TECHNISCHE
UNIVERSITÄT
WIEN
Vienna University of Technology

Institut für
Computertechnik
Institute of
Computer Technology

Example-driven Interconnect Synthesis for Heterogeneous Coarse-Grain Reconfigurable Logic

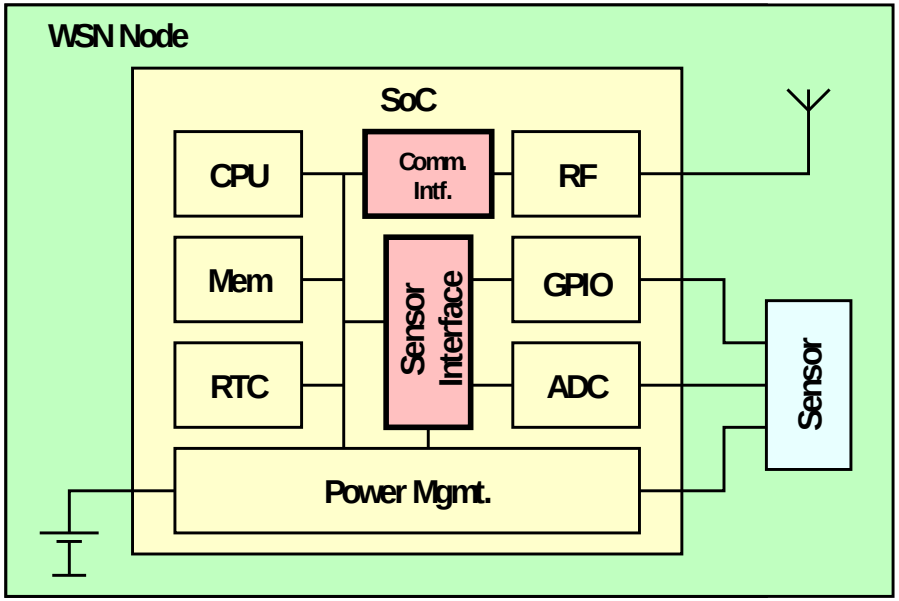
Clifford Wolf, Johann Glaser,
Florian Schupfer, Jan Haase, Christoph Grimm

Overview

- Ultra-Low-Power Electronics
- Reconfigurable Modules
  - Structure
  - TR-FSM
  - Design of Reconfigurable Modules
- Interconnect Generation
  - Interconnect Topology
  - Optimization Algorithm
  - Evaluation Results
- Full Design Flow
- Summary

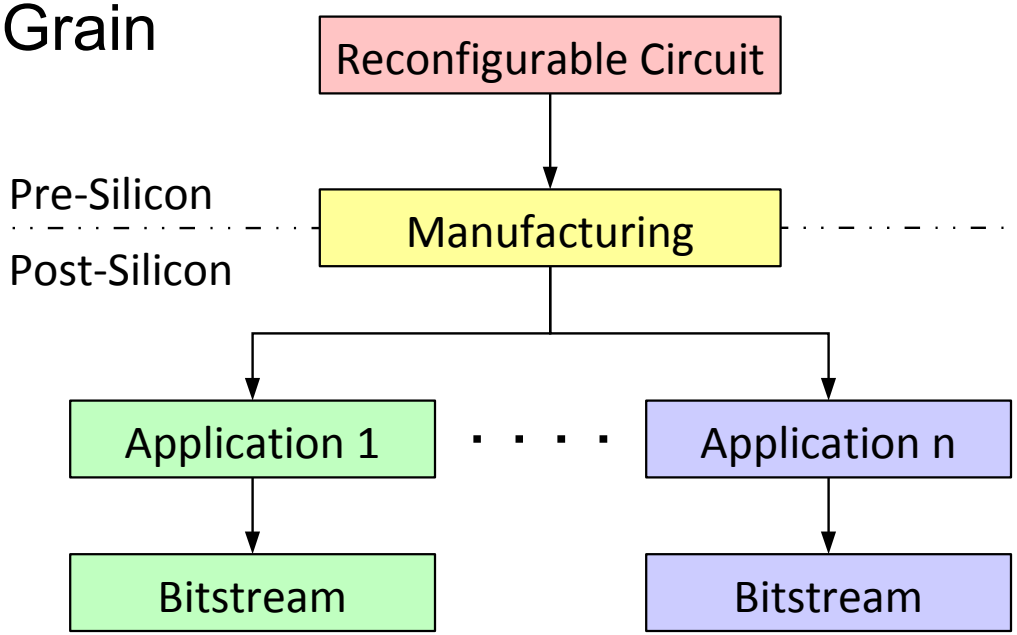
Ultra-Low-Power Electronics

- Application: WSN Node in a SoC
- CPU as Master
- Sleep, Wakeup
- Offload-Engines
- Multiple Modules
- Reconfigurable
  - Various Applications
  - Adapt to Environment



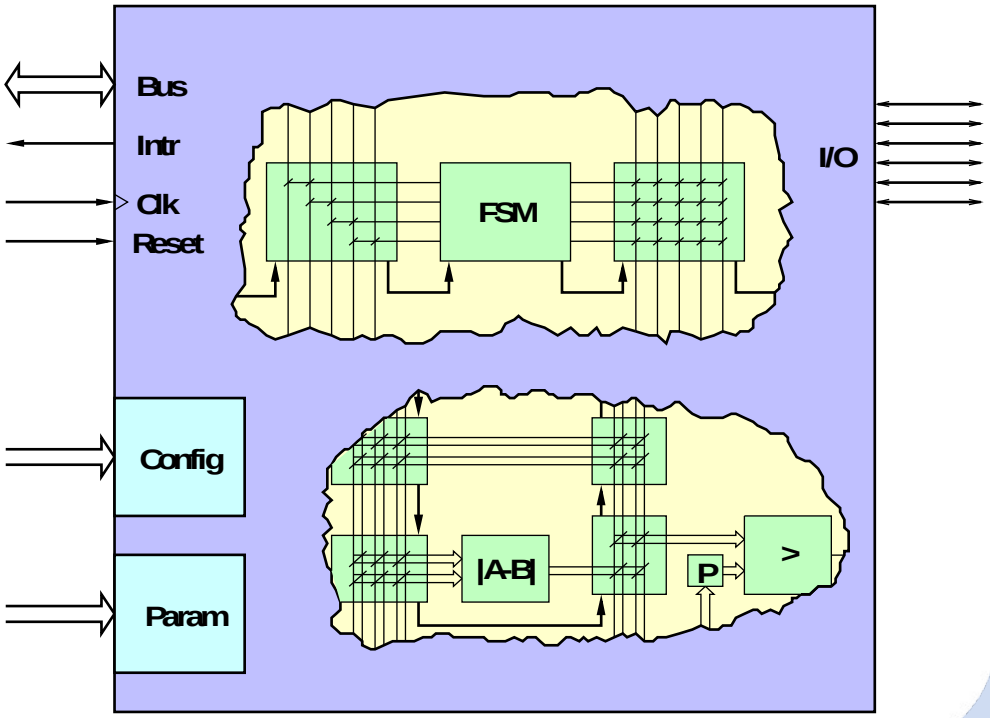
Reconfigurable Logic

- Development
  - Pre-Silicon vs. Post-Silicon
- Fine-Grain vs. Coarse-Grain
- Heterogeneous
- Domain Specific
  - Application Class



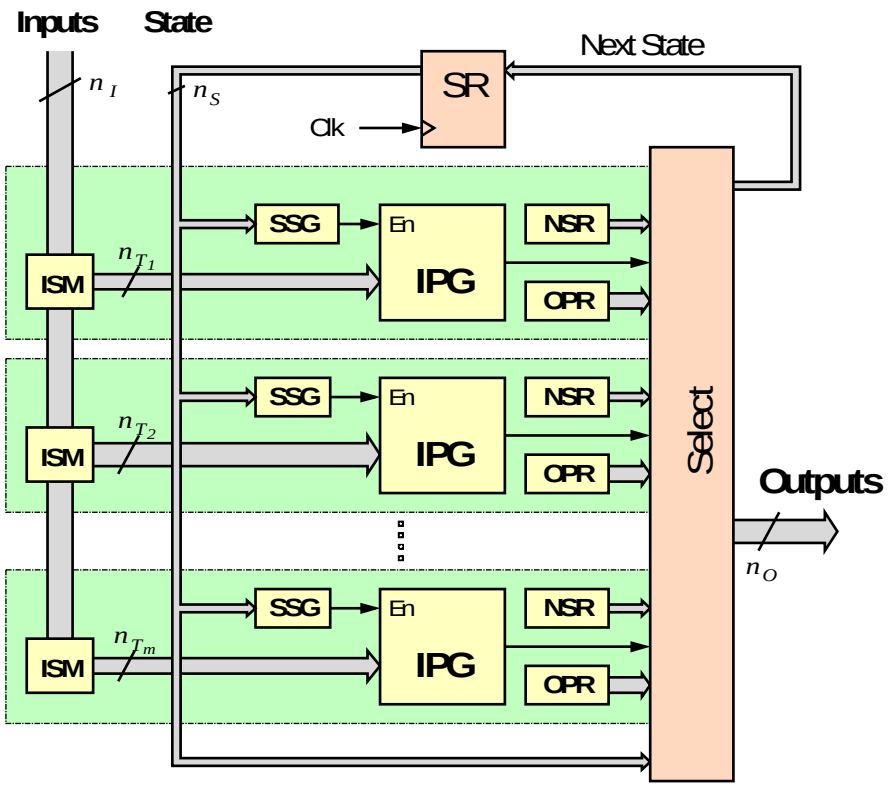
Structure

- Interface
- FSM+D
  - ➔ Control, Data, Arithmetic
- Cell Types
  - FSM
  - Memory
  - Add, Subtract, |A-B|, ...
- Reconfigurable
  - Routing
  - Cells



Transition-based Reconfigurable FSM

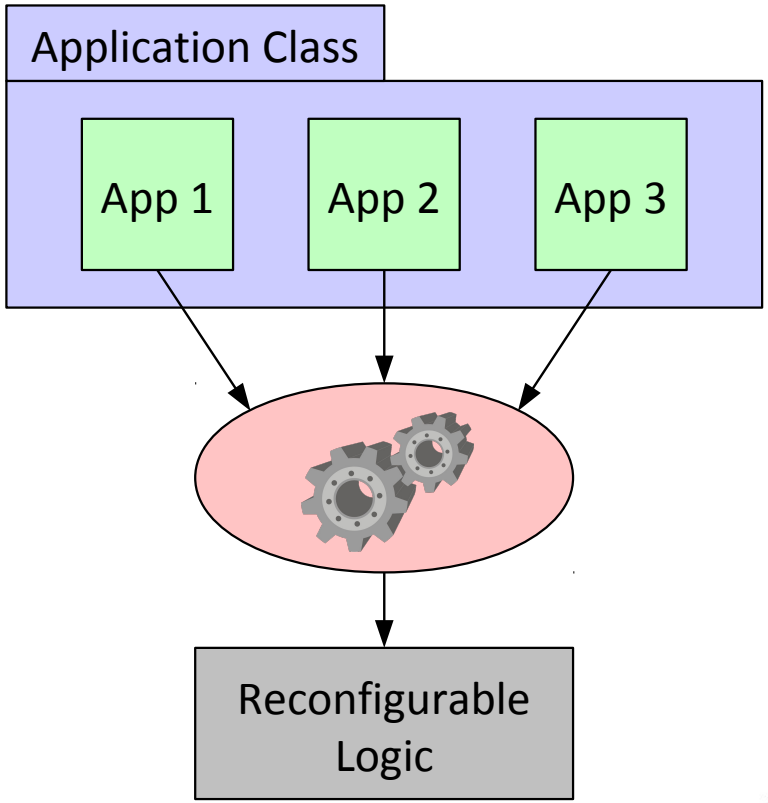
- Reconfigurable Cell
- Focus on Transition
- Transition Row
  - SSG: State Selection Gate
  - ISM: Input Switching Matrix
  - IPG: Input Pattern Gate
  - NSR: Next State Register
- VHDL
- Silicon-proven



Design

- Pre-Silicon Phase
- Define Application Class
- Specify Example Applications
- “Common Denominator”
- Pre-Silicon limits
  Post-Silicon Design Space
- Future Applications
  - Might need more Resources

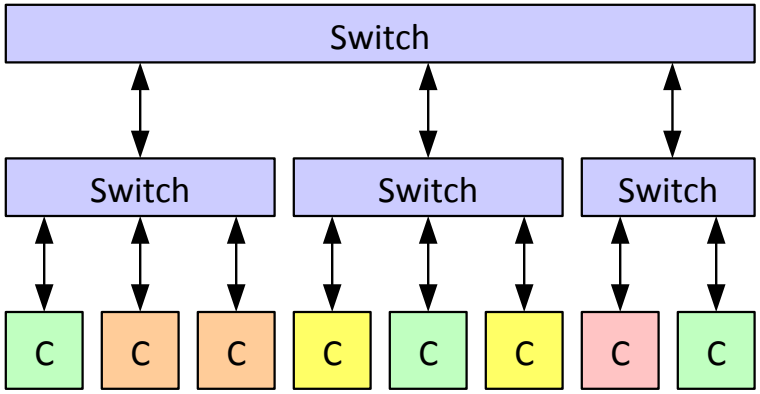
➔ Oversizing
  - Additional Cell Instances
  - Allow more Connections



Connection Topology

- Connection Types
  - e.g. Bit, Byte, Word
- Requirements
  - Random Connections
  - Simple Characterization
  - Allow Optimization
  - Limit Optimization
  - Oversizing
  - Synthesizable

- Tree Topology



- Parallel Trees
  - All Ports in all Trees
  - Alternate Position
  - Routing in any Tree

InterSynth: Overview

- Automatic Interconnect Generation and Optimization

- Pre-Silicon
  - Input
    - Example Netlists
    - Cell Types
    - Connection Types
  - Output
    - Synthesizable RTL Code
    - Configuration Bitstreams
    - Static Timing Analysis Rules
    - LaTeX TikZ Image
    - Internal Representation

- Post-Silicon
  - Input
    - Internal Representation
    - Netlist
  - Output
    - Configuration Bitstream

InterSynth: Algorithm

- Degrees of Freedom
  - Number of Cells per Type
  - Interconnect Resources
  - Mapping of Cells in Netlist (“Nodes”) to Physical Cells
  - Placement of Physical Cells in Interconnect Tree

- Optimization Steps
  - Optimize Cell Placement
  - Node-to-Cell Mapping
  - Repeat Until no Further Improvement

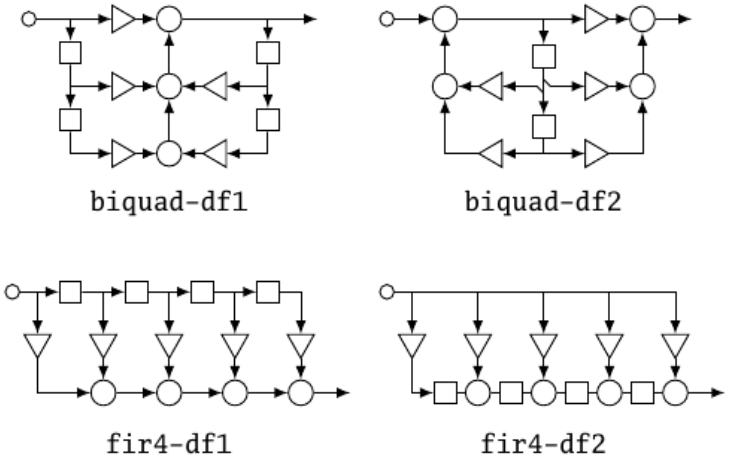
- Optimization Algorithm

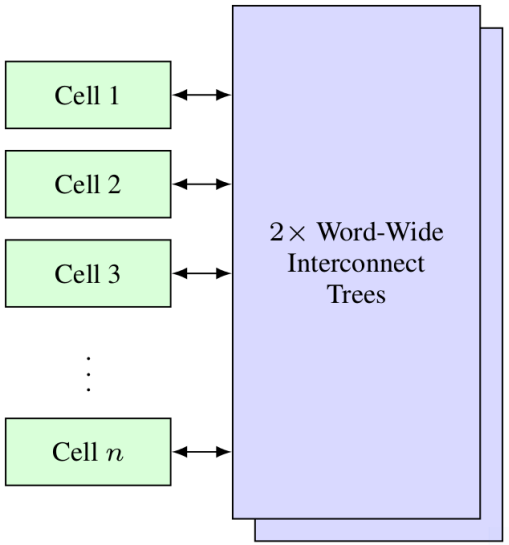
  Modified Kernighan-Lin algorithm
- Optimization Goal

  Minimize Interconnect Resources

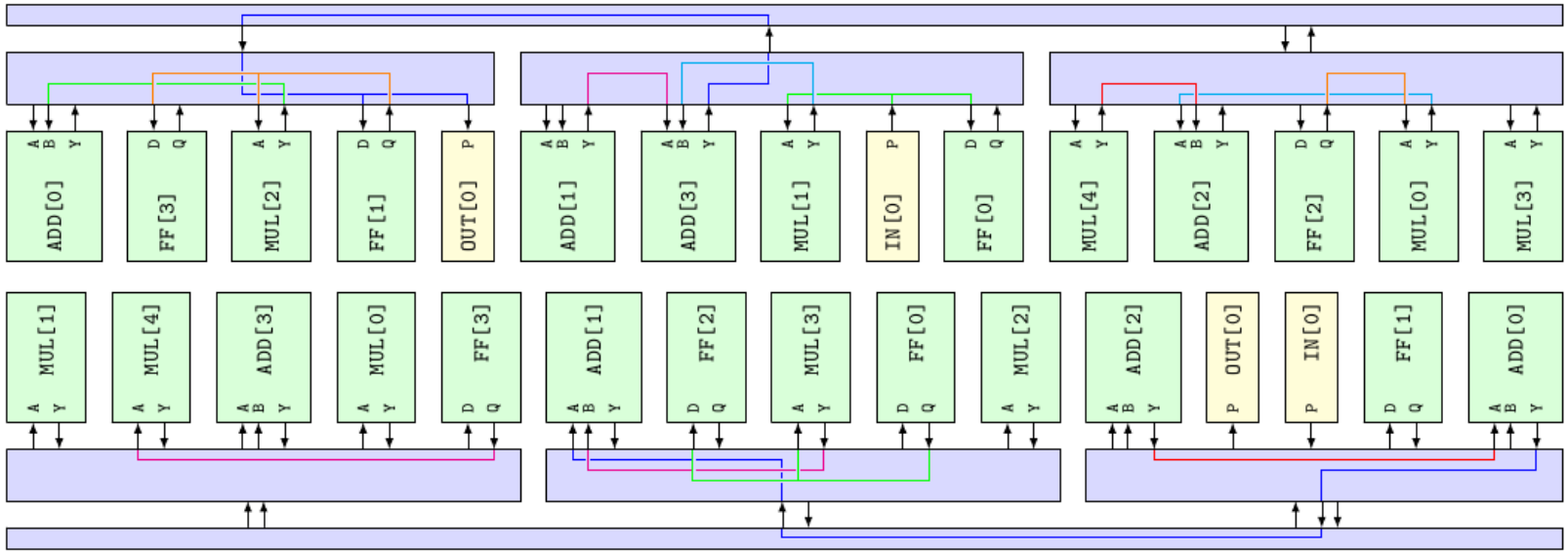
Digital Filters

- Application Class
  - Simple Digital Filters
  - Cell Types
    - Adders
    - Scalers
    - Delays
  - Netlists for Pre-Silicon
    - `biquad-df2`
    - `fir4-df1`
    - `fir4-df2`
  - Extra Netlist for Post-Silicon
    - `biquad-df1`
  - 2 Parallel Interconnect Trees

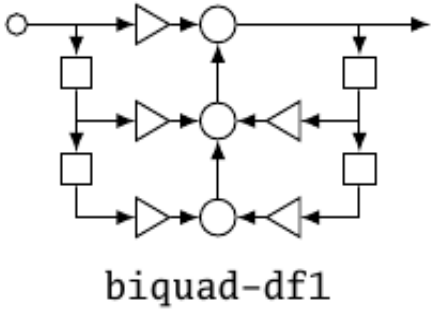




Digital Filters Example

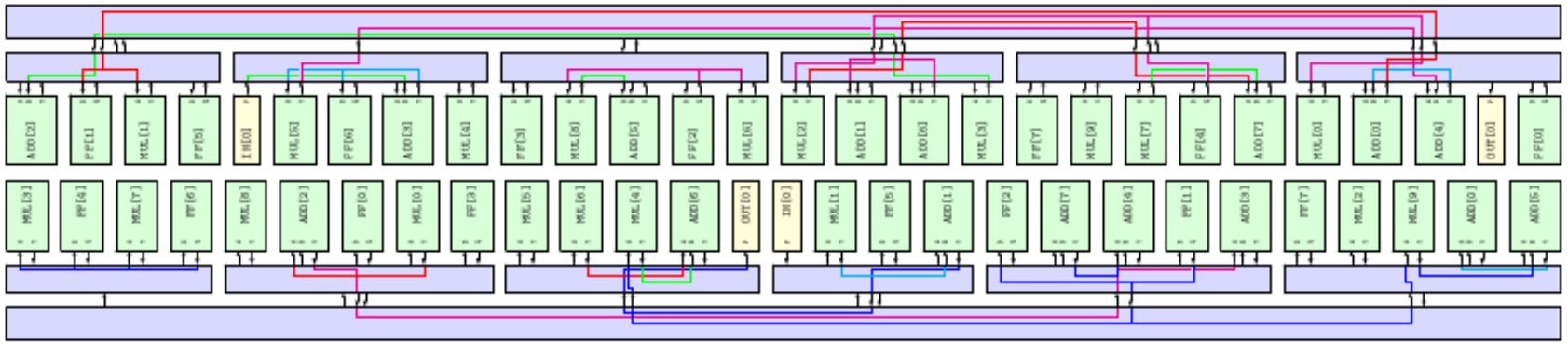


- Post-Silicon
- Netlist: `biquad-df1`



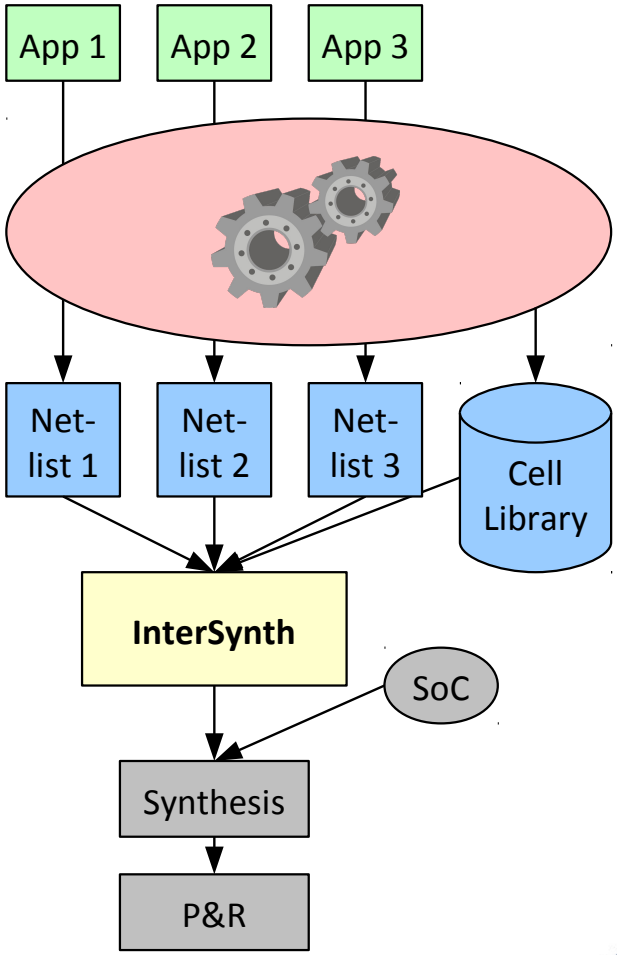
Digital Filters Example

- Higher Order Filters
  - Combine two Stages
  - 16 different Netlists
  - Pre-Silicon: 2-4 Examples enough



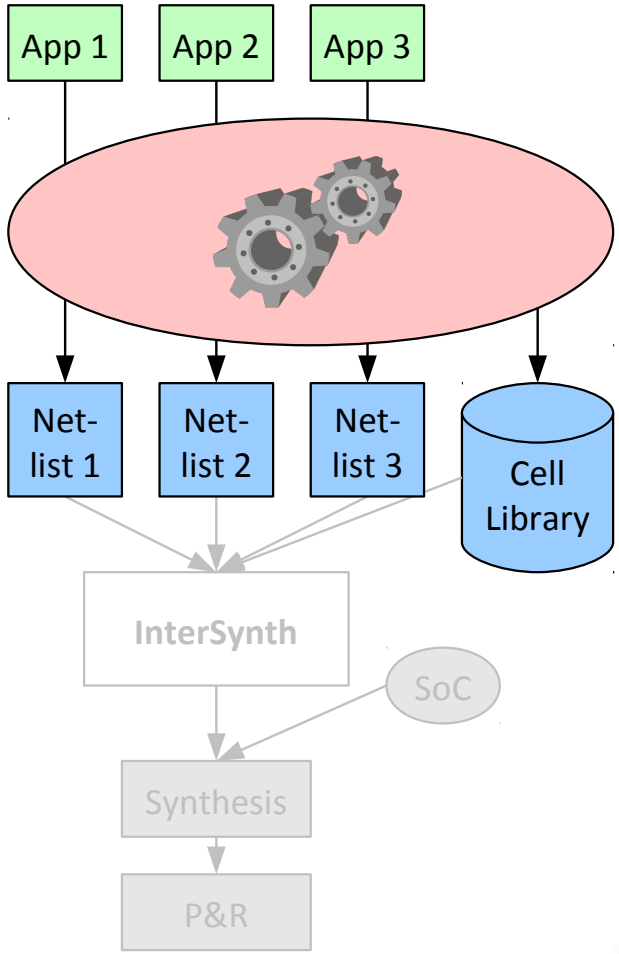
Full Design Flow

- Pre-Silicon Phase
- After InterSynth
  - Synthesizable HDL
    - Instantiate Cells
    - Reconfigurable Routing
  - SoC Integration
  - Synthesis
  - Place and Route
  - Chip
- Input of InterSynth
  - Example Netlists
  - Cell Library



Full Design Flow

- Specification
  - Easy to translate to netlist
  - No new type of description
  - Verifiable
  - ➔ VHDL, Verilog
- Special Synthesis
  - Coarse Grain
  - Identify Cells
  - FSM Extraction
  - Bitstream Generation

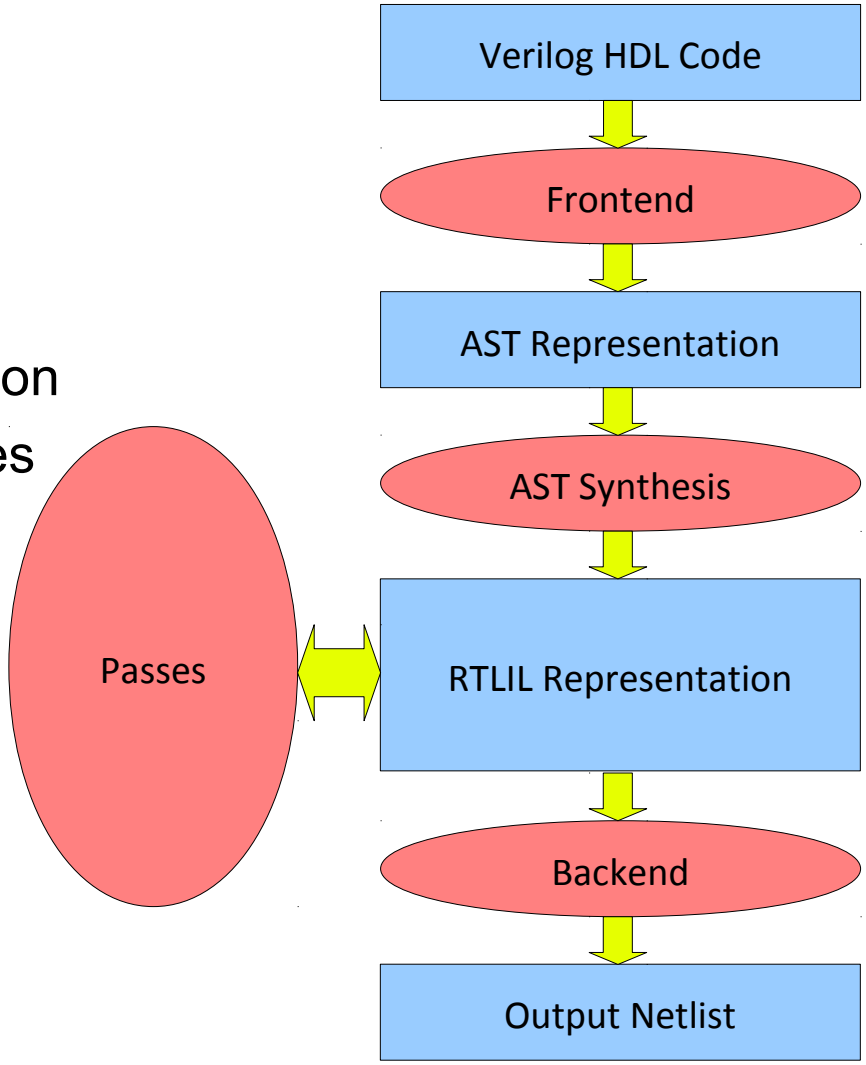


yosys

- Yosys Open SYnthesis Suite
- Extensible RTL Synthesis
- Verilog (VHDL will follow)
- Coarse-grain and Fine-grain
- Focus on easy extensibility
  - Preserve HDL Information
  - Structure
    - Frontend
    - Multiple Passes
    - Backend
- Currently under development

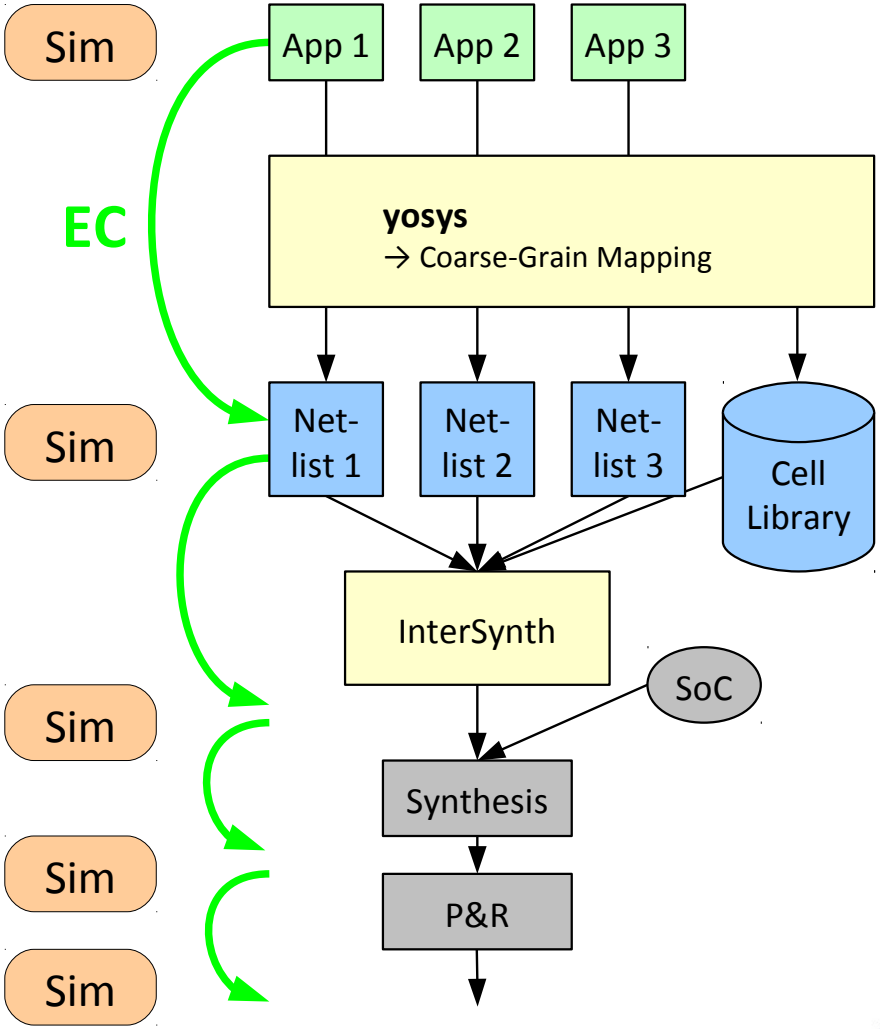
yosys

- Abstract HDL Constructs
- RTLIL
  - RTL Intermediate Language
  - Simple Internal Representation
  - Modified by Synthesis Passes
- Passes
  - Transformation of High-Level Constructs to Low-Level Constructs
  - Optimization
  - Coarse-Grain Mapping



Full Design Flow

- Example Applications
- Coarse Grain Synthesis
- Cell Library
- Verification
  - Simulation
  - Equivalence Checking
  - Wrapper
  - Configuration
- Pre-Silicon
- Post-Silicon



Summary

- Ultra-Low-Power Electronics
- CPU Offload-Engines
- Reconfigurable Modules
- Interconnect Generation
- Coarse-Grain Synthesis
- Full Design Flow

- Future Work
  - yosys VHDL Frontend
  - More on Coarse-Grain Synthesis