# Experiences Using a Novel Python-Based Hardware Modeling Framework for Computer Architecture Test Chips

## This Poster...

Describes a taped-out 2x2 mm 1.3M-transistor test chip in IBM 130nm designed and implemented using PyMTL, a novel Python-based hardware modeling framework

Goal of tapeout was to demonstrate the ability of this framework to enable Agile hardware design flows

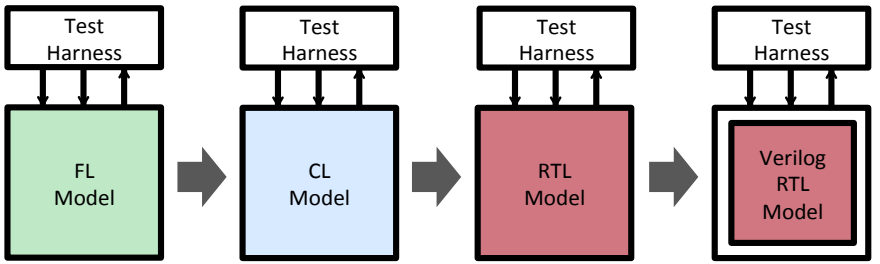
## PyMTL: A Unified Python-Based Framework for FL, CL, and RTL Modeling

**Functional-Level Modeling (FL)**
- Behavior

**Cycle-Level Modeling (CL)**
- Behavior
- Cycle-Approximate
- Analytical Area, Energy, Timing

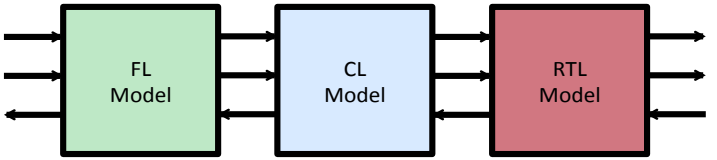
**Register-Transfer-Level Modeling (RTL)**
- Behavior
- Cycle-Accurate Timing
- Gate-Level Area, Energy, Timing

## What Does PyMTL Enable?

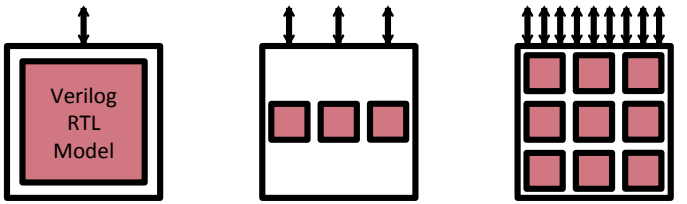



1. Incremental refinement from algorithm to hardware implementation

2. Automated testing and integration of PyMTL-generated Verilog




3. Multi-level co-simulation of FL, CL, and RTL models



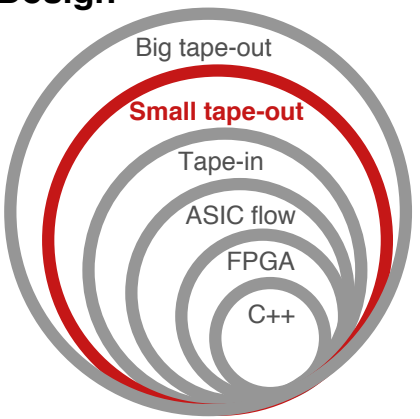

4. Construction of highly parameterized RTL chip generators

# PyMTL for Computer Architecture Test Chips

## Why Build Computer Architecture Test Chips?

**Key Aspect of Agile Hardware Design**
- Rapid design iteration
- "Building the right thing"
- Reduces cost of validation

**Benefits Research**
- Builds research credibility
- Highly reliable power and energy estimates for new architecture techniques




* Adapted from Yunsup Lee IEEE Micro 2016

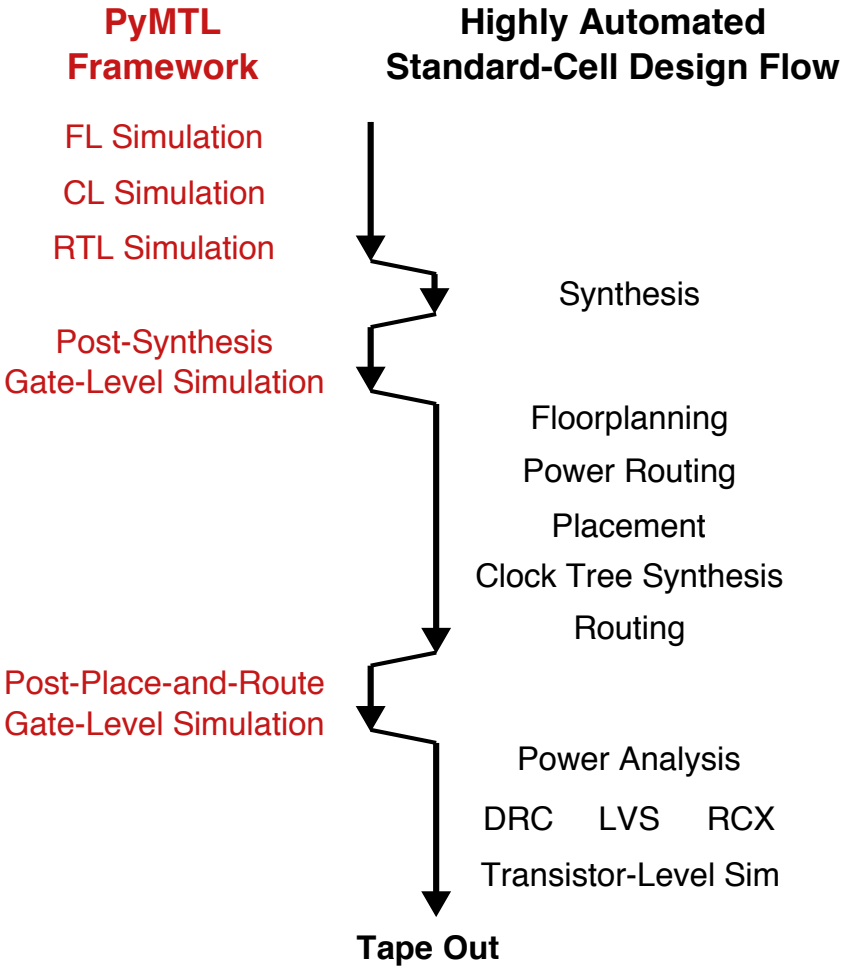
## Design Methodologies: Large Chips vs. Small Chips

**Large-Scale Commercial Chips**
- High-volume and high-yield production
- Overcome design challenges with large teams

**Computer Architecture Test Chips**
- Low-volume and reasonable-yield production
- Overcome design challenges despite small teams and limited resources

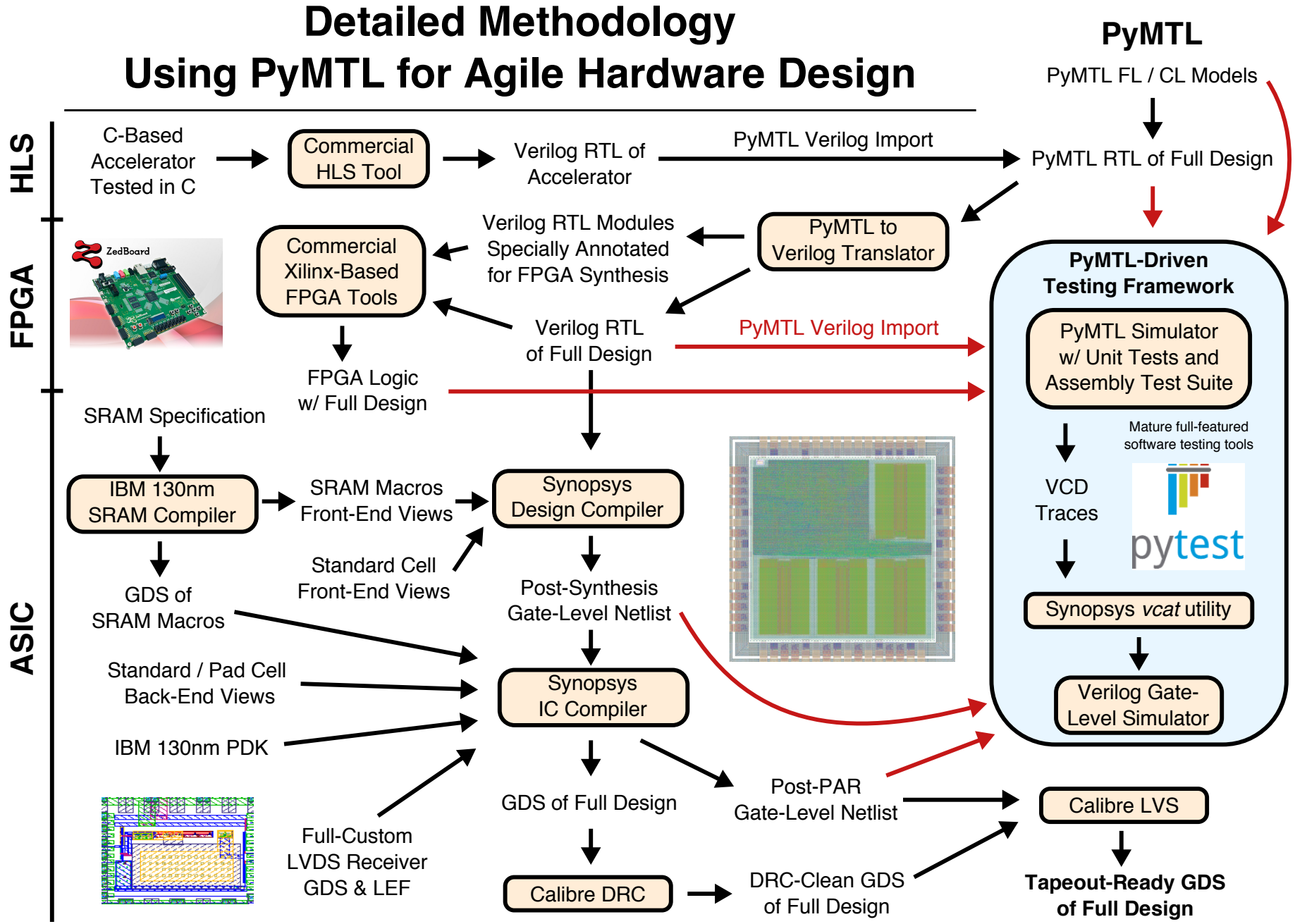
↳ Provide small teams with highly productive development frameworks to shorten time to tapeout

## PyMTL for Agile Hardware Design

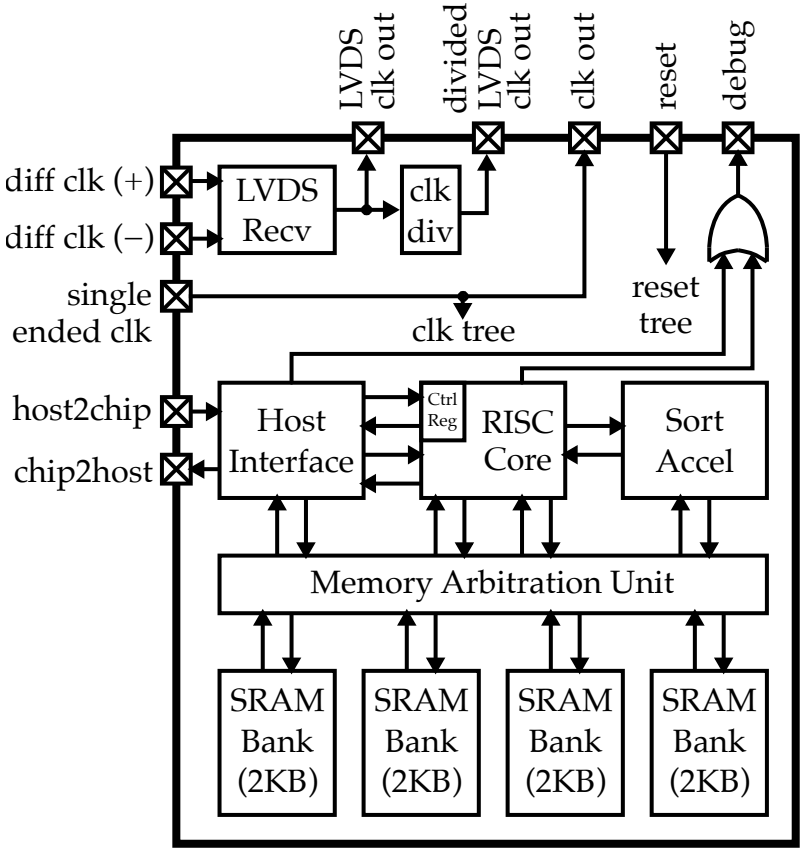



Small teams push RTL to layout with validated gate-level netlist within a day

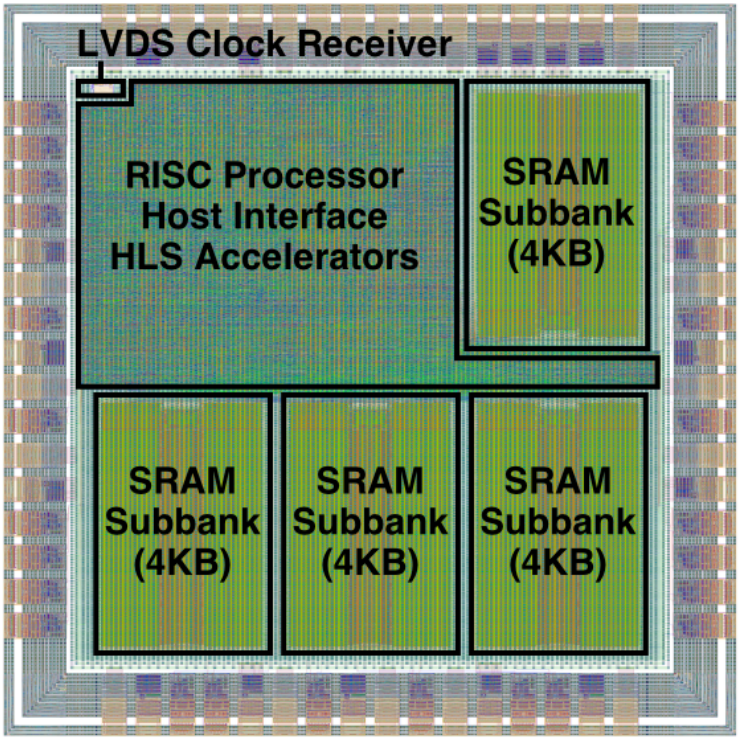
# Detailed Methodology Using PyMTL for Agile Hardware Design



# PyMTL in Practice: BRG Test Chip 1




Taped out in March 2016
Expected return in Fall 2016




**Testing Plans After Fabrication**

The testing platform enables running small test programs on BRGTC1 to compare the performance and energy of pure-software kernels versus the HLS-generated sorting accelerator

**Taped-out Layout for BRGTC1**

2x2mm 1.3M transistors in IBM 130nm
RISC processor, 16KB SRAM
HLS-generated accelerators
Static Timing Analysis Freq. @ 246 MHz