Low-cost, High-speed Parallel FIR Filters for RFSoC Front-Ends Enabled by CλaSH

#1466

Craig Ramsay, Louise H. Crockett & Robert W. Stewart

University of Strathclyde — 2021 Asilomar Conference on Signals, Systems, and Computers

Objectives

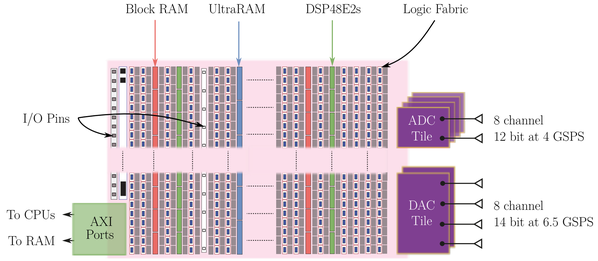
We present a new FIR filter structure for front-end applications with Xilinx's RFSoC, addressing:

- The need for **inexpensive parallel filters** to process at full RFSoC data rates
- Benefit of **zero DSP48E2** usage at the front-end
- Implications of **real-world coefficient** patterns with our proposed designs
- The need for modern, **functional hardware description languages** to facilitate verifiable, parameterised designs with such resource savings

Introduction

Below is a simplified view of the FPGA and RF capabilities of the ZCU111 RFSoC development board. Note in particular:

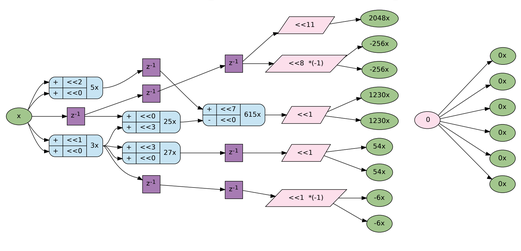
- Relative scarcity of the DSP48E2 units
- The number and throughput of RF channels ($\approx$ 140 GB/s)



Many radio/instrumentation applications demand the full spectrum or tight Digital Down/Up Converter (DDC/DUC) responses, justifying a bypass of the hardened DDC/DUCs. The fabric clock speed is necessarily lower than the RF sampling clock, so parallel filters are required. Designs making full use of the device will also require many ADC/DAC channels, amplifying any resource savings achieved with a single filter structure.

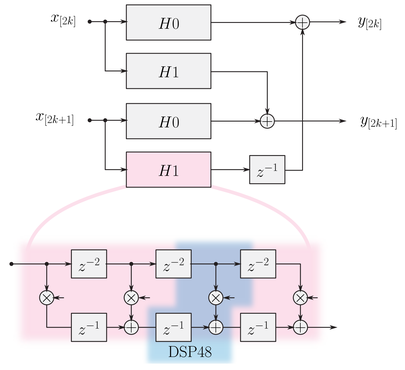
An Example MCM block

Below is a Multiple Constant Multiplication (MCM) graph showing the shifts and adds required to implement an example coefficient set — the 15 tap half-band filter (`fir0`) present in the ZCU111's DDC, using a modified H_{cub} algorithm.



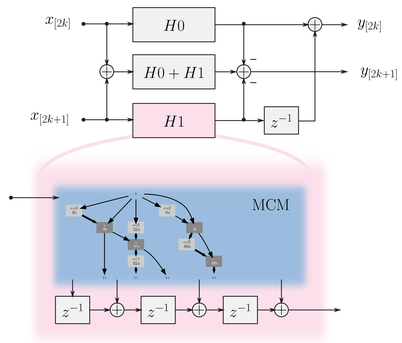
Traditional Implementation

The standard RFSoC filter implementation is a parallel, single-rate filter (exemplified by the LogiCORE FIR Compiler/System Generator "SSR" blocks). It uses a polyphase structure for parallelism and systolic FIR subfilters with DSP48E2 blocks.

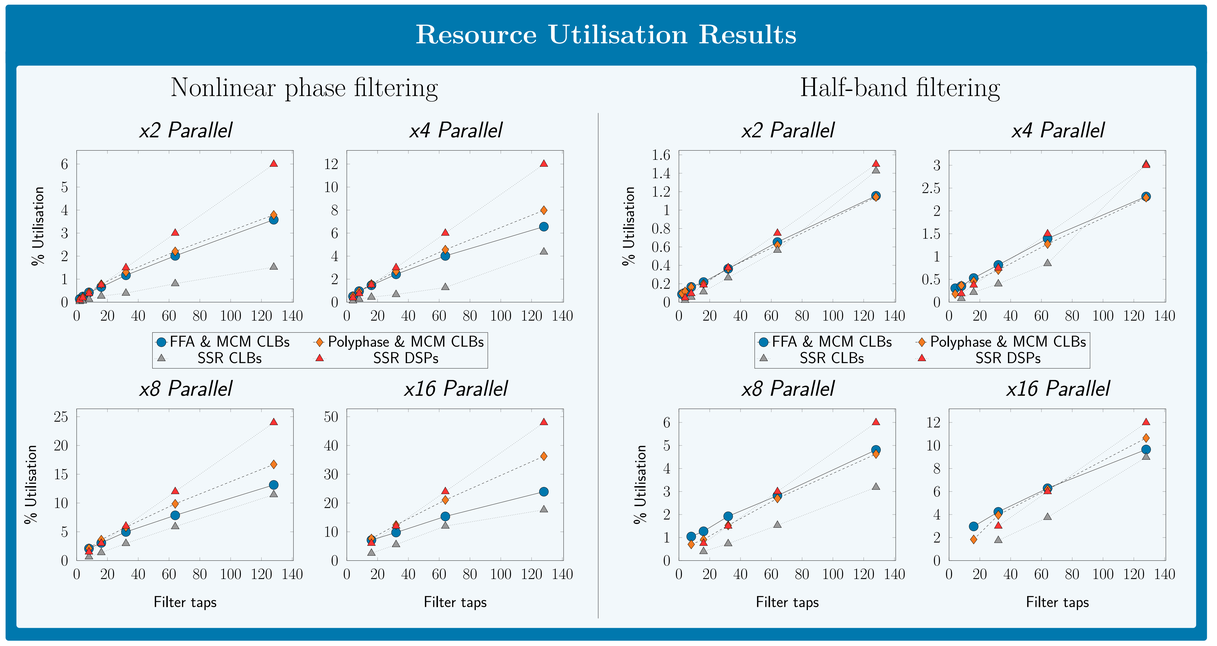


Proposed Implementation

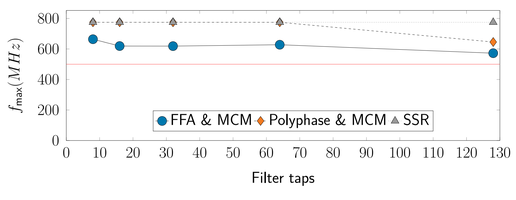
We propose a Fast FIR Algorithm (FFA) structure for parallelism, and MCM blocks (using a variant of H_{cub}) for multiplierless transpose subfilters. This can greatly reduce required multiplications, exploits coefficient symmetry/duplication, and avoids use of *any* DSP48E2 blocks.



Resource Utilisation Results



Timing Results for x8 Parallelism



Multiplier Counts Under Symmetry

Although FFA generally reduces the number of multiplications, the preadder and $H_0 + H_1$ response can cause less favourable performance under coefficient symmetry. We quantify this effect below for a 2^p-parallel filter with $2^p N$ weights.

Coeffs	Polyphase	FFA
Nonlinear phase	$4^p N$	$3^p N$
Padded Type-I	$2^p \left\lceil \frac{2^p N}{2} \right\rceil$	$N(2 + \sum_{k=1}^{p-1} 3^k + 2\sum_{i=0}^{p-2}\sum_{j=0}^{i} 3^j) + (p-1)\left\lceil \frac{N}{2} \right\rceil$
Type-II	$2^p \left\lfloor \frac{2^p N}{2} \right\rfloor$	$\left\lceil \frac{N}{2} \right\rceil + 2N \sum_{i=0}^{p-1} 3^i$
Padded Type-III	$2^p \left\lceil \frac{2^p N}{2} \right\rceil$	$N(2 + \sum_{k=1}^{p-1} 3^k + 2\sum_{i=0}^{p-2}\sum_{j=0}^{i} 3^j) + (p-1)\left\lfloor \frac{N}{2} \right\rfloor$
Type-IV	$2^p \left\lfloor \frac{2^p N}{2} \right\rfloor$	$\left\lfloor \frac{N}{2} \right\rfloor + 2N \sum_{i=0}^{p-1} 3^i$
Half-band	$2^p \left\lceil \frac{2^p N}{4} \right\rceil$	$1 + 2^{p-1} + N + 4N \sum_{i=0}^{p-2} 3^i$

Conclusion

We have demonstrated that, in the context of RFSoC applications, a combination of FFA parallelism and MCM-based subfilters can generate area-efficient and high-speed filters. These filters quite consistently exchange the traditional architecture's DSP usage percentage for a smaller percentage of the generic fabric resources (CLBs) — often under half of the equivalent DSP usage for nonlinear phase filters. This is ignoring the CLB "overhead" incurred by the traditional architecture as well — there are (somewhat extreme) scenarios where our *full* implementation has a smaller CLB area than the traditional implementation's *overhead* alone.

There are some interesting edge-cases for small filters with low parallelism where our polyphase structure with shared MCM blocks will often outperform an FFA equivalent, due to better exploitation of coefficient symmetry. Both implementations are presented here, and are available under open source licenses.

Acknowledgements

The authors would like to thank Xilinx for supplying EDA tools for this project and acknowledge funding for Craig Ramsay under EPSRC grant EP/N509760/1. All source code underpinning this publication is openly available at `github.com/cramsay/conifer`.

References

[1] Xilinx, PG149 "FIR Compiler v7.2 — LogiCore IP Product Guide", 2021

[2] Xilinx, PG269 "Zynq UltraScale+ RFSoC RF Data Converter v2.4 Gen 1/2/3", 2020

[3] Y. Voronenko et al, "Multiplierless Multiple Constant Multiplication", ACM Transactions on Algorithms, Vol. 3, 2007

[4] D.A. Parker, K.K. Parhi, "Area-efficient parallel FIR digital filter implementations", ASAP '96, 1996

StrathSDR