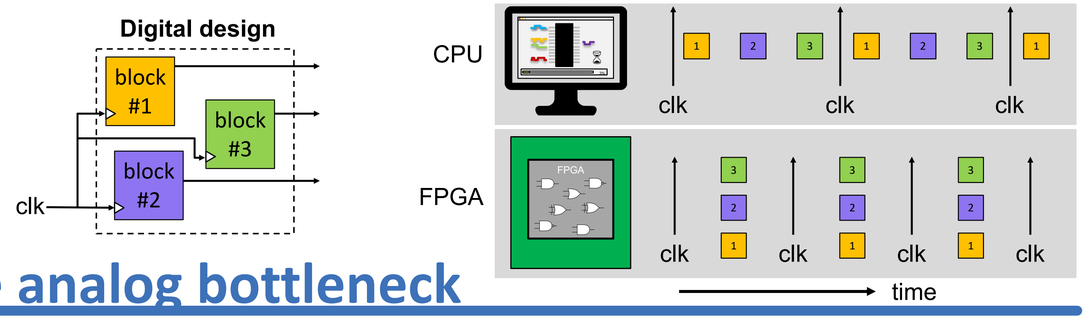
Fast FPGA Emulation of Analog Dynamics in Digitally-Driven Systems

Steven Herbst, Byong Chan Lim, Mark Horowitz

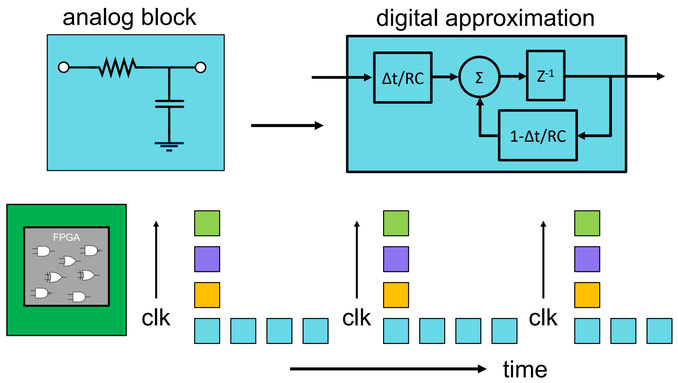
Introduction

- Simulations of entire chip designs are often slow because the CPU has to process blocks sequentially.
- Faster approach: implement the design in the programmable logic of a field-programmable gate array (FPGA). FPGAs can have millions of programmable gates, so this results in a massively parallel simulation.



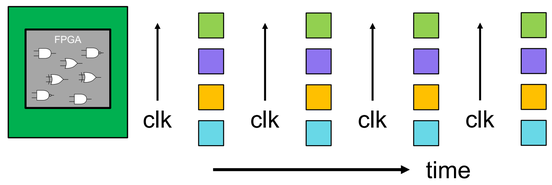
The analog bottleneck

- A digital approximation of analog blocks is needed to map onto the programmable logic of the FPGA for emulation.
- **Conventional approach**: discretize analog dynamics using a fixed timestep.
- **Problem**: if fine time resolution is required, the timesteps have to be really small, so the emulator has to go through many "analog-only" cycles.
- Just one analog block can slow down an entire system emulation.



Our approach

Goal: use only one emulation cycle per digital clock edge. Skip "analog-only" time steps, while preserving fine time resolution.

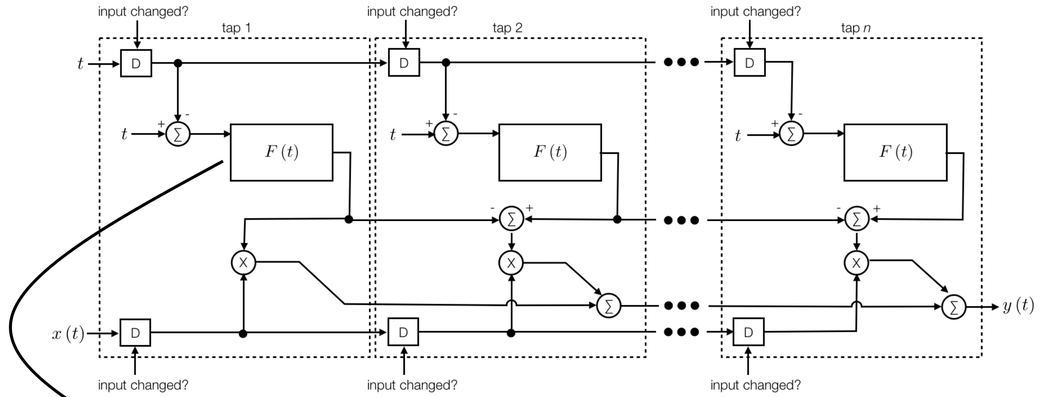


Analog blocks are often "digitally-driven", meaning that their inputs change on digital clock edges and outputs are sampled on digital clock edges. For these types of systems, we can implement analog dynamics precisely using only a precomputed step response, which can be determined prior to emulation.

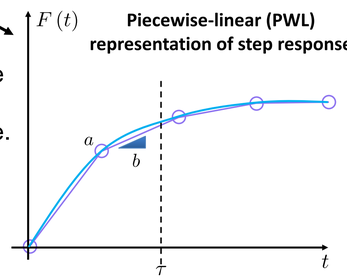
Analog dynamics implemented as a superposition of step responses. The step response is precomputed and stored in block RAM (BRAM) on the FPGA.

FPGA implementation

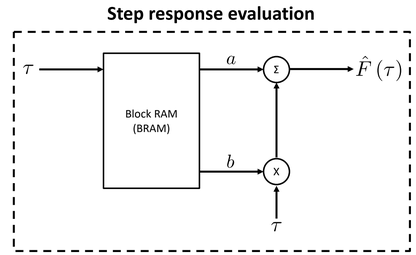
Implementation is similar to a classic finite impulse response (FIR) filter, but coefficients are time-dependent and change every emulation cycle.



Every tap computes one value of the step response.



This PWL approach can be implemented efficiently using FPGA resources.

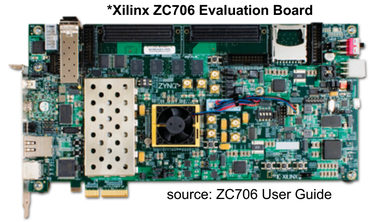


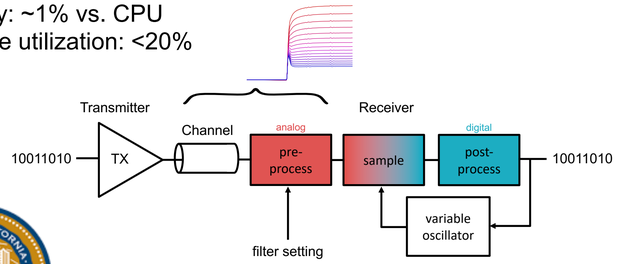
Results

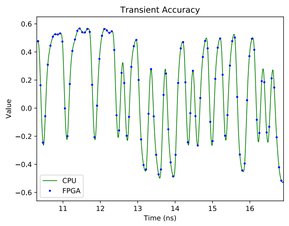
- Applied digitally-driven approach to an 8 Gb/s high-speed link model.
- Implemented emulator with a Xilinx ZC706 board.
- Speedup:
  - >**1000x** vs. CPU sim
  - 25x vs. conventional FPGA approach
- Accuracy: ~1% vs. CPU
- Resource utilization: <20%

Emulation and Simulation Throughput

Type	Platform	Rate (µs/s)
PWL	CPU	0.469
PWL	FPGA*	6.25
Oversampled	FPGA*	50.0
Digitally-Driven	**FPGA***	**1,250**







-0.7/+1.1% agreement with CPU simulation

Work to appear at ICCAD '18 (Session 10B)

ISTC Agile