EIU

ELECTRONICS INFORMATION UPDATE

A Mouser Magazine

EUROPE | APRIL 2024

mouser.com

Practical Robotics

FEATURES

- Revolutionizing logistics
- Mastering multi-axis robotics
- Enabling robot operating systems
- MCUs to improve robot motor control
- Driving Cobot motors reliably and efficiently

PLUS

REGULARS

- Industry News:
  - Superconducting spin-tech for cryogenic electronics
  - Biden's $20B stimulus for Intel
  - KR & Europe strengthen 6G and chips co-op
  - u-blox helps save penguins
- Embedded Linux: NEW mini-series
- Low-power mindset for IoT
- Test & Measurement
- Connector Geek
- Tech Ideas
- Dev Kit pick
- NPI

Cover image credit: Falookii/Shutterstock.com

EIU
ELECTRONICS INFORMATION UPDATE

mouser.com

In this issue...

Robots are all around us, and they are more powerful and flexible than previously imagined. Our April feature looks at practical robotics technologies, with articles including: MCUs to improve robot motor control; Mastering multi-axis robotics; Revolutionizing logistics; Enabling robot operating systems; and Driving Cobot motors reliably and efficiently.

Adam Taylor begins a six-part exploration of Embedded Linux, and Qoitech continues to probe the low power IoT mindset. David Pike picks connector winners, Stuart Cording makes the STEM classroom robotics-ready, and Tech Tips investigates the relationship between electronics and 'open source'. Plus Dev Kit Pick, the news round-up, and, of course, a review of the most innovative products now in stock at Mouser. Enjoy!

EIU – Electronics Information Update
Publisher: Mouser Electronics Inc., Georg-Brauchle-Ring 53, 80992 Munich, Germany, represented by Mark Patrick, Director Technical Content EMEA, eu.mouser.com

EIU Editorial Management and Production: Nick Foot, BWW Communications, Suite 6, Festival House, 39 Oxford Street, Newbury, RG14 1JG, United Kingdom, nick.foot@bwwcomms.com, bwwcomms.com

Sales & Marketing: Claudia Bertaccini, Director Marketing Communications EMEA, claudia.bertaccini@mouser.com

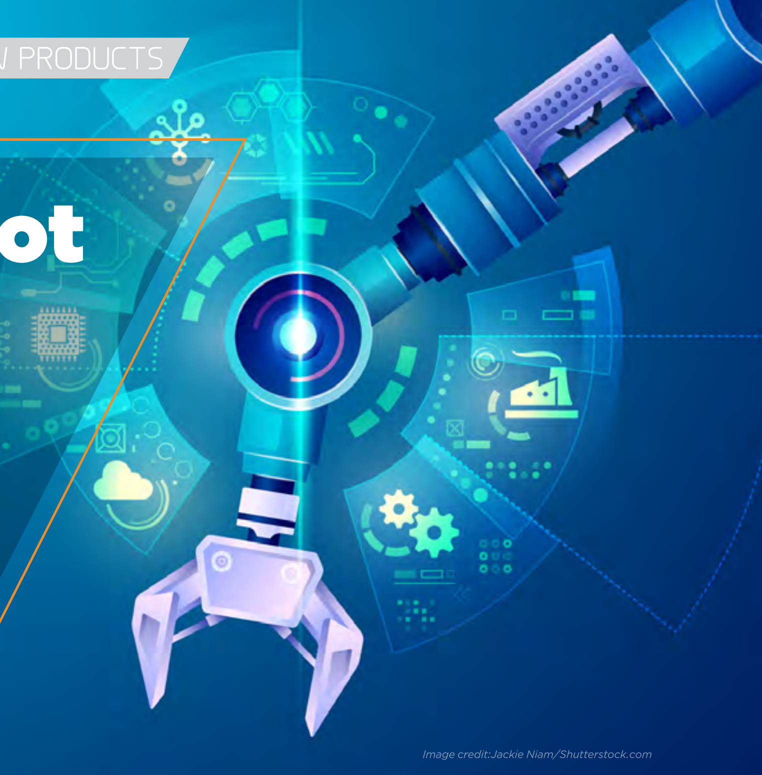


Image credit:Jackie Niam/Shutterstock.com

Engineering Cobot Motor Drives for Reliability and Efficiency

Contributed by Nexperia

Industrial robots have brought programmable automation to factory floors and areas for storage, sorting, and packing, as businesses seek to increase their productivity and reduce operating overheads.

More recently, collaborative robots (cobots) have arrived and created an extra, new world of application opportunities. Designed to work safely alongside humans, they bring the unique ability to act as co-workers contributing to the processes that must be completed to achieve the result.

This softer image of the robot as a coworker – in contrast to those unapproachable, caged superhumans used for hazardous jobs like heavy lifting, welding, and paint spraying - promotes the adoption of cobots in a wider range of tasks.

It raises the exciting prospect of combining human qualities such as natural dexterity and visual acuity, combined with the superior speed, repeatability, and strength of the robot.

Cobots are also assumed to have a reliability advantage over humans, with no tendencies towards illness, lateness, or loss of concentration while working. As important as performance, efficiency, and safety, reliability needs to be designed in to ensure that the cobot will deliver the operator's expected return on investment (ROI).

Motor-Drive Design Requirements

Figure 1 describes the underlying electrical architecture of a typical industrial cobot.

The motors and motor-drive circuits that move each of the cobot's joints must be able to handle the necessary loads and movements without straining or overheating. Motors that offer a high torque-to-weight ratio and small form factor in relation to the demands at each joint are usually preferred to permit a lightweight and compact robot design that can be safe and easy to work with around humans.

However, such requirements put a significant level of expectations on the motor-drive circuitry where the power electronics reside. The smallest possible module size is required, consistent with the goals of compactness and light weight. The size constraints limit thermal performance, which increases the stress on power semiconductor components.

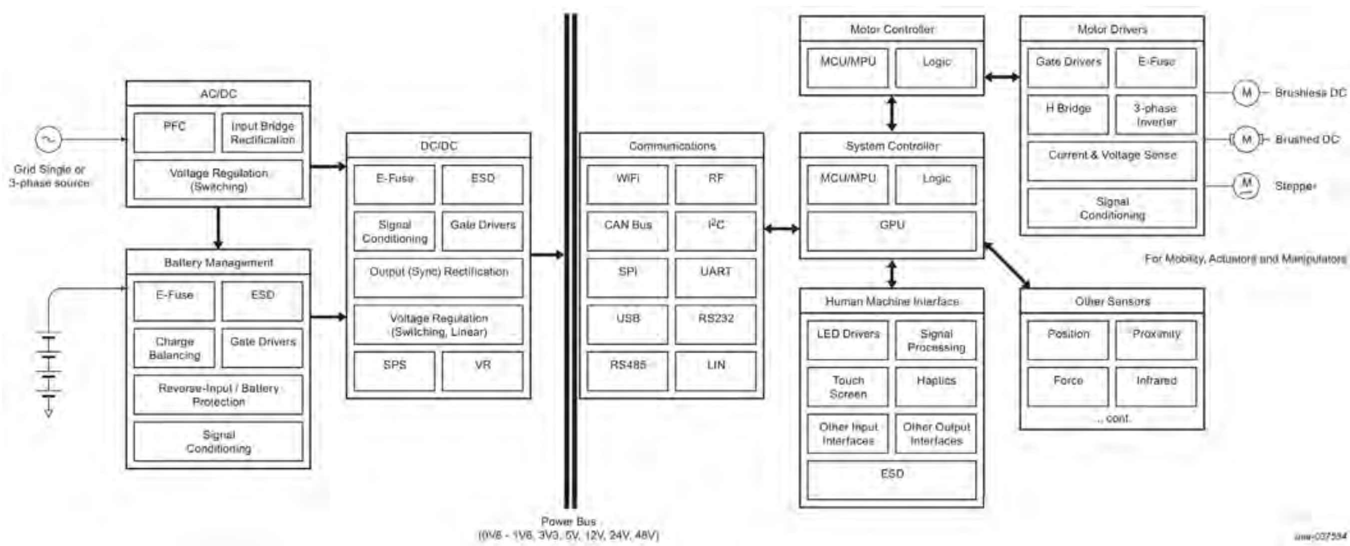


Figure 1: Cobot power, communication, sensing, and driving subsystems.

Enabling Robot Operating Systems

— Introducing the ADI Trinamic Motor Controller ROS1 Driver

Image credit: NicoElNino/Shutterstock.com

By Krizelle Paulene Apostol, Jamila Macagba and Maggie Maralit, Analog Devices

Abstract

Robot Operating System (ROS) drivers were developed on Analog Devices products so that they can be readily used within a ROS ecosystem. This article will give an overview on how to use and integrate these drivers in their applications, products, and systems (for example, autonomous navigation, safety bubble maps, and data collection robots); and how doing this will enable them to evaluate new technology immediately and avoid interoperability issues with third-party products. Among all products that will be discussed here, the focus will be on the recently released ROS driver for ADI Trinamic™ motor controllers—which are complete, board-level modules for embedded motion control, equipped with ADI Trinamic motion control expertise and ADI's analog process technology and power design skills.

What Is ROS?

ROS is robotics middleware containing a set of software libraries and powerful developer tools from drivers to state-of-the-art algorithms—upon which robotic systems or applications can be developed. It is multidomain (for example, consumer, industrial, automotive, etc.), and supports multiple platforms that is, Linux, Windows, MacOS, and some embedded platforms)—plus it's 100% open source with commercial options.

Support for ROS is abundant due to the dedicated resources from the global community, giving users an easier path for their designs and applications.

How Does the Technology Work?

ROS started in 2007 and became one of the most popular prototyping platforms for robotic development in fields such as self-driving cars, industrial robots, aerial vehicles, and more. It has continuously evolved and now has two versions: ROS1 and ROS2.

ROS1 and ROS2 systems must be isolated but the ROS bridge enables communication and exchange of data between them. More info is available at the ros2/ros1_bridge page.

Factors	ROS1	ROS2
Communications protocol	XMLRPC + TCPROS	DDS
Architecture	ROS master + distributed	Fully decentralized
Build system	Catkin (cmake-based)	colcon/ament (cmake-based)
Build output	ros_ws/devel	ros_ws/install
Parameters	Global parameter server Dynamic reconfigure	Per-node parameters
Launch	XML	Python (+XML, YAML alternatives)
Commands	roslaunch, rosrun, rostopic, etc.	ros2 launch, ros2 run, ros2 topic, etc.
Platforms	Primarily ubuntu	Linux, MacOS, Windows

Table 1. Major Differences of ROS1 and ROS22

ROS Supported Platforms

ROS Noetic is the final version of ROS1 that will be supported until May 2025, while ROS2 has continuously rolling development distributions since its introduction in June 2020.

For a complete list, check these links for ROS1 supported platforms and ROS2 supported platforms.

ROS Basic Concepts

Some of the basic concepts of ROS, as shown in Figure 1, are packages, nodes, topics, services, and messages.

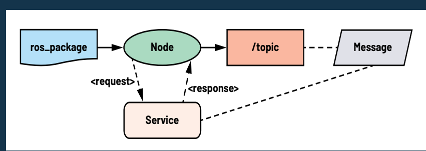


Figure 1. A ROS basic data flow.

Note: The ROS basic concepts discussed in the following paragraphs are similar for both ROS1 and ROS2.

Package

The ROS package is the main organization system of ROS programs or nodes.

This is the most atomic build/release item in ROS. When creating a ROS package, it's important to set up a dedicated ROS workspace. This workspace is called the catkin workspace, wherein catkin is the official build system of ROS.

Revolutionizing Logistics: The Age of AI Truck Loading Robots


Image credit: Andrey Suslov/Shutterstock.com

By Rudy Ramos for Mouser

Before products are displayed on a supermarket shelf or dispatched from a post office, they must be transported from their point of origin to their destination. The method of transportation—whether by land, air, or water—depends on factors such as weight, volume, type of cargo, and destination.

The logistics sector is a cornerstone of the global economy, serving the needs of nearly all industries. Since the start of the Industrial Revolution, the need for transporting manufactured goods has skyrocketed, fueling economic growth. Key players in this industry include DHL, UPS, and FedEx. Moreover, online giants like Amazon are delving into logistics to enhance their supply chains.

According to Statista, "In the fiscal year of 2022, the FedEx Ground segment of FedEx Corporation delivered just over three billion packages." With such a high volume of shipments moving globally every day, the logistics of transporting cargo is getting a much-needed hand from robotics technology.

Operation

An essential step in transporting freight is loading and unloading the semi-trailers used by the key players in the industry. Standard dry van semi-trailers are designed to transport dry products due to the fully enclosed nature of the trailer that protects freight from the elements. Palletized products are the most common freight as they can be efficiently and safely loaded or unloaded with a forklift or pallet jack from a dock.

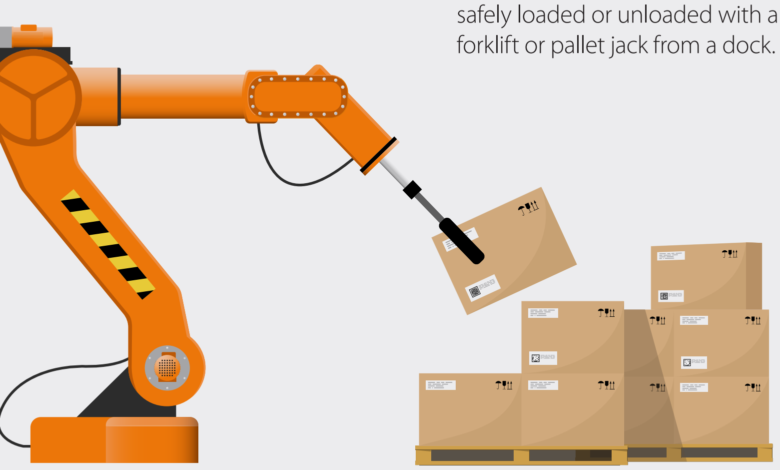


Image credit: AliaksaB/Shutterstock.com

These semi-trailers vary in length but are usually from 15-17m and can have a pallet count of 24 to 26 with a cargo weight of approximately 20,000kg.

But palletizing products is not always possible, and sometimes trailers must be loaded manually, package by package. This is an exhaustive labour-intensive operation that often requires two workers and the use of a telescopic truck loading conveyor.

As personnel strive to meet demanding loading and turnaround requirements, shipping companies will need to spend extra time and extra dollars maintaining truck loaders, which constitutes a significant drain on resources.

Additionally, in high-traffic shipping areas, such as loading docks, collisions between forklifts and loading equipment are almost inevitable, and these machines' repair costs can easily impact the bottom line.

Well, all this is about to change. With the introduction of artificial intelligence (AI) truck loading robots, the dawn of a new era in semi-truck loading promises to radically alter how semi-trucks are loaded.

Novel Robotics

Dexterity AI and FedEx unveiled a first-of-its-kind robotics trailer loading technology. These dock-ready smart AI robots (Figure 1) will make short work of semi-trailer loading and unloading and integrate easily with existing dock infrastructure. Their dual-arm design makes it possible to provide twice the pick rate and full truck reach. This dock-ready system will autonomously drive in and out of the trailers as it loads or unloads. Additionally, there is no pre-sequencing or look ahead required, and conveyable staging is not needed. Small boxes and big boxes. Stiff cardboard and soft deformable plastic.

This AI robot system can turn them all into densely packed trailers, turning random boxes into stable walls.

Figure 1: AI robots offer a new path to efficiency with groundbreaking trailer loading technology (Source: WS Studio 1985/stock.adobe.com)

How MCUs increase system performance in robot motor control designs

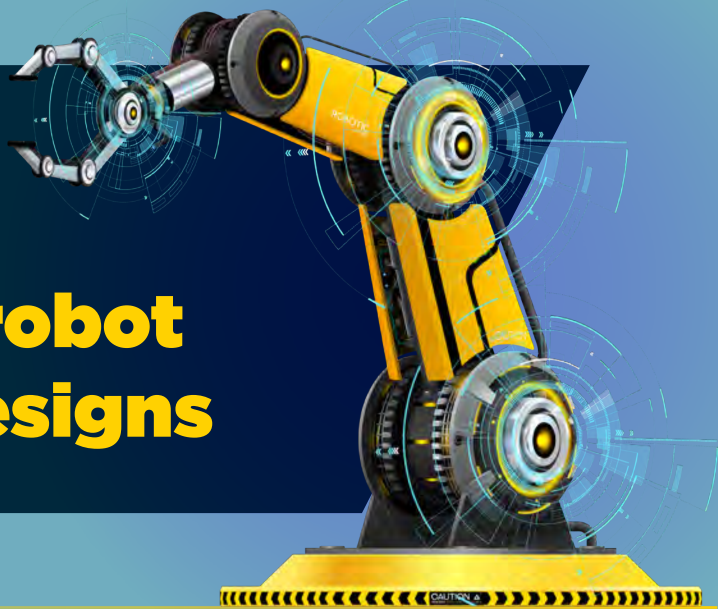


Image credit: Gorrilar Vector/Shutterstock.com

Contributed by Texas Instruments

Robotic systems automate repetitive tasks, handling complex, powerful motions and working in environments that are dangerous or harmful to humans. More integrated, higher-performance microcontrollers (MCUs) enable higher power efficiency and smoother and safer motion with much higher accuracy, therefore increasing productivity and automation. For example, higher accuracy – sometimes within 0.1mm – is important for applications that handle laser welding, precision coatings, or inkjet or 3D printing.

The number of axes in a robot arm and the type of control architecture needed (either centralized or distributed, also known as decentralized) will ultimately determine the right MCU or motor-control integrated circuit (IC) for that system. Modern factories use a mix of robots with varying numbers of axes, defined as the degree of movement and rotation across x, y or z planes, in order to handle different phases of manufacturing; therefore, there are various control architectures throughout the factory floor.

It's important to choose MCUs with additional performance headroom to provide future scalability, as well as the capacity to support add-on features.

Planning for scalability and additional features can reduce cost, time and complexity during the design process.

In this article, I'll explore centralized and distributed motor-control architectures and the design considerations for the integrated, real-time MCUs enabling them. To read about key motor drive design challenges for robotics, read our technical article, "How motor drive innovations are helping solve robotic movement design challenges."

A centralized architecture

In a centralized system (Figure 1), one MCU controls multiple axes. This approach is effective at handling heat dissipation in higher-power motor drivers (typically above 2-3 kW) that require large heat sinks and cooling fans.

In this architecture, position data is usually acquired externally through a resolver board or aggregator connected to the encoders. Typically, in this architecture, multiple power stages are on the same printed circuit board or in close proximity; thus, one MCU can control multiple axes.

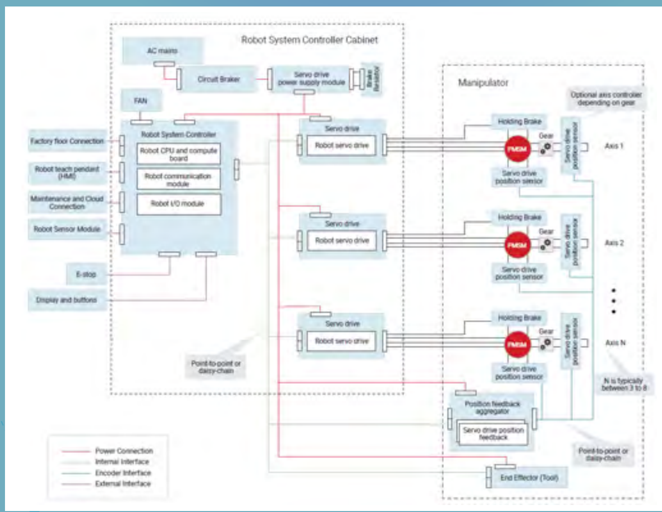


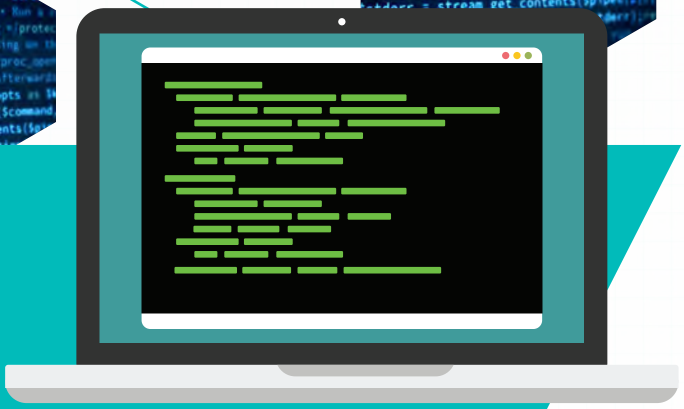
Figure 1 Block diagram of a centralized motor-control architecture for multiple-axis systems

This approach simplifies the real-time control and synchronization between multiple axes, as there is no need for longer communication lines between multiple motor-control MCUs. The motor-control MCU/microprocessor in a centralized architecture mainly requires high-performing real-time processing cores such as Arm® Cortex®-R5F cores or digital signal processors, real time communication interfaces such as EtherCAT, and enough pulse-width modulation (PMW) channels and peripherals for voltage and current sensing. MCUs such as the AM2432 can enable scalable multiple-axis systems and offer real-time control peripherals for as many as six axes and real-time communications in a single chip.

In the past, field-programmable gate arrays or application-specific integrated circuits were primarily used for centralized motor control in automated systems. However, the high level of integration and cost-effectiveness of Arm Cortex MCUs can help you meet your system's performance requirements while also enabling design scalability and flexibility.

While centralized control architectures can meet the performance and efficiency design requirements of high-powered automated systems such as heavy-payload industrial robots, they do require additional cables from the cabinet to the mechanical motor in the joint, as well as from the position sensors to the aggregators. Those wires do not only cost more, they also wear out and require maintenance.

Embedded Linux



This month, Adam Taylor begins a new series of articles on embedded Linux. Over the next six months, Taylor will take a deep dive into the subject, covering its basic elements, looking at development and debugging, working with common interfaces, networking and communication, and concluding with advanced elements such as Real Time and AI / ML. He begins with an introduction:

What is embedded Linux and why is it critical in our world?

It can be argued the world is increasingly software defined - there are over 180 billion Arm-based devices deployed according to estimates from Arm Holdings alone. Each of these devices requires software to control and enable its functionality and to allow interaction with the user. For smaller processors, the development may use bare metal or a simple real time operating system (the subject of a future series), however, as tasks are required scale and the application needs more complex functionality such as human machine interfaces, network stacks and files systems capable of running easily across multiple cores, we need a more powerful and capable operating system. This is where embedded Linux comes into play as it provides developers with a very scalable solution, of course supports multi tasking, and allows for multiple users.

Obviously, embedded Linux is a derivative of Linux created in the early 90's by Linus Torvalds. Linux gained in popularity due to its free and open source nature, offering a good alternative to the proprietary Operating Systems in use at the time. This popularity naturally led to developers porting Linux to embedded solutions which are more resource constrained.

Although it happened naturally, running Linux on embedded solutions makes a lot of sense for most applications. It also creates a wide skills base of developers who are familiar with the operating system, thus reducing the barrier to entry for development. No longer do developers have to learn a new operating system each time they change roles or projects.

One of the most famous examples of an embedded Linux system is the Raspberry Pi, which has been used to create all of the examples and screen shots in this series.

In this series we are going to look at how we can work with embedded Linux on microcontrollers. However, before we can do that we first need to understand the basic elements of the embedded Linux architecture.

At the heart of the embedded Linux system is the Kernel, which is responsible for managing the CPU (s), memory and peripheral devices. Looking deeper, the kernel is also responsible for a diverse range of functions; perhaps the most important is process management, the scheduling of processes and threads across the available CPUs. The allocation of CPU time for a process or thread will depend on the scheduling algorithm used. Along with scheduling the running of processes, the kernel also provides life cycle management for processes and threads enabling creation and termination.

To Kernel also helps ensure user space applications cannot corrupt the other programs should they experience issues. The kernel manages the memory and, crucially, the virtual memory which is an abstract from the real physical memory. It also enables us to use more memory than is physically available thanks to paging and swapping to disk. Visualization of the memory also ensures applications within the user space are isolated from each other and ensures they cannot easily access physical memory.

Lighten up

Residential buildings, workplaces, and roads have experienced a remarkable change thanks to the implementation of LED lighting.

LED bulbs save money in terms of reduced running costs in comparison to outdated filament bulbs, and they have significantly longer lifespans, often lasting tens of thousands of hours. And LED lighting is not limited to traditional applications and is driving progress across many industries. Matrixes and dynamic animated solutions are being employed across industrial and smart home goods to provide visual feedback.

In fitness and healthcare, LEDs are playing a crucial role in optical vital sign monitoring in smartwatches and other wearables. As a result, LEDs are now widespread in their use and, for many projects, are an essential electronic component.

New LED product highlights available from Mouser include:

LUXEON 2835 commercial mid-power LEDs from LUMILEDS are designed for commercial indoor lighting solutions, especially suiting applications when the key development metrics are lumens per watt and lumens per cost. The LUXEON 2835 range offers best-in-class flux, colour consistency, robustness, and reliability for commercial indoor luminaires, while an industry-standard footprint allows easy drop-in replacement. With a variety of correlated colour temperature (CCTs) and colour rendering indexes (CRIs) available, the 2835 lineup is suitable for applications such as panel/soft lights, linear lighting applications, and troffers. All 2835 LEDs are RoHS compliant and offer exceptional illumination with a wide 160° total included angle.

With its compact size and versatile design, the APBA2006 RA bi-colour SMD LEDs from Kingbright are ideal for a wide range of applications, such as backlights, status indicators, smart appliances, and wearables, including medical devices. The SMD LEDs exhibit low power consumption, high-intensity bi-colour output and robust construction, with a working temperature range of -40°C to 85°C and a moisture sensitivity level of 3.

These bi-colour SMD LEDs are available in five different colour combinations, are halogen-free, RoHS-compliant, and offer a wide viewing angle of 140° from the side-emitting diodes.

The low-power MULTILED® SFH 7018x from ams OSRAM are engineered to support the ever-expanding digital diagnostics and smart health wearable devices. Featuring a cutting-edge three-in-one multi-emitter, the SFH 7018x enables precise and reliable monitoring of heart rate and SpO2 levels in green, red, and infrared wavelengths. The sophisticated 2-cavity design minimizes interference between the green chip, red chip, and IR chip and allows for optimal placement of the light sources in relation to their respective photodiodes. Two RoHS-compliant variants are available, with the SFH 7018A optimized for low Vf at a high current.

The LP5813 4 × 3 matrix RGB LED driver from Texas Instruments is a synchronous boost RGB driver with autonomous animation engine control. The LP5813 is designed for LED animation and indication in devices ranging from portable and wearable electronics to smart devices and industrial HMI. The integrated synchronous boost converter ensures high efficiency and consistent LED brightness across a broad range of operating voltages, and supports LED forward voltages of 3V to 5.5V in steps of 0.1V. The autonomous animation engine can substantially reduce the real-time loading of the controller. Each LED can be individually programmed to create vivid and dynamic lighting effects, with synchronization of effects across multiple devices possible.

Click for More Information

Auto eval kit

Infineon Technologies' KIT_T2G-B-H_LITE TRAVEO T2G evaluation kit enables the development of customer applications using the TRAVEO T2G Body High MCU.

The MCU is a true programmable embedded system-on-chip, integrating up to two 350MHz Arm® Cortex®-M7 as the primary application processors, a 100MHz Arm Cortex-M0+ that supports low-power operations, up to 8 MB flash, 1 MB SRAM, a Secure Digital Host Controller (SDHC) supporting SD/SDIO/eMMC interfaces, and programmable analogue/digital peripherals that allow faster time-to-market.

The new evaluation kit is targeted at automotive electronics applications such as body control modules (BCM), heating, ventilation, and air conditioning (HVAC), motor control, lighting, and automotive gateways.

Click for More Information