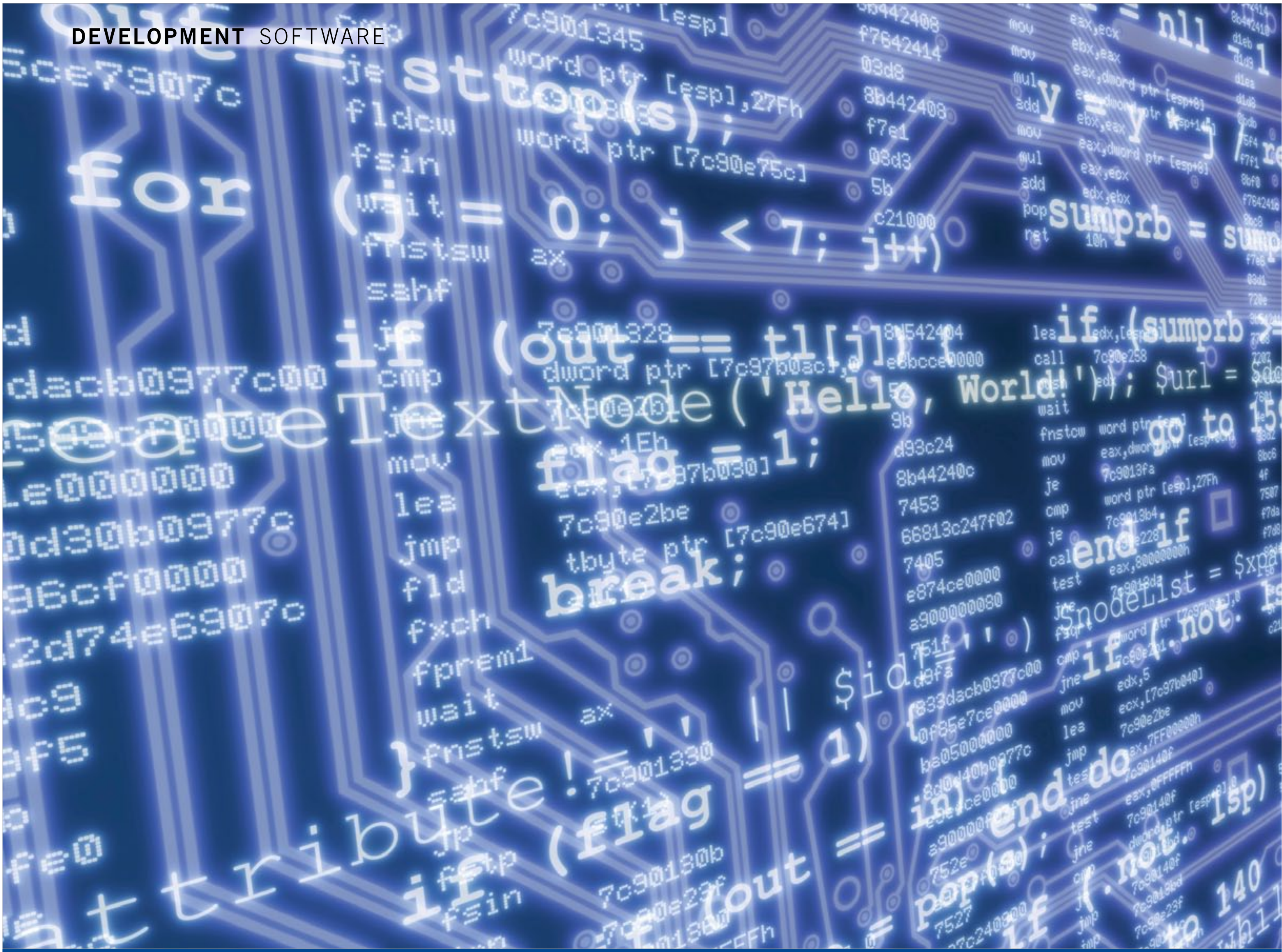


FIGURE © tyndyra / iStockphoto.com

SYSTEMS SOFTWARE
FROM AIRCRAFT TO AUTOMOTIVE NICHES

In the past few years, many developers have come to appreciate the advantages of open source software: among them – beside cost advantages – the open implementation, certifyability, and adaptability of functionalities to accommodate individual demands. For many companies, the decision to use a specific operating system platform is of strategic importance as it ought to be stable for many years and many follow-on projects. The principle of free, open-source software provides independence from the decisions of operating system providers. RTEMS therefore wins market shares not only in the aerospace sector but also in industrial and automotive applications, as the company embedded systems describes.

AUTHOR

THOMAS DÖRFLER
is Managing Director
of embedded brains
in Pucheim near Munich
(Germany).

HISTORY OF RTEMS

RTEMS originated with the US Army. It was developed for control purposes to replace a commercial real-time operating system. Its source code was published and made available under a modified GPL (general public license) relatively quickly. Since then, its evolution has focused on user requirements, while still building on the pillars of real-time capabilities, open source, and availability free of charge. Many users have now come to appreciate this stability.

FASTER TIME TO MARKET IS A PRIORITY

For years, more and more embedded systems are being equipped with special operating systems. Beside the classic functions, such as multi-tasking and inter-task communication, enhanced functionalities, such as file systems for flash cards as well as USB and network connectivity, continue to gain importance. A turnkey operating system facilitates development tremendously. Being supported by ready-to-use services and functions of the operating system, design engineers can redirect their focus to product-specific challenges.

RISK OF CLOSED SOURCE

There could be a nasty surprise if the operating system is suddenly no longer available for one or the other reason: because manufacturers do no longer support individual processor architectures, realign their strategy, or simply vanish from the market. The continued use in existing products is usually ensured but engineers will have to live without bug fixes or functional enhancements. The utilisation of the same operating system in a follow-on product is impossible, however, and the acquired expertise in the operating system and the product-specific source code is then mostly lost.

ALTERNATIVE OPEN SOURCE HAS PROVEN ITSELF

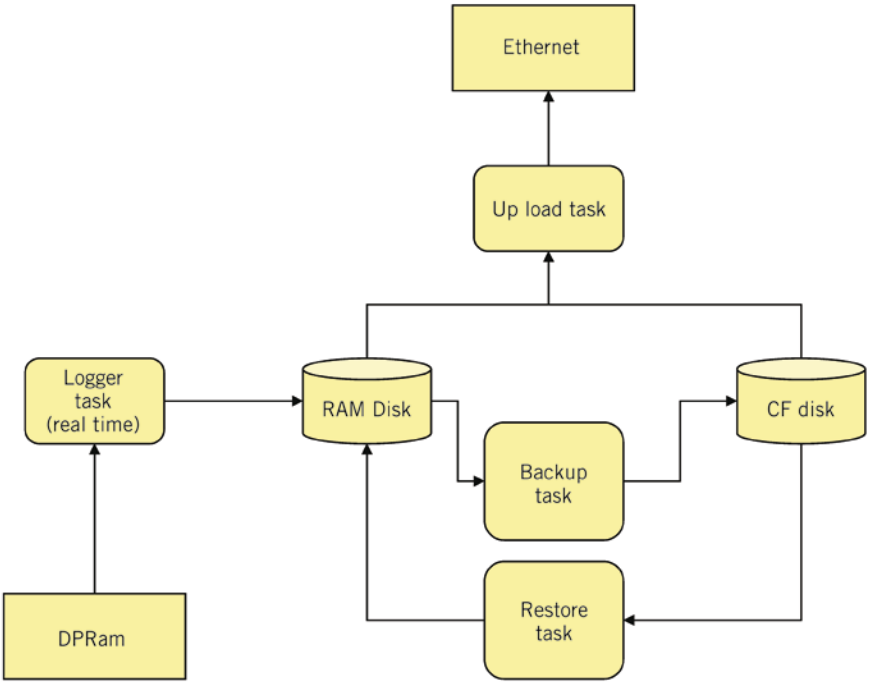
This cannot happen with RTEMS. The open source principle of RTEMS in connection with a free license for use removes this dependency from the system manufacturer. The open source code allows a deeper understanding of the operating system; application development is greatly facilitated, and engineers are encouraged to share their knowledge. While users can access global, commercial support, they do not depend on it. In addition, users have developed many improvements for RTEMS, which have been integrated into the system, and are therefore available to all users and can be continually maintained as part of the operating system.

A PRACTICAL EXAMPLE

As an example for possible applications of RTEMS a data recording system (logger) for racing cars is described here, which the author co-developed. The core requirements were quite sophisticated:
: integration of hardware into the specially designed engine control system
: boot time under five seconds
: data recording rates in the range of 500 to 1000 Kbyte/s
: maximum recording time of up to several hours
: data storage on flash card
: data retrieval via Ethernet with min. 80 Mbit/s
: flexibility for changing deployment scenarios
: CAN bus connectivity.
The hardware was specifically designed for this application. As central microcontroller, Freescale's MPC5200 has been used, which does not only provide all required interface modules but also offers sufficient computing power.

REASONS FOR RTEMS

RTEMS was the operating system of choice after an analysis of requirements, ❶: all APIs required for the necessary functionalities were available, and due to



❶ Base structure of the software

the open source structure it was obvious that the adaptation to the system's specific requirements would be supported.

Further development then followed in several stages:

- The porting of RTEMS to the specific hardware platform and its interfaces was performed by an RTEMS support center.
- The application software structure was planned and then implemented by software developers within the racing teams with support from the RTEMS support center.
- The performance of critical data paths was reviewed and some weaknesses in the underlying hardware structure were identified. Therefore the interaction of the involved software components has been optimised, from the interface drivers and the RTEMS file system support all the way to the application.

The performance optimisation in particular proved to be relatively easy, as the operating system modules are also accessible for analysis. The source code exhibits the general structure and functionality and allows modifications and adjustments. This also allowed the optimisation of the utilised file system's buffer management to handle large files; DMA-capable flash driver support was also improved. The ambitious goals could be achieved in this way.

Here, the advantages of RTEMS' open source principles are exemplified: the improvements to the operating system code performed for this customer's project were adopted into the RTEMS source code repository. On one hand, all RTEMS users can now take advantage of the improvements. On the other hand, the optimisations are available to the customer's project in future operating system versions as well, as they continue to be maintained there.

The developed data recording system has worked successfully for several years now. Its success has triggered discussion about possible uses of this flexible operating system in other, similar application areas.

DETAILS ON RTEMS

The operating system has already been deployed in many market segments [1, 2], to include the automotive market as well as driverless transport systems, satellite technology, telecommunications, medical, robotics, audio, defense technology, and industrial applications, ❷. RTEMS can now be utilised on some 16-bit and nearly all 32-bit architectures, among them PowerPC, ARM, SPARC, MIPS, M68k, ColdFire and Microblaze. 25 Kbyte RAM already suffice for a minimal system.

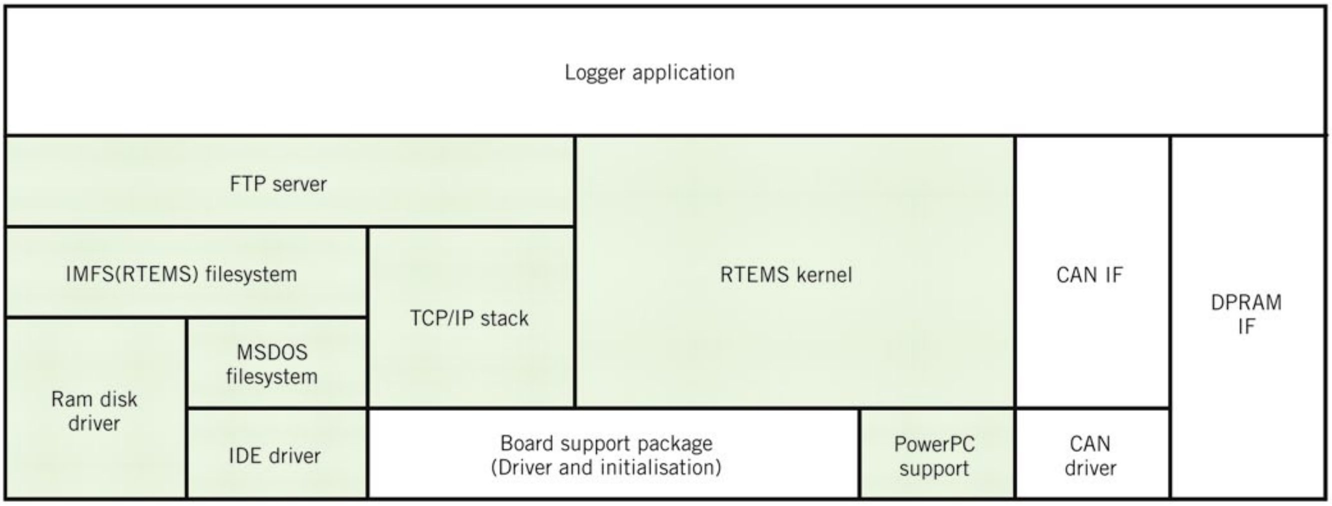
The system core allocates computing time, memory, and I/O resources to various tasks while supporting the POSIX as well as RTEID API. Additional services then provide various file systems as well as USB and network connectivity (Ethernet, CAN and others), among others.

All possible development tools are available; for example, the RTEMS community provides GNU C compilers (gcc) specifically tailored to RTEMS as well as the associated assemblers and linkers (binutils). These tools can also be embedded into integrated development environments (IDE), e.g. Eclipse. GNU debugger (gdb) is commonly used as a debugger but commercial debugging systems with extended capabilities can also be used. The TRACE32-ICD by Lauterbach, for example, offers an extensive RTEMS awareness for the visualisation of RTEMS objects.

Licensing is simple: RTEMS is published under a modified GPL (general public license) V2. The GPL ensures that modifications and improvements to the RTEMS source code are disclosed. On the other hand, the RTEMS-specific modification allows that proprietary source code linked to the RTEMS kernel may still be kept confidential. Control over product-specific knowledge is therefore guaranteed. Yet extensive support is also available and offers openly accessible documentation and a project Wiki as well as mailing lists [4] for an exchange among RTEMS users. Commercial support tailored to specific project needs is also offered.

REFERENCES

[1] http://www.embedded-brains.de/index.php?id=rtems_applications
[2] http://rtems.org/wiki/index.php/RTEMSApplications
[3] http://www.rtems.org/onlinedocs/doc-current/share/rtems/html/
[4] http://rtems.org/wiki/index.php/RTEMSApplications
[5] http://rtems.org/wiki/index.php/RTEMSMailingLists
[6] http://www.embedded-brains.de



❷ Data flow and task allocation