White Paper

High-Performance Motor Control Made Quick and Simple with Optimized Arm Microcontroller, Tools, and Code

October 2020

Abstract

Automation in our homes, the built environment, and industry, is increasingly reliant on the use of electric motors. Where once we did the laundry by hand, walked upstairs rather than riding the elevator, and relied on muscle power to run industrial processes, now we use motors. This trend is accelerating as what were standalone devices gain network connections and become part of the Internet of Things (IoT); think of garage door openers, smart washing machines, and connected production lines as examples.

Design engineers who are adding electric motors to existing products or creating entirely new products that apply electric motors in novel ways, face familiar tradeoffs as well as some new challenges. The familiar tradeoffs include pressures to reduce time-to-market, constrain overall design-engineering effort, and minimize end-product costs. New challenges for some will include the learning curve involved in implementing efficient motor-control strategies in their products for the first time. All will face the challenge of ensuring the safe and secure operation of their designs, particularly those that have a network connection.

Given these challenges, it makes sense to look for solutions that draw on as much established knowledge and best practice as possible. Renesas is combining its experience in applied motor control, particularly for consumer and industrial applications, with the power of the Arm processor ecosystem, to offer a hardware, tools, and software solution to the challenges outlined above. The solution is in three parts: An Arm-based microcontroller with extensive specialist features for high-precision motor control; evaluation board with which designers can rapidly explore practical motor-control strategies; and a software development environment in which they can use worked examples and extensive libraries to produce motor control code.

The Processor

The Renesas RA 32-bit MCU Family, built on Arm® Cortex®-M core architecture, offers a wide range of products supporting different application needs.

The RA6T1 group supports motor control applications, providing Cortex-M4-level performance with ready-to-use motor control software. Its PWM timer and advanced analog circuits make it easy to implement high-precision and safe motor control, while minimizing the design's bill of materials (see Figure 1).

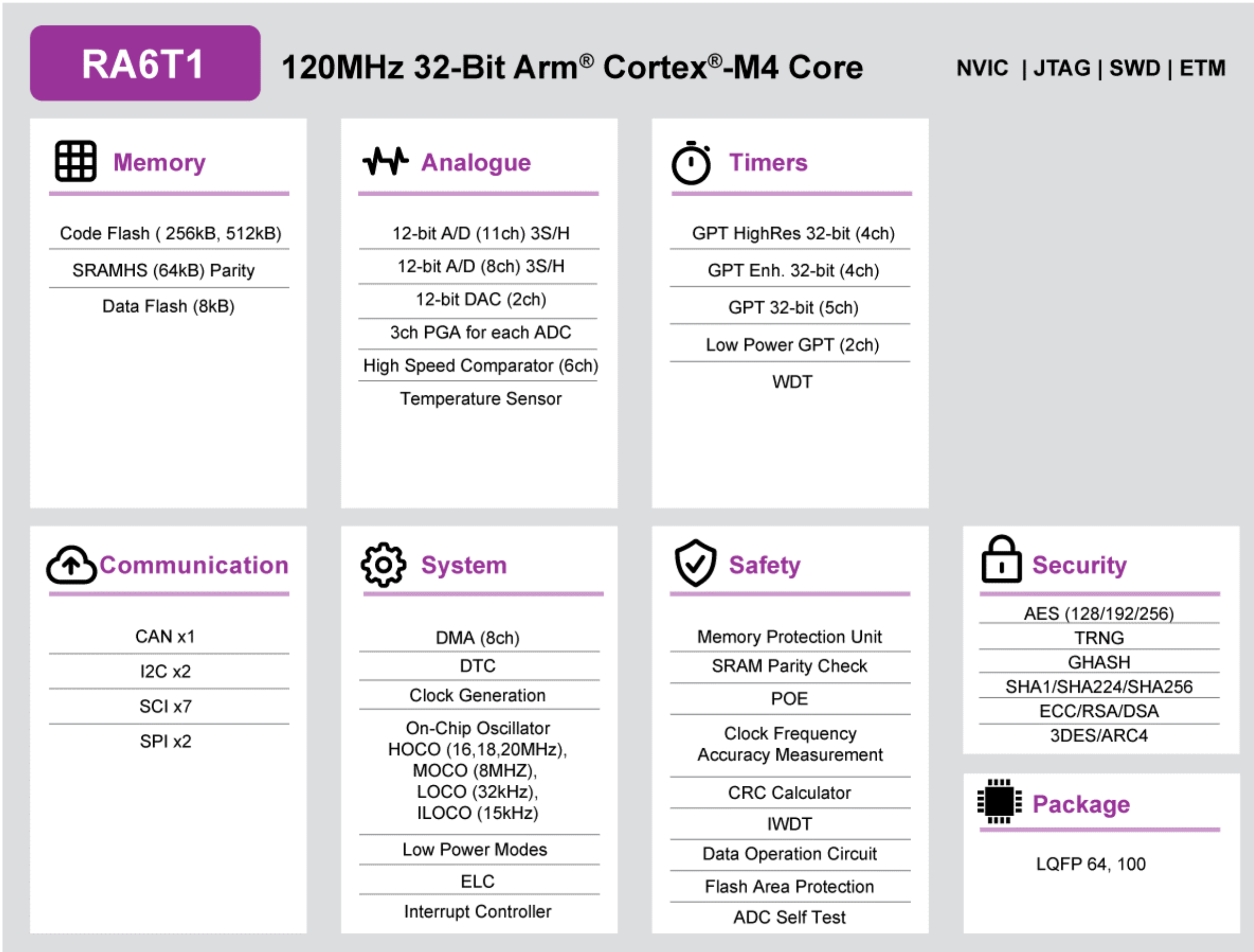

Figure 1: The basic specifications of the Renesas RA6T1 group microcontroller

The RA6T1 includes a 120MHz Arm Cortex-M4 processor with its associated floating-point unit, for high-accuracy computation. The part has 64kB of high-speed parity-protected SRAM, and either 256 or 512kB of flash memory for code. There is also 8kB of data flash memory.

Analog features include an 11-channel 12-bit analog to digital converter (ADC), three of whose channels have sample-and-hold circuits. There is a separate 8-channel 12-bit ADC, again with three sample-and-hold channels. Having two ADCs makes it simpler to implement feedback loops when designing systems that control two motors. Each ADC has a 3-channel programmable gain amplifier. The microcontroller includes a 12-bit digital to analog converter, a 6-channel comparator, and a temperature sensor. The inclusion of these features reduces the bill of materials for motor control designs.

The RA6T1 also has an extensive range of timers, useful as building blocks for creating pulse-width modulated (PWM) motor control waveforms. There is a high-resolution, 4-channel 32-bit PWM timer, for which the duty ratio of its output waveform can be adjusted in 260ps increments to ensure high-accuracy, repeatable waveforms. There is also an enhanced 4-channel 32-bit PWM timer, a 5-channel 32-bit PWM timer, and a 2-channel low-power timer.

The PWM timers have seven operating modes (see Figure 2) and can be synchronized to enable three-phase complementary outputs. They can also be programmed to implement 0% and 100% duty operation, to improve the responsiveness of the motor control switching element when the load it is switching suddenly changes and the amount of current flowing has to be adjusted rapidly.

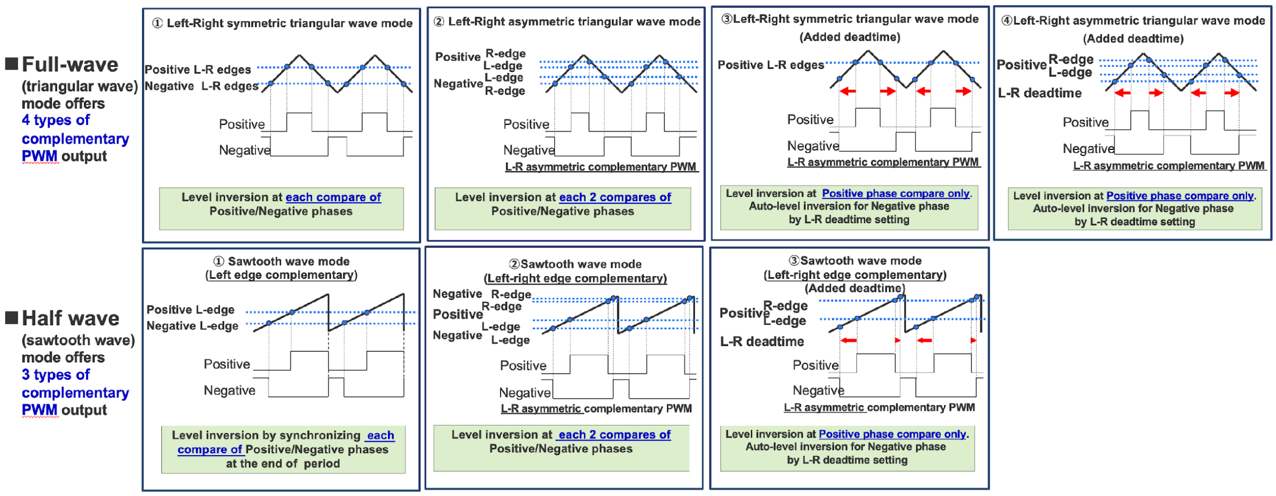

Figure 2: PWM timer support for seven basic waveforms simplifies the implementation of motor control algorithms

The RA6T1 also has a double-buffering arrangement for the register that holds the value to define the timer's output signal. In standard motor controllers, a comparator checks the value of the timer register every time it is set, to see whether it has reached its peak value. If it has, an interrupt is sent to signal that it is time to start decrementing the value. In the RA6T1's double-buffered arrangement, once the timer's output value has reached its maximum, the down count is automatically transferred to another register, which only sends an interrupt when it reaches the trough value of the waveform. This approach halves the number of interrupts that need to be serviced by the motor-control software to shape the timers' waveforms.

Safety features of the microcontroller include a Port Output Enable (POE) pin, which controls the timers' outputs. Toggling this signal shuts down the microcontroller's PWM outputs, independently of the CPU, enabling designers to include robust safety measures in their motor control schemes. The POE signal can be asserted in four ways: by feeding motor-current sensor signals to the RA6T1's ADC and comparator to detect abnormal motor currents; by detecting abnormal motor control waveforms; by sensing if the microcontroller's clock has stopped; or by the assertion of the POE pin.

The RA6T1 also has other systemic safety features, such as a memory protection unit, a way to check the accuracy of the clock frequency, self-test for the ADCs, a CRC calculator, and protection for the flash memory.

Communications features include a CANbus and extensive serial interfaces, while system features include a clock generator and various low-power modes.

The Development Boards

Renesas has produced a pair of development boards to support the use of the RA6T1 microcontroller in motor control applications: a CPU card (Figure 3) and an inverter board (Figure 4).

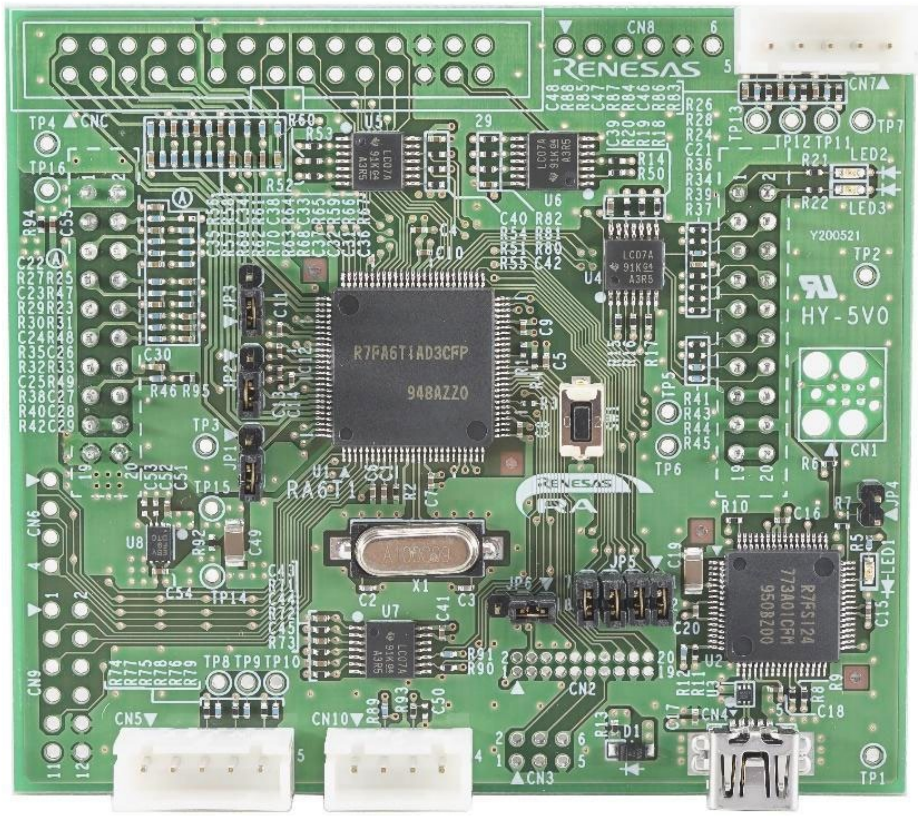

Figure 3: The CPU card for the RA6T1 development kit

The CPU card can be powered over USB, or by connecting it to the inverter board. It includes the 512kB variant of the RA6T1 processor, in its full 100-pin package (the processor is also available in a smaller package with fewer pins). Debug is supported over a USB connection to the onboard J-Link debugger, or via through-hole PCB tracks for connection to an Arm debugger. The board has connections to two motor controller boards, one through an installed connector and the other via through-hole PCB tracks. There are also connectors for Hall sensors, encoders, and an SCI connector to link to the Renesas Motor Workbench development tools.

The accompanying inverter board, as part of the RSSK Evaluation System for BLDC Motor, is pictured in Figure 4 (below).

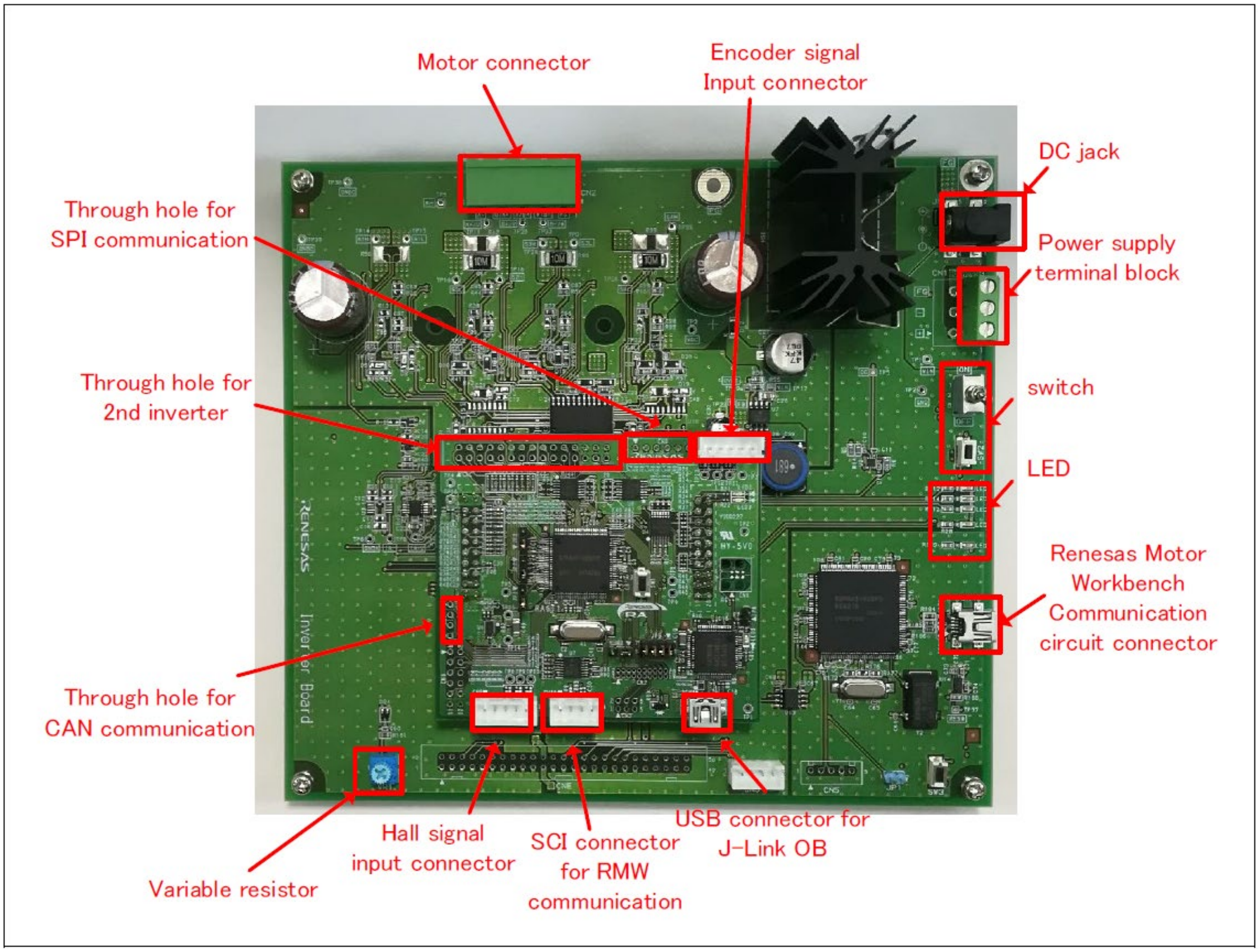

Figure 4: The inverter board (includes the RA6T1 CPU card)

The inverter board will operate from 24V to 48V DC at up to 250W and will output up to an AC current of 5A. Switching frequency range is 2kHz to 20kHz. There are detection circuits for DC input voltage, 3-phase output voltages and currents, and a 10A overcurrent detector. Interfaces include a USB connector, CPU card connectors, the power input, and the motor drive output.

The two cards connect as seen in Figures 5 and 6.

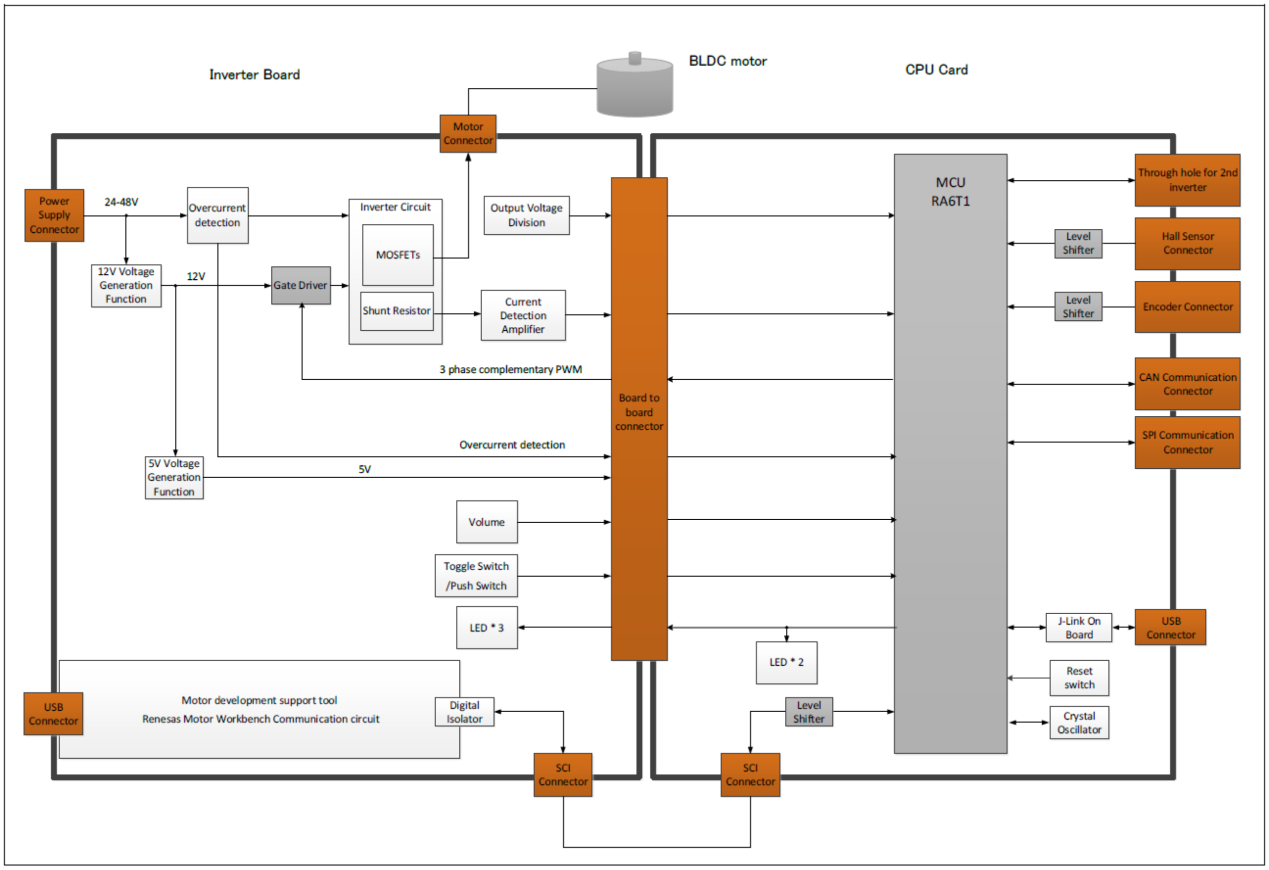

Figure 5: Block diagram of the interconnection between the CPU and inverter board

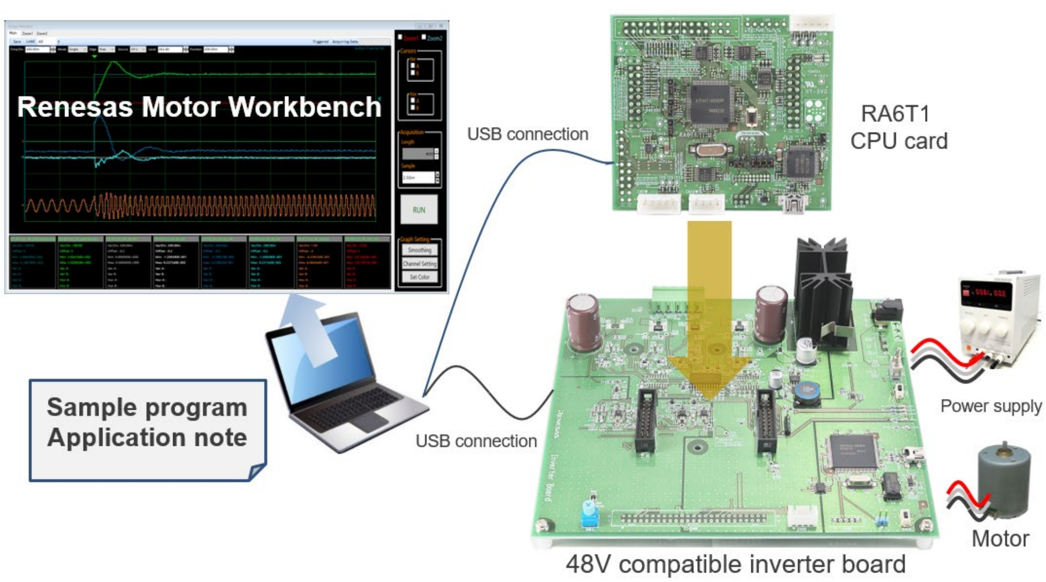

Figure 6: Schematic of the RA6T1 development system

The Motor Control Development Environment

The development environment for solutions based upon the RA6T1 draws on the combination of Renesas' experience in motor control and the open ecosystem that Arm has developed over the past 30 years. The combined offering includes on-chip debug facilities, integrated development environments (IDEs), compilers from multiple sources, and the Renesas Motor Workbench 2.0, a graphical user interface that enables users to develop, analyze, and tune their motor control strategies.

On-chip debug

The Renesas RA Family devices are supported by a variety of on-chip debuggers, which take advantage of the powerful Arm CoreSight debugger IP included in their design. These include the extensive Segger J-Link Family of debuggers, as well as Renesas's own E2/E2 Lite debuggers. These offer support for SWD and JTAG connections and will provide ETM and ETB trace data.

The debuggers also integrate with the e² Studio IDE offered by Renesas and third parties such as Keil and IAR Systems. Compilers are available from GNU, Arm, and IAR.

IDEs

The Renesas e² Studio IDE is based on the Eclipse open-source C/C++ development tooling. It provides an advanced GUI to guide users through device selection, pin mapping, project generation, project building, debug and project management. The tool integrates GNU compilers, a powerful C/C++ code editor, and headless build options for continuous integration and unit testing. e² Studio also integrates with open-source project-management extensions such as GitHub and other version control systems.

The Renesas e² Studio IDE has a standalone Flexible Software Package (FSP) configurator, which can be used to select and prepare FSP driver configurations for use by third-party IDEs.

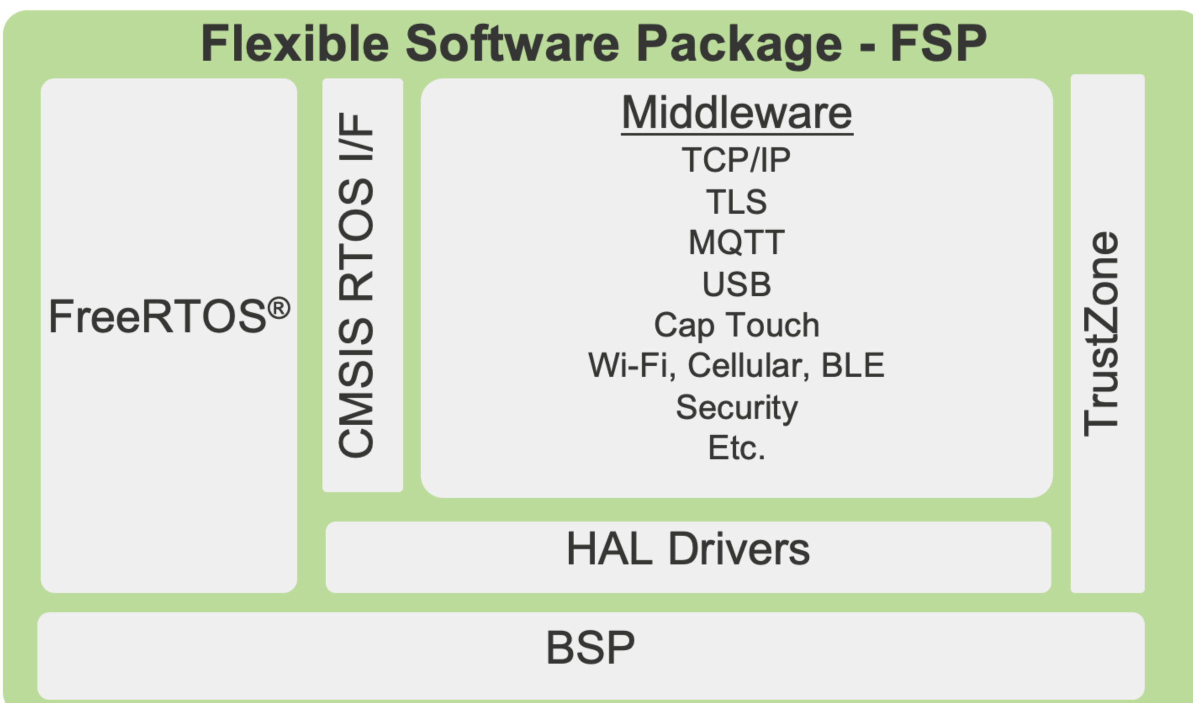

Figure 7: e² Studio can configure the right mix of enabling software

Access to this FSP configurator ensures that application development can begin immediately, based on the right choice of high-performance, efficient drivers that are also provided with source code. The FSP configurator also makes it easier to implement middleware such as the communications and security features shown in Figure 7, all of them compliant with Arm's Cortex Microcontroller Software Interface Standard, a vendor-independent hardware abstraction layer for microcontrollers based on Cortex processors. This ensures access to Arm's open software ecosystem, flexible use of legacy code, and easier collaboration with third parties.

Renesas Motor Workbench 2.0

The Renesas Motor Workbench 2.0 enables users to take advantage of the company's experience in motor control. It is particularly useful in implementing sensorless vector control of brushless DC motors at high efficiency and low cost. This complex control method can be challenging for beginners to work with, but the tool includes standardized procedures and tutorials to simplify the process. Renesas also provides example code blocks through its online support pages.

Although the speed of the motor can be controlled directly from a variable resistor on the motor control board, connecting the board to the Renesas Motor Workbench 2.0 software running on a PC will provide more insights into the way the motor is behaving.

Figure 8: One view of the Renesas Motor Workbench 2.0 user interface

Motor control parameters can be displayed in real time on the PC in an oscilloscope-like format, with various triggering, offset, and display options. The Renesas Motor Workbench 2.0 enables users to dynamically read and write up to 225 variables from the microcontroller at once to see how changing control parameters affect the motor's operation. This enables rapid evaluation and optimization of the control algorithm. There are also facilities for user-defined batch processing of the results.

The Renesas Motor Workbench 2.0 software also has features to enable automatic tuning of a motor-control strategy, both by characterizing the motor in use and adjusting the parameters that control it. The tool can automatically measure motor characteristics such as resistance, inductance, induced voltage constant, inertia and friction, based on rated current, rated power and pole pair details input by the user.

The tool can automatically adjust PI control gain levels to achieve particular currents, speeds, or positions, and adjust estimating gain in sensorless vector control strategies. Renesas Motor Workbench 2.0 also supports fine adjustment with manual PI control tuning and can output results in various formats.

There are mathematics functions to enable new waveforms to be derived from input values; for example, to display the deviation between the motor speed that the algorithm is trying to achieve with the actual motor speed. The tool also has programmable status indicators to highlight how key variables are changing with multiple conditions for each variable that is being monitored in this way.

Conclusion

The increasing level of automation in homes, buildings, and industry, is leading to much greater use of electric motors. Designers who are already using electric motors face the requirement of optimizing their implementations.

For engineers who are adding them to their designs for the first time, understanding how to do so quickly and effectively becomes a challenge.

Renesas offers a combination of an optimized microcontroller, ready-to-use motor control software, and a tool chain that draws on open source and proprietary offerings, and takes advantage of Arm's open development ecosystem. The Renesas Motor Workbench 2.0 adds to this offering by enabling users to take advantage of Renesas' experience of advanced control strategies. The overall combination of features will help users increase their debugging efficiency and shorten their development cycles as they implement products using electric motors based on the latest Renesas RA6T1 group.

Learn More

1. RA6T1 Product Page
2. RA6T1 Motor Control Evaluation System
3. RA Partner Ecosystem
4. Flexible Software Package (FSP)
5. RA Family of MCUs