-Reliable Low-Latency Communication (URLLC) competency for internal communications is that it is wasted on the very limited latency capabilities of external connectivity solutions like industrial Wi-Fi. As 5G begins to roll out, promising URLLC of up to under 1 ms will increase the value of low-latency communication for non-safety-critical elements.

NEW APPLICATIONS FOR OUTDOORS

While the majority of robotic shipments will be based indoors in structured environments, there is also a burgeoning market for industrial and service robots that are designed to operate outside in crowded public spaces. Recent partnerships between couriers like FedEx and autonomous delivery vendors like Nuro point to a not-too-distant future when robots delivering groceries will be normalized within university campuses, hospitals, and small suburban zones. These robots will require the same level of certification as automobiles and are much more likely to take up functional safety solutions from proprietary RTOS vendors. ABI Research projects that as many as 1 million delivery robots will be on pavements and roads by 2030.

Combined with increased compute centralization and the popularization of robot-specific SoCs, the growth of robot markets with high latency demands and mixed-criticality communications will spur significant interest in hypervisors from robotics vendors and developers.

PROPRIETARY AND OPEN-SOURCE SOLUTIONS; SYNERGY OVER COMPETITION

CURRENT STANDARD IN THE INDUSTRY

Much like with other OSs, whether a robotics vendor uses proprietary or open-source RTOS solutions depends on their place in the market. The established industrial robotics market, which still represents 81% of the robotics installed base in 2020, uses proprietary RTOS due to their improved latency performance and certifiability.

Mobile robotics vendors are very different. They tend to be small, build solutions before seeking regulatory approval, and generally do not want to develop their own safety solutions. Therefore, they rely on a mixture of open-source RTOSs and certified safety PLCs from component manufacturers. Cobot vendors are also concerned about the cost of royalties.

The adoption of proprietary RTOSs and hypervisors correlates with two factors:

- **The Size of the Vendor**: Smaller vendors and teams are more likely to use open-source solutions, as they are looking for the fastest route to market. Larger vendors that have larger fleet sizes are often more open to ruggedizing their safety to go beyond the bare minimum.
- **The Nature of the Application**: For outdoor robots being placed in and around the general public, the need for professional RTOSs and hypervisors is clear. It is also true that fixed industrial robot vendors need to rely on very low latency, so they are generally prioritizing high-end safety-certified communications over expediency. While modern AGVs and AMRs require less in the way of safety requirements, more sophisticated mobile manipulation systems and higher speeds will eventually lead to higher demand for proprietary RTOSs and the added functional safety capabilities they provide. Ultimately, the more stringent the latency demands and the higher the potential danger, the more inclined developers are to pay for proprietary solutions.

SYNERGY BETWEEN OPEN-SOURCE ECOSYSTEM AND PROPRIETARY SOLUTIONS

The Open Source Robotics Foundation (OSRF) has played an instrumental role in the development of OSs for AMRs through its popular ROS (ROS & ROS 2.0) projects. An increasing number of companies use their software stack, and they have even developed Micro-ROS, a middleware that can run on smaller microcontrollers. Currently, ROS only supports Windows, Ubuntu, and Mac OS as multi-level partners, and is not partnering with any proprietary RTOS. The OSRF has indicated that such partnerships are possible in the future, and it perceives the proprietary RTOS as a critical component of future software architectures. Given ROS's open-source nature, this does not stop proprietary vendors from using them as they wish.

The work of the OSRF and the development of the ROS have significantly altered the landscape of robotics, with significant uptake for AMRs in particular. In the drone space, the open-source PX4 software ecosystem has evolved with open-source developer Auterion to become a key OS software infrastructure for consumer and commercial drone use cases. The further refinement of ROS into ROS 2.0 has attempted to iron out the challenges of open-source middleware and make open source the standard for robotics in industry. This

has not dented the popularity of proprietary systems though. Solutions like BrainOS, that use parts of ROS have proliferated with considerable success. Some newer entrants in the industrial AMR market, such as BMW IDEALworks, are using the proprietary NVIDIA Isaac Software Development Kit (SDK) for navigation and simulation development. Importantly, there is a high level of compatibility and collaboration between these proprietary systems and the open-source community.

Likewise, in the embedded systems space, there is a dichotomy between open-source solutions offered by FreeRTOS and Linux, and proprietary solutions offered by suppliers like BlackBerry QNX. At the same time, there is a lot of communication between the two ecosystems. For example, QNX is compatible with Linux message and is POSIX-compliant, meaning that there can be a high degree of segmentation for mixed-criticality communication between them and the open-source community.

While there have been attempts to certify Linux, BlackBerry QNX and other proprietary vendors currently have the advantage of certification under IEC 61508 and the specialized ISO regulations. Their adoption is heavily dependent on the SIL required for safe operation. For lower SIL classifications, a compliant open-source solution may be feasible, but in other instances, certified providers are in much higher demand.

THE ROLE AND LIMITATIONS OF OPEN SOURCE FOR FUNCTIONAL SAFETY

Historically, most RTOSs for industrial robotics have been proprietary. The larger vendors, including ABB, Comau, Yaskawa, Kuka AG, and others, all use proprietary RTOS solutions. As established vendors, the companies prioritize the certifiability of proprietary RTOS and have long-running partnerships with their collaborators.

On the cutting edge of the articulated arm market, cobot vendors are increasingly relying on FreeRTOS, due to the fact they can attain approval from relevant regulators. Cobot vendors have also highlighted how development costs for royalties affected their decision. Importantly, many cobots are operating in low-risk applications with low payloads and force-limited robots. Higher SIL applications would necessitate a certified proprietary RTOS in the long term.

AGV and AMR startups are heavily reliant on open-source solutions and generally highlight the royalty charges as the single most important factor in their decision to choose open source. However, there are hidden costs associated with open source, notably Linux solutions, including the need for developers to spend more of their time maintaining and updating patchwork solutions. As safety certification rules change and are affected by external events, pre-certified products that are packaged will become more convenient and less expensive to deploy. This is especially true considering robots are beginning to enter the public space in retail stores and on sidewalks, in particular.

THE VALUE OF PROPRIETARY SOLUTIONS

Vendors will gravitate more toward proprietary solutions depending on the level of potential danger. An AMR vendor was quoted as saying:

> *We use proprietary solutions for meeting any high-SIL/high-PL requirements, mainly because they are encapsulated within products which we purchase from our development partners. Lower-SIL/lower-PL subsystems tend to be built using open-source solutions for time to market and cost reasons.*

The AMR industry can, therefore, expect to see growth in market share among proprietary RTOSs and hypervisors to both enable robots to meet high SIL requirements and to orchestrate between different OSs.

As high-SIL requirements become more common due to the adoption of robots for outdoor and public-facing environments, the RTOS market for AMRs and mobile systems will resemble that of the more established fixed industrial robot market.

KEY TAKEAWAYS

The established industrial robotics market already uses proprietary RTOS solutions and, currently, as much as 81% of installed commercial and industrial robots use proprietary RTOSs. But for mobile solutions, the norm is either an open-source RTOS or even a General-Purpose Operating System (GPOS).

Nascent mobile robotics vendors are generally building their solutions with a permissive attitude on the basis that the regulatory challenges will be worked out through an evolutionary process that will not impede time-to-market. Part of this is due to the significant limitations already placed on robots from a safety perspective, and the fact that they do not have the same requirements as automobiles or medical robots when it comes to latency.

The demands for safety standardization, gradual compute centralization, and the expanded growth of mission-critical robotic applications will make the products from proprietary RTOS and hypervisor vendors, such as the BlackBerry QNX, even more relevant.

Published July 2021

157 Columbus Ave, 4th Floor
New York, New York 10023 USA
Tel: +1 516-624-2500

www.abiresearch.com

About ABI Research

ABI Research helps organizations–and visionaries within those organizations–successfully conquer digital transformation. Since 1990, we have partnered with hundreds of leading technology brands, cutting-edge companies, forward-thinking government agencies, and innovative trade groups around the globe. Through our leading-edge research and worldwide team of analysts, we deliver actionable insight and strategic guidance on the transformative technologies that are reshaping industries, economies, and workforces today.