Design Methodology for Efficient Space Time AdapteR

Cyrille CHAVET¹, Philippe COUSSY², Pascal URARD¹, Eric MARTIN² ¹STMicroelectronics, Crolles, FRANCE. {prénom.nom@st.com} ²LESTER, Université de Bretagne Sud, CNRS FRE 2734. {prénom.nom @univ-ubs.fr}

 $f \in \mathcal{F} \setminus \mathcal{T} \in \mathcal{F} \setminus \mathcal{R}$ Laboratoire d'Electronique des Systèmes TEmps Réel





- * Reference design generated using a widespread
- industrial HLS tool.
- * Three different STARs.
- (Two \neq metric setting and one sea of register)

Mode	Ref		FL 7 (Tx 95%)		FL 15 (Tx 90%)		no FL		Thumahaut	
	Saved	Ctrl	Saved	Ctrl	Saved	Ctrl	Saved	Ctrl	Throughput	
300	0	300	56	77	60	240	60	240	434,8	
600	0	600	83	101	130	470	130	470	438	
1200	0	1200	96	117	120	609	168	1032	412,4	

* Total area is 14% smaller using our tool, without merging algorithm.

ReCONFIGURABLE DATAPATH RESULTS

* Multi-Algorithms and Multi-Throughput test cases.

104	CA Area	SDACT ND Area	Our onneoch	Improvement (%)		
wiwi system		SPACI-INIK Area	Our approach	CA	SPACT-MR	
FFT64, FFT32, FFT16, FFT8	781082	524737	350821	55,1	33,1	
FIR64, FIR32, FIR16	54132	26634	18786	65,3	29,5	
FIR19, FIR15, FIR11, FIR7	37701	11103	9249	75,5	16,7	
FFT16, IFFT16	118538	109238	81017	31,7	25,8	
FFT8, IFFT8	36033	31545	25561	29,1	19	
LMS16, FIR16	51396	38884	36016	29,9	7,4	
DCT, PRODMAT	774351	530143	324809	58,1	38,7	
DCT. FIR	370115	345813	335817	93	2.9	

* Gain Vs CA : 44%.

* Gain Vs SPACT-MR : 22%.

- Binding algorithm.
- Optimization step.
- * Design space exploration through user plotted metrics.
- * Generation of multi-mode datapath architecture in HLS design methodologies.
- * Incoming works will focus on :
 - Pipelined architecture.
 - Semi-automated design space exploration through ILP methodologies.
- * STAR is used in GAUT (HLS tool from LESTER lab.).

A Design Space Exploration for Space-Time AdapteRs Abstract CHAVET Cyrille¹, COUSSY Philippe², URARD Pascal¹, MARTIN Eric² ¹STMicroelectronics, Crolles, FRANCE. {firstname.lastname@st.com} ²LESTER Lab, UBS University, CNRS FRE 2734. {firstname.lastname@univ-ubs.fr}

Digital Signal Processing (DSP) applications are know widely used from automotive to wireless communications. The ever growing design complexity, and the performance requirements, and constraints, on design costs and power consumption still require significant parts of a design to be implemented using a set of dedicated hardware accelerators. A classical complex DSP application architecture uses several complex processing elements, a lot of memories, data mixing modules (interleaver for TurboCodes, Spatial redundancy blocks for OFDM/MIMO systems...), and is based on a point to point communication network for inter processing element communications. Such a system may also require to include several applications in a single architecture ((re)configurable systems). Today, their cost in terms of memory elements is very expensive; that's why the designers try to reduce the size of the embedded buffers in order to reduce the overall design area and consumption, and to enhance design performances. In our work, we focus on the optimisation of component communication interfaces. This problem can be seen as the synthesis (1) of interfaces for IP cores integration, (2) of data mixing blocks (such as interleavers) with multi-modes architectures, and (3) of (re)configurable datapath synthesis in high level synthesis flows.

We propose a design methodology to automatically generate and optimize a communication adapter named Space-Time AdapteR (STAR). Our design flow inputs (1) a timing diagram (constraint file) or (2) a C description of I/O data scheduling (an interleaving formula), and user requirements (throughput, latency...), or (3) a set of scheduled and bound CDFGs, and formalizes communication constraints through a formal Multi-Modes Resource Constraints Graph (MMRCG). The MMRCG properties enable efficient architecture space exploration to generate a Register Transfert Level (RTL) STAR component.

The STAR architecture is composed of a datapath (using FIFOs, LIFOs and/or registers) and the associated control state machines. Spatial adaptation (a data can be send from any input port to any/several output ports) is performed by an interconnection logic. Timing adaptation (data reordering) is realized by the storage elements. The STAR component uses a LIS interface (Latency Insensitive System) that enables to implement a *gated clock* mechanism. The proposed design flow can generate multi-modes architectures.

The design flow is based on the following tools:

- *StarTor* inputs a C level algorithmic description which specifies the interleaving scheme, and user requirements (latency, throughput, communication interface, I/O parallelism...). It extracts I/O data order by generating a trace from the C functional description. Next, it generates the constraints file. This tool is used to generate the constraints from a C description.

- *StarDFG* inputs a set of CFDGs generated by a High Level Synthesis tool. These CDFGs are supposed to be scheduled and bound. This tool extracts data communication order. Then, it generates the constraints file. This tool is used to generate the constraints from a CDFG.

- *STARGene*, based on a five-step flow, generates the STAR architecture: (1) Muli-Modes Resource Compatibility Graph construction from constraint file (generated by StarTor or StarDFG)), (2) Modes merging step, (3) Storage resource binding on the MMRCG, (4) Architecture optimization and (5) VHDL RTL generation.

- StarBench generates a test bench based on constraints in order to validate the design by comparison of simulation results.

In a first experience [1], our design flow has been used to generate an industrial Ultra Wide Band interleaver example. This is an industrial test case and these experiments have been performed in collaboration with STMicroelectronics. Using our flow, we show that we can save memory resources and decrease the latency in any case, compared to classical approach based on memory. Moreover the number of structure to be controlled is smaller, with our model, than in the reference design from STMicroelectronics. Currently, the total area of the generated design is about 14% smaller than the reference design from STMicroelectronics (generated with a widespread commercial HLS tool).

In a second experiment [2], we use de STAR design flow in a HLS flow in order to generate a reconfigurable (muli-modes) datapath. These experiments have been performed to generate multi-throughputs (FFT 64 to 8, FIR 64 to 8...) and multi-configurations (FFT and IFFT, DCT and FIR...) architectures. These experiments show the efficiency of the combination of (1) our approach and (2) the multi-modes scheduling and binding algorithms developed in the HLS tool GAUT developed at the UBS University / LESTER Lab, for the generation and the optimization of the memorising part and the steering logic of a datapath. We reduce the total area up to 75% compared to a cumulative architecture, and up to 40% compared to the systems generated by a dedicated multi-modes design flow (SPACT_MR).

Bibliography

[1] CHAVET Cyrille, COUSSY Philippe, URARD Pascal and MARTIN Eric, "A Methodology for Efficient Space-Time Adapter Design Space Exploration: A Case Study of an Ultra Wide Band Interleaver", *IEEE International Symposium on Circuits and Systems, ISCAS 2007*

[2] CHAVET Cyrille, ANDRIAMISAINA Caaliph, COUSSY Philippe, JUIN Emmanuel, URARD Pascal, CASSEAU Emmanuel and MARTIN Eric, "A design flow dedicated to multi-mode architecture for DSP applications", *IEEE International Conference on Computer-Aided Design, ICCAD 2007.*