~ Sli.do Q&A ~
CoRA-ZynQ

Automatic tools provides SW and FPGA functions but how is the interconnection in the integrated SoC? Is still the user hand coding or integrating the modules?

The toolchain user will only have to define his/her function at application level (e.g. in C or Simulink) and automatically obtain SW and HW implementations of that function. All middleware interconnection (SW and HW parts) are transparently generated by the toolchain, based on the AADL model and ASN.1 data types defined for the function's interfaces.

What would be the advantages over GOMSpace Nanomind 7000?

The selected HW target in this activity was the Xilinx 7000 SoC as it appears in the Xilinx ZC706 board. The Nanomind 7000 apparently sports the same chip – and since TASTE and Bambu are open-source and the resulting designs have no external dependencies, with some adaptations the toolchain should be able to easily support any Zynq7000-based targets – including the Nanomind 7000.

Why if Mathworks tools for c autocoding is used the FPGA Mathworks HDL Coder is not used? Is Bambu in your experience better for this task?

Indeed the VHDL from Simulink models could be generated directly using HDL Coder - but (a) this wouldn't cover generic C code, only code generated from Simulink and (b) this is a proprietary tool. Bambu is open-source, FPGA vendor independent, and can be used on any C codebase (even code using pointers). It was therefore chosen in the preceding CORA project and - is already bundled into the TEC-SWT's TASTE distribution, for direct (and completely automated) use from any user of TASTE.

Will the tools be available for free? Is everything already integrated in the current release of TASTE ?

The toolchain (TASTE and implemented extensions) are available for free, yes (https://taste.tools). Currently, a specific TASTE Git branch (CoRA-Zynq) is being used to develop the code for the activity. It is under continuous integration and testing and will be soon merged into the master branch – at which point any user of TASTE will automatically get this via a call to "Update-TASTE". The TASTE VM will also get a new release soon (within the next month - based on Debian Buster) and will contain the CoRA-ZynQ functionality.

Can we also use FreeRTOS?

Currently is not supported, but nothing prevents the toolchain to be extended to use/reuse a different RTOS. A new target could therefore be added in the TASTE Deployment View (e.g. "FreeRTOS on ARM"). In this activity, we chose to use "RTEMS 5.1 on ARM" because of its Space representativeness - RTEMS is the most commonly used OS in our missions. It's also the OS that the Agency pre-qualified (back in the era of RTEMS 4.8) and is qualifying now again in its SMP configuration (RTEMS 5.1).

Is it needed any modification in the source models depending on whether target processor is the ARM or the FPGA part of the Zynq?

No. Both the SW and HW forms are generated from a single unique source model/code.

Does the TASTE model allow for timing constraints?

TASTE allows to define non-functional SW properties related with timing, such as the period or the worst-case execution time for a certain interface. On the HW part, timing constraints, such as clock design, are currently configured and handled transparently at middleware level. However, nothing prevents that more (advanced) configuration properties are made available at application level to the user to, for instance, configure the clock period option passed to Bambu for high-level synthesis. That would allow the user to obtain and compare HW designs running at different frequencies - and use variable amounts of HW resources. Having said that, note that timing constraints at the HW level are also a matter usually left at the hands of the designer – and since the TASTE outputs are not cast in stone, but instead follow the "target modelling tools conventions" (in this case, Vivado's) the designer can modify the generated design at will - if he so chooses.

What about Zynq7000 radiation tolerance? For which space missions can it be used?

It makes much more sense to speak about radiation tolerance at the level of specific board designs. On that front, Zynq-7000 based designs are already being considered and used in CubeSat missions and in some instruments. Both CNES and NASA have demonstrated interest in using such designs.

Is TASTE already available for evaluation?

TASTE is fully available, for free, in open-source form (https://taste.tools – with the complete source behind all code generators available here: https://gitrepos.estec.esa.int/taste/taste-setup/).