Chiplets
Never defer to Tomorrow what you can defer Forever

August 25, 2022

Bill Dally
Chief Scientist and SVP of Research, NVIDIA Corporation
Adjunct Professor of CS and EE, Stanford

DAC 2013

21st Century Digital Design Tools

William J. Dally
NVIDIA & Stanford University
2701 San Tomas Expressway
Santa Clara, CA 95050
408-486-2000
bdally@nvidia.com

Chris Malachowsky
NVIDIA
2701 San Tomas Expressway
Santa Clara, CA 95050
408-486-2000
chris@nvidia.com

Stephen W. Keckler
NVIDIA & UT-Austin
2701 San Tomas Expressway
Santa Clara, CA 95050
408-486-2000
skeckler@nvidia.com

ABSTRACT

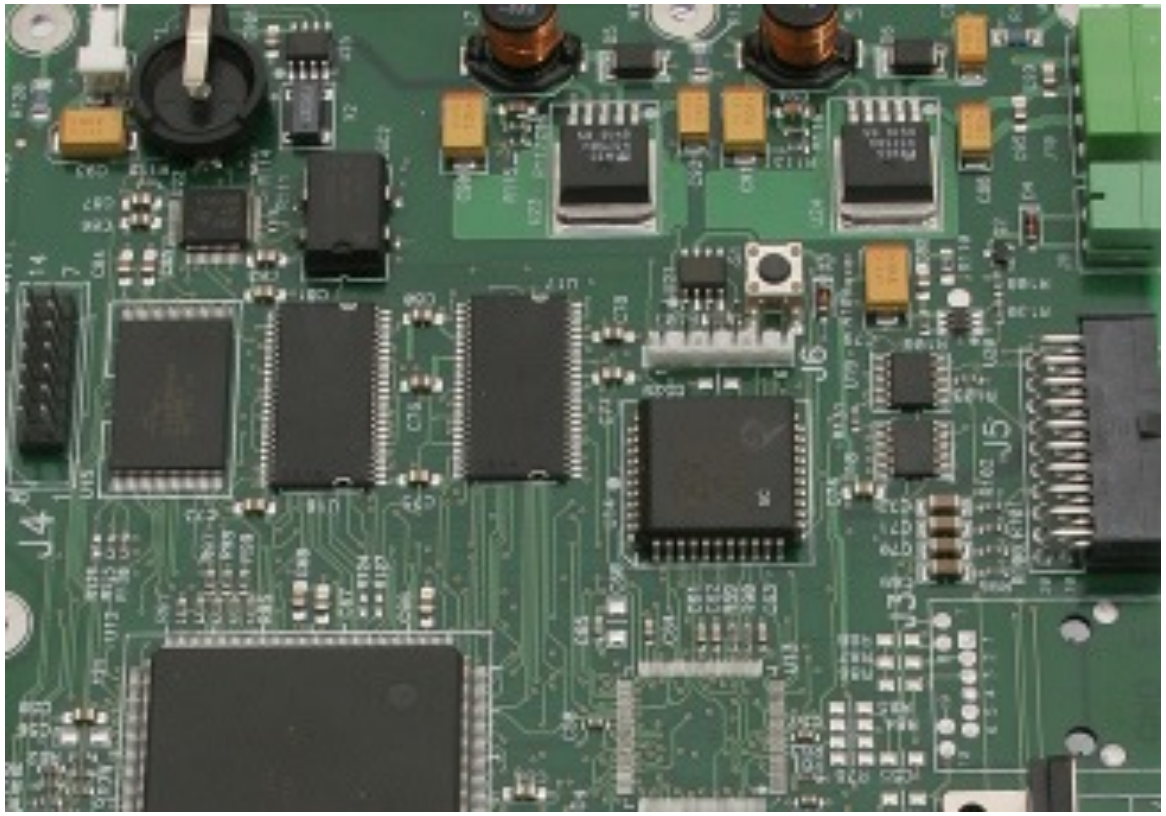
Most chips today are designed with 20th century CAD tools. These tools, and the abstractions they are based on, were originally intended to handle designs of millions of gates or less. They are not up to the task of handling today's billion-gate designs. The result is months of delay and considerable labor from final RTL to tapeout. Surprises in timing closure, global congestion, and power consumption are common. Even taking an existing design to a new process node is a time-consuming and laborious process.

Twenty-first century CAD tools should be based on higher-level abstractions to enable billion-gate chips to go from final RTL to

takes only a few hours and changes can be accommodated with a minimum of rework. These designers are working at the board or system level. In contrast, designers working at the chip level take months to complete a design of similar complexity.

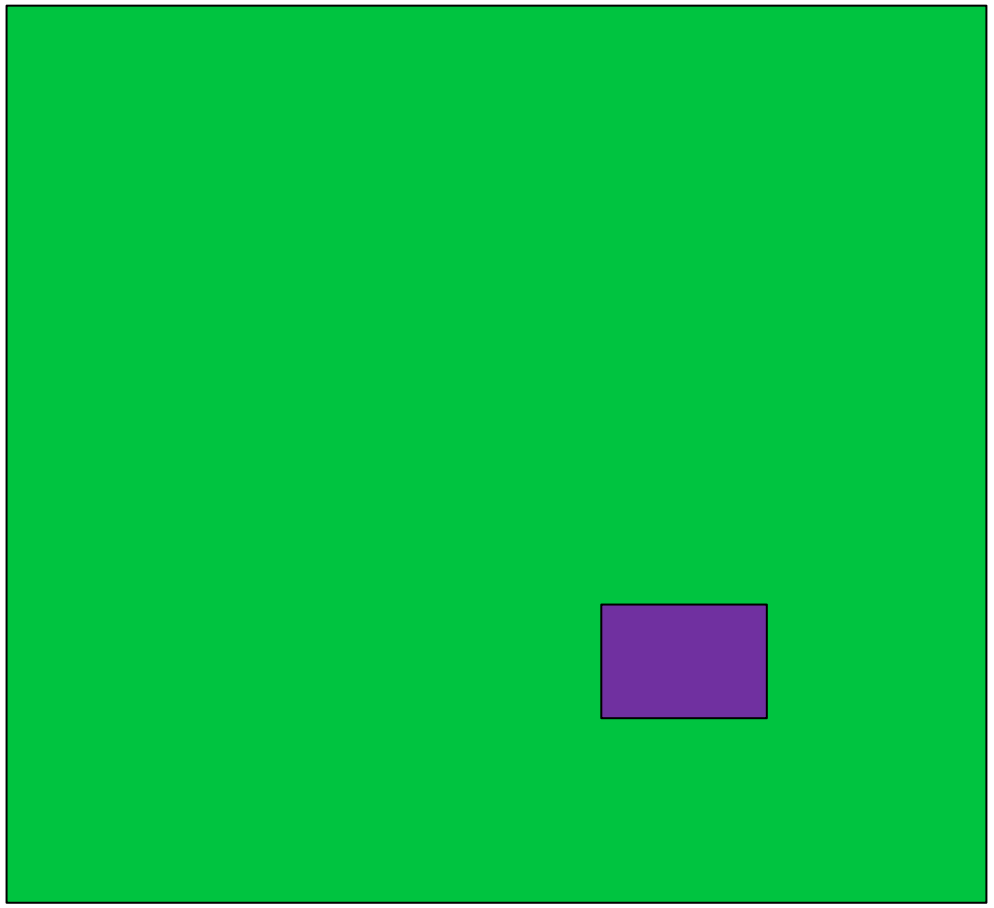
The system designer is able to achieve a high level of productivity because the packaging of the components they are composing enforces modularity. Modularity implies information hiding and a fixed, often standard, interface. The system designer typically sees only the specification of a chip they are using. They cannot see or alter the implementation of the chip. Each chip can only interact with the rest of the system over its package pins, a set of fixed and often standard interfaces. The modularity enforced by

We can design a system in days

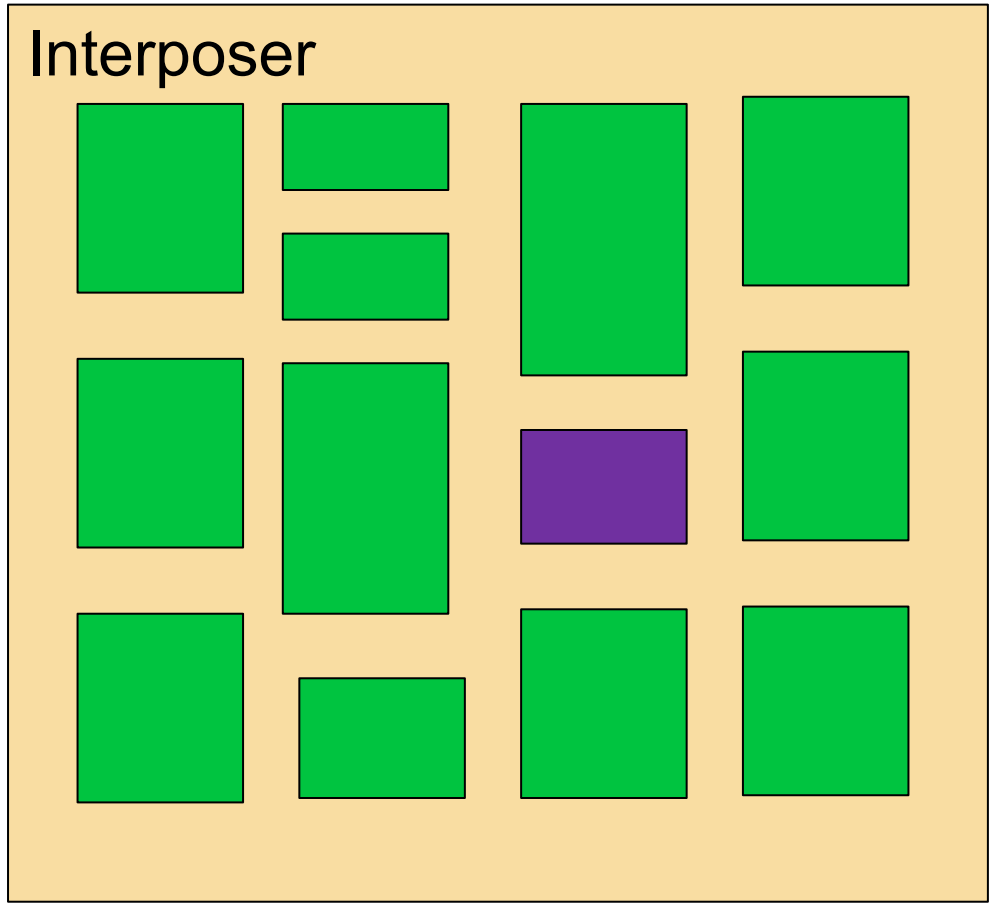


Assembling catalog parts on a printed-circuit board

Chiplets – Catalog Parts with Efficient Interfaces

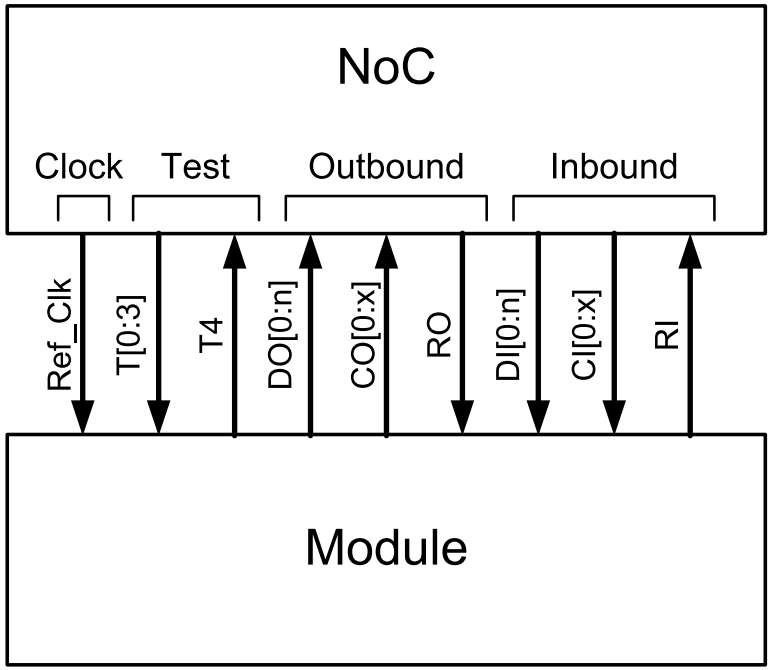


Monolithic SoC



Chiplet Design

Chiplet Issues



- Standard interfaces
  - instruction, memory client, pipeline unit
  - Various bandwidths
  - ABI/AXI is not adequate
- Bootstrapping an ecosystem
  - Need a catalog of standard parts
  - CPUs, GPUs, memory, memory controllers, NoCs, MODEMs, CODECs, …
- Economics
  - Additional cost of interposer/assembly
  - Organic substrates possible