An automated design flow for fast prototyping of Simulink models onto MPSoC

{frobino, johnnyob}@kth.se

KTH Royal Institute of Technology, Stockholm, Sweden

KTH Information and Communication Technology

Problem

Simulink is an industrial de-facto standard for building executable models of embedded systems at system-level.

Once the Simulink system-level model has been validated through simulation, the problem arises as **how to automate the generation of a working prototype from the Simulink system-level model**.

In addition, the techniques enabling fast-prototyping of Simulink system-level models should target implementations **onto state of the art multi-processor systems on chip (MPSoCs)**.

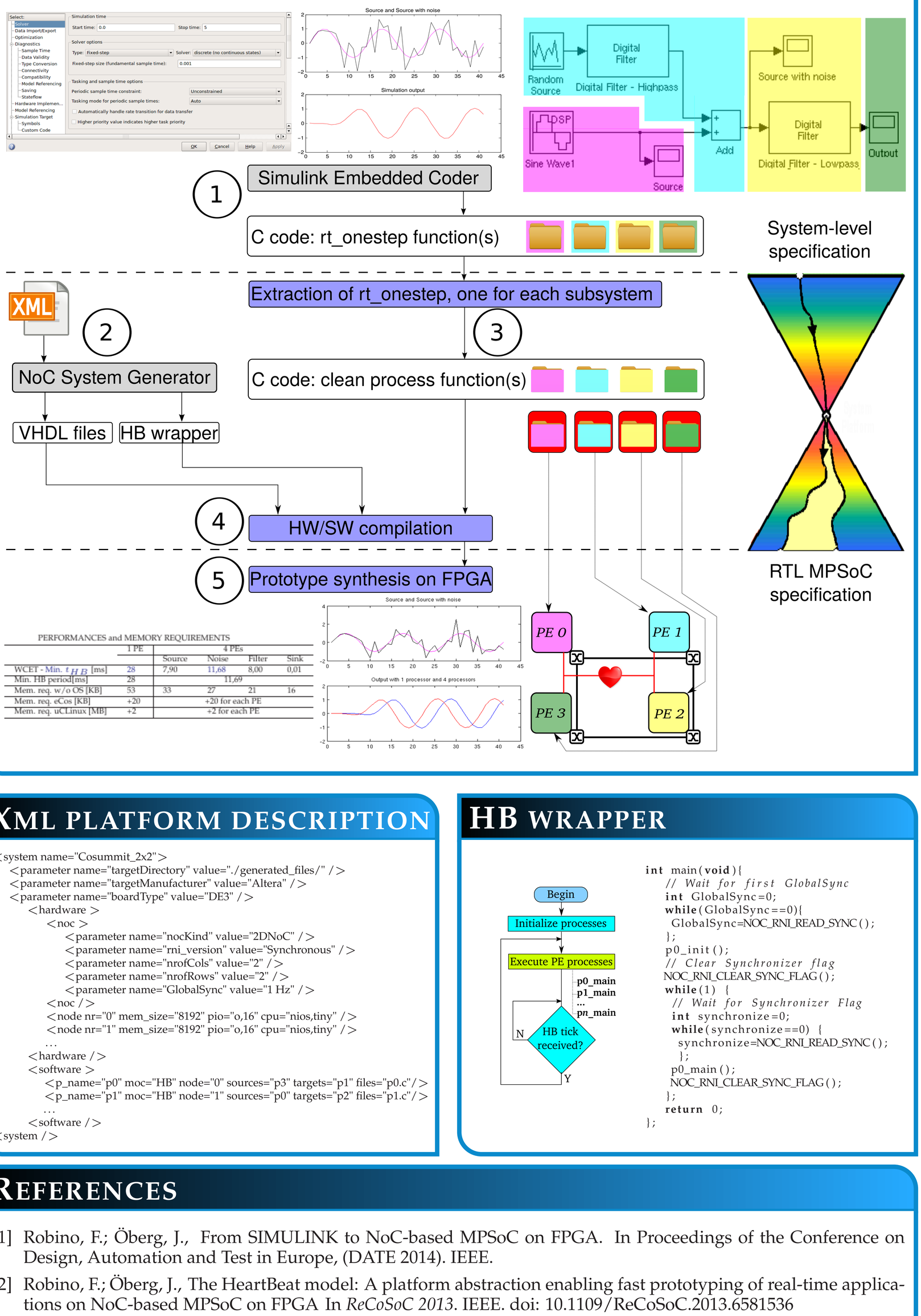
Contributions

1. to **automate the synthesis** of a Simulink model **onto a network-on-chip (NoC) based MPSoC** implemented on FPGA;
2. to constrain the **Simulink** model and its **MPSoC implementation** to share a **common semantics domain**;
3. to propose a **generic method** based on the established **theory of computation models** (MoC);

Tool features

- Prototype heterogeneous NoC-based systems for Xilinx and Altera FPGAs
  - Nios2, Leon3, uBlaze, custom HW
- Design and prototyping time heavily reduced
- Targets low cost FPGAs (low memory and low logic elements availability)
- Easy integration with commercial tools for FPGA development (Altera QSYS and Xilinx XPS)

Platform-based Design Flow



PERFORMANCES and MEMORY REQUIREMENTS

	1 PE	4 PEs			
		Source	Noise	Filter	Sink
WCET - Min. t_{HB} [ms]	28	7,90	11,68	8,00	0,01
Min. HB period[ms]	28	11,69			
Mem. req. w/o OS [KB]	53	33	27	21	16
Mem. req. eCos [KB]	+20	+20 for each PE			
Mem. req. uCLinux [MB]	+2	+2 for each PE			

Design Flow steps

Step 1

System-level model with Simulink;
Simulation of the system;
Embedded Coder: C code generation;

Step 2

XML description of the target platform;
NoC System Generator:

- generation of platform HDL;
- generation of process wrappers;

Step 3

Extraction of `rt_onestep` function from the Embedded Coder generated C files;
Embed the "clean" `rt_onestep` function in the process wrapper (HB wrapper), scheduling its execution on the HB ticks;

Step 4

Compilation of the HDL for FPGA;
Compilation of the C code for each PE;

Step 5

Configure the FPGA;
Download and run the compiled SW for each PE;
Collect the prototype results;

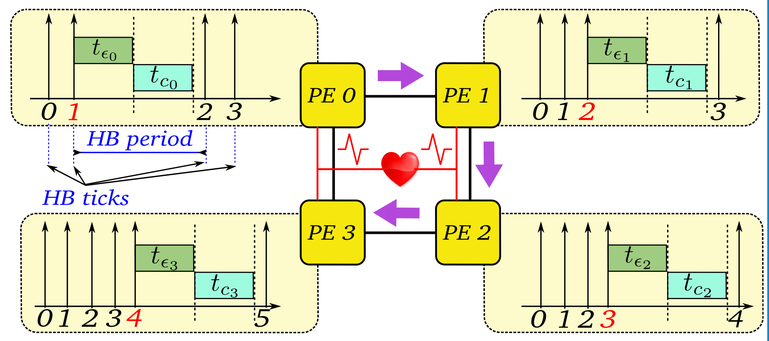
XML platform description

```
<system name="Cosummit_2x2">
    <parameter name="targetDirectory" value="./generated_files/" />
    <parameter name="targetManufacturer" value="Altera" />
    <parameter name="boardType" value="DE3" />
        <hardware >
            <noc >
                <parameter name="nocKind" value="2DNoC" />
                <parameter name="rni_version" value="Synchronous" />
                <parameter name="nrofCols" value="2" />
                <parameter name="nrofRows" value="2" />
                <parameter name="GlobalSync" value="1 Hz" />
            <noc />
            <node nr="0" mem_size="8192" pio="o,16" cpu="nios,tiny" />
            <node nr="1" mem_size="8192" pio="o,16" cpu="nios,tiny" />
            ...
        <hardware />
        <software >
            <p_name="p0" moc="HB" node="0" sources="p3" targets="p1" files="p0.c"/>
            <p_name="p1" moc="HB" node="1" sources="p0" targets="p2" files="p1.c"/>
            ...
        <software />
<system />
```

HB wrapper


```
int main(void){
    // Wait for first GlobalSync
    int GlobalSync=0;
    while(GlobalSync==0){
     GlobalSync=NOC_RNI_READ_SYNC();
    };
    p0_init();
    // Clear Synchronizer flag
    NOC_RNI_CLEAR_SYNC_FLAG();
    while(1) {
     // Wait for Synchronizer Flag
     int synchronize=0;
     while(synchronize==0) {
      synchronize=NOC_RNI_READ_SYNC();
      };
     p0_main();
     NOC_RNI_CLEAR_SYNC_FLAG();
    };
    return 0;
};
```

HB compliant MPSoC



References

[1] Robino, F.; Öberg, J., From SIMULINK to NoC-based MPSoC on FPGA. In Proceedings of the Conference on Design, Automation and Test in Europe, (DATE 2014). IEEE.

[2] Robino, F.; Öberg, J., The HeartBeat model: A platform abstraction enabling fast prototyping of real-time applications on NoC-based MPSoC on FPGA In *ReCoSoC 2013*. IEEE. doi: 10.1109/ReCoSoC.2013.6581536

[3] `http://forsyde.ict.kth.se/noc_generator`

Future work

- Extend wrappers to other MoCs
- Automatic generation of XML file from system-level model through Design Space Exploration