

From Deep Space to Deep Sea



Altreonic NV.
www.altreonic.com

Linden, 1st September 2015

VirtuosoNext™ Press release

Altreonic is proud to announce the release of the version 1.1 of VirtuosoNext™, its high level yet very performant design and programming solution for trustworthy systems development. VirtuosoNext is derived from the formally developed network-centric and distributed OpenComRTOS who's functionality is inherited.

VirtuosoNext adds the capability to apply fine-grain time and space partitioning when the hardware supports it. VirtuosoNext provides safety measures to trap runtime errors without the system coming to a halt. The approach separates the code in a trusted zone (managed by the VirtuosoNext kernel) and an untrusted zone (executing application tasks). Contrary to traditional hypervisor approaches, the code size and performance penalty is minimal safeguarding the real-time response of a traditionally unprotected RTOS as standard priority preemptive scheduling is maintained.

VirtuosoNext was initially disclosed in April 2015 at <http://www.altreonic.com/content/virtuosonext%E2%84%A2-fine-grain-space-partitioning-distributed-rtos>. This new release has also restructured the code resulting in even lower code sizes. On the ARM-M3, the VirtuosoNext kernel measures between 8 to 11.5 Kbytes. On the ARM A9, the kernel measures between 5 to 22 Kbytes.

Supported processors are ARM M3/M4/R4/R5 and the A7/A9/A15 processor family. The user has the option to enable the protection on specified processing nodes in his distributed or manycore target system. VirtuosoNext 1.1 will hence on make the previous and stable OpenComRTOS v.1.6 superseded as its functionality is integrated.

With v.1.1, following new services and improvements were added to VirtuosoNext:

***New!* MoveData service**

Altreonic's VirtuosoNext now makes it easier to build distributed signal processing systems by introducing the new MoveData Service. This service can be described as a distributed version of the memcpy() function. It works the same way when used on a single processing node as when used across distributed processing nodes, a unique distinguishing feature of Altreonic's Virtual Single Processor programming model.

Moreover, the MoveData interaction only returns to the caller once the last byte of the source buffer has been copied which avoids accidental data corruption by the sender

Altreonic N.V. Gemeentestraat 61A bus 1, B-3210 Linden, Belgium Tel.+32 16 20 20 59
BTW 0899.997.375 Argenta 979-1525460-43 Iban BE66 9791 5254 6043 Bic ARSPBE22



Task. The latter can be problematic when using memcpy, especially when the physical copying uses DMA engines.

***New!* Visibility of ISRs in the modelling environment**

Developing custom Interrupt Service Routines (ISRs) has been made easier with the latest edition of VirtuosoNext' Visual Designer. ISRs are now first class Entities of the Application Diagram in Visual Designer. This means that the developer can see the Hub-Interactions by drawing them in the diagram. This has two major benefits: Firstly, it provides the programmer with a complete overview of all interactions in the System, Secondly, inserting the Interactions using the Application Diagram means that the developer saves time and avoids a source of errors and hence improves productivity.

***New!* Split header and data packets reduce further memory requirements**

While VirtuosoNext already requires little memory (the kernel will fit in 5 to 22 Kbytes depending on the target processor), this has been further optimised. The change has separated the service request from its data part meaning that the corresponding data part is no longer allocated when not used. For example Tasks or ISRs that do not send data save 1KB per unused data packet, reducing further the memory requirements. Furthermore, this change improves the performance as it reduces the number of memory copy operations in the Port- and FIFO-Hubs. These hubs now simply switch data parts instead of performing the copy operation. This also makes their computational complexity independent of the amount of data that is present in the Packet, i.e. $O(1)$ instead of $O(N)$. The new Header- and Data-Packets results in VirtuosoNext becoming even more suitable for deeply embedded systems. The end result is less power consumption and thus a longer battery life. The combination of the network-centric architecture, small code size and the support for low latency distributed real-time functionality make VirtuosoNext a very attractive offer to develop IoT applications.

VirtuosoNext™ is available under an Open Technology Licensing scheme.

For more information, contact:

Eric Verhulst

eric.verhulst (@) altreonic (dot) com

+32 - (0)477 - 708 - 339