

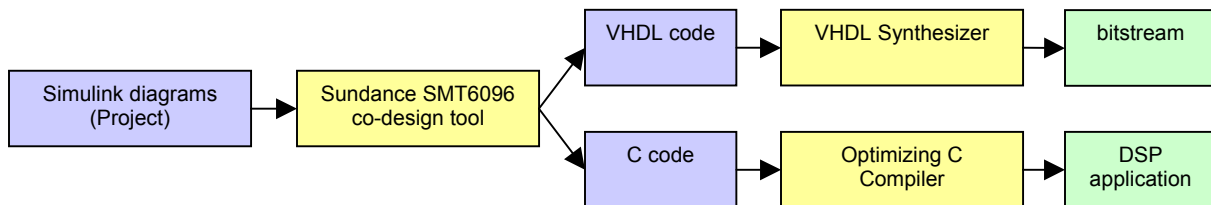
## *HW/SW Co-Design of Digital Signal Processing Systems is as easy as ...*

Rapid prototyping systems employ various components such as CPUs, FPGAs, ADCs and DACs. The *Developing Digital Signal Processing System* (DSPS) therefore requires designing many SW, digital HW and analog HW subsystems, which tightly interact. Design validation requires simulating the interacting subsystems as a whole; this invariably leads to appropriate co-simulation and co-design techniques.

The Sundance innovative [SMT6096](#) HW/SW co-design tool is based on The MathWork Simulink<sup>®</sup> and MATLAB<sup>®</sup> simulation environments. It allows easy conversion of a Simulink diagram into a hybrid DSPS for the Sundance [SMT8096 SDR kit](#). Co-simulating all subsystems (HW+SW+Analogue), together with the ease offered by the Simulink<sup>®</sup> environment, is its other main feature.

The user can describe his/her algorithm by means of a set of interconnected blocks, which are functionally identical to those from the Simulink library (math and logical operators, non-linear and trigonometric functions, vector and matrix operations, modulators, etc.). The Sundance-provided blocks have a Data Flow calculation paradigm, just like Simulink, and they are capable of accurate simulation of the effects of their SW, digital and analogue real HW equivalents; at the same time, the entire system is kept independent of the chosen HW implementation.

As soon as the DSPS has been co-designed, co-simulated and tuned (by using the powerful MATLAB capabilities), it can be automatically converted into C code (for all SW blocks) and VHDL code (for all HW blocks), HW/SW interfaces included (which are converted into both C and VHDL sub-blocks). The generated code is then compiled with Texas Instrument Code Composer Studio's<sup>™</sup> optimizing Compiler and Xilinx ISE Foundation's<sup>™</sup> VHDL synthesizer respectively. As a last step, the resulting bitstreams and DSP applications are downloaded into the FPGA configuration memory and the DSP program memory.



### *Zoom in on HW/SW Co-Simulation and Co-Design*

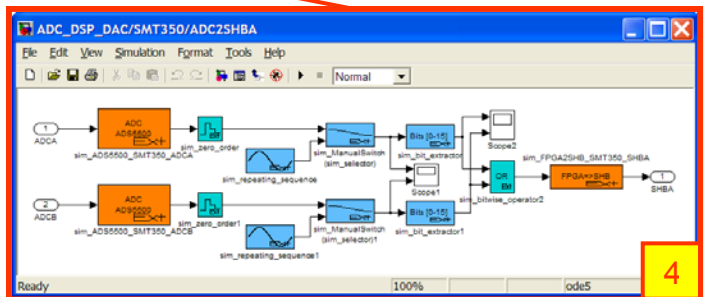
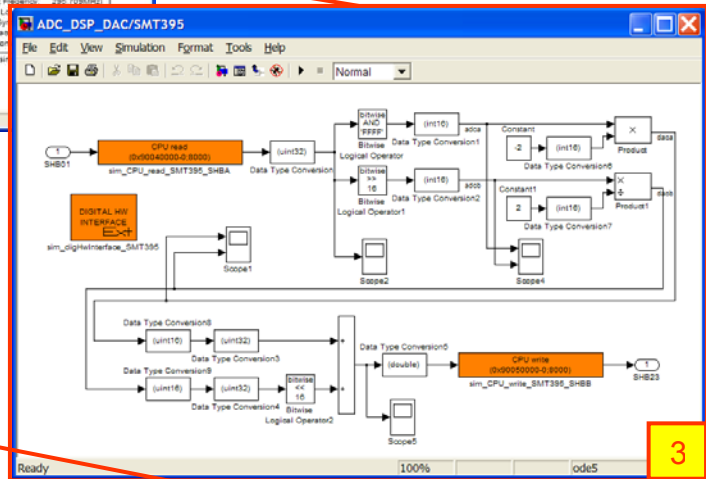
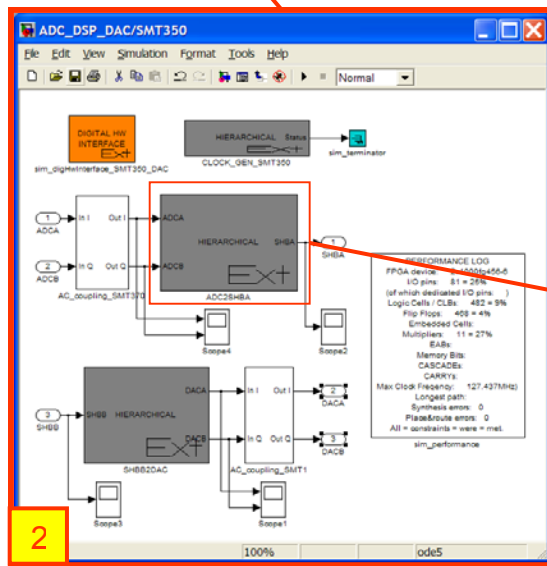
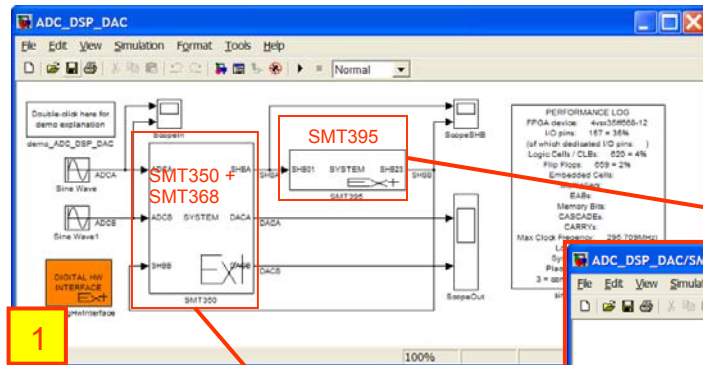
HW/SW co-design is an innovative and very high-level approach to the design of high-performance and low-cost systems. Any algorithm is described in terms of general-purpose functions (mathematical, logical, signal & video processing, etc.), and is kept implementation-independent.

The user can select in a simple check box if he prefers to have C or VHDL code automatically generated as output for each SW or HW module, thus enabling easy experimentation with optimal task allocation between the DSP and the FPGA.

Afterwards, the algorithm is validated inside the application environment where it will run (for example, camera acquisition or radio signal processing).

An implementation (either analogue or digital or software) is then associated with each subsystem and the whole system is then simulated to assess the overall effects of the implementation details (resolution, SW data types, etc.). MATLAB tools can be used to further optimize and tune the system.

As soon as the algorithm has been validated and the implementation parameters have been optimized, C code (for the SW blocks) and VHDL code (for the HW blocks) will be automatically generated, and will be subsequently compiled by the appropriate C and VHDL compiler. Finally, they will be downloaded onto the [SMT8096](#) **Software Defined Radio Development Kit** boards, where the system will be tested for real.



SMT8096 model. **Picture 1** shows the top level containing an SMT350 “block” and an SMT395; interconnections are made with SHB channels; the input stimuli in this case are two analogue sine waves; a few scopes are employed to monitor internal and output signals.

**Picture 2** depicts the schematics that will be compiled and placed inside the SMT350 “block” (which actually consists of the SMT368 FPGA module and the SMT350 ADC/DAC module). The SMT350 “block” comprises many sub-blocks: a dual-ADC (shown in **Picture 4**), a dual-DAC (details not shown) and relative input and output AC coupling.

**Picture 3** contains the schematics for the SMT395 (DSP module), which will process the signals coming from the ADCs and going to the DACs. Compilation of this model will generate all the relevant C and VHDL code for the SMT368/350 and for the SMT395. Xilinx ISE® and Texas Instruments CCS® will then be used to build the two bitmaps and an executable.

## **Requirements:**

### ***The MathWorks:***

- [MATLAB](#)<sup>®</sup> 7.3 and [Simulink](#)<sup>®</sup> 6.5
- [Real Time Workshop](#) (version found in Matlab<sup>®</sup> 7.3)

### ***Texas Instruments:***

- [Code Composer Studio](#)<sup>®</sup> 3.1

### ***Sundance:***

- [SMT8096](#)

### ***Xilinx:***

- [Xilinx](#) ISE<sup>®</sup> 8.2 Foundation