The Secret to Building IP at the Cutting Edge

John G. Maneatis, Ph.D.
President, True Circuits, Inc.

June 21, 2017

The Secret:

it’s maximizing reuse !

The Dilemma

I need it customized for these 3 applications, and I need this metal stack and process variant.

But I need to sell the same thing N times to make money.

Plus, you’ve buying IP to reduce time and risk…. customization adds time and risk!

Answer: Reuse and Configurability

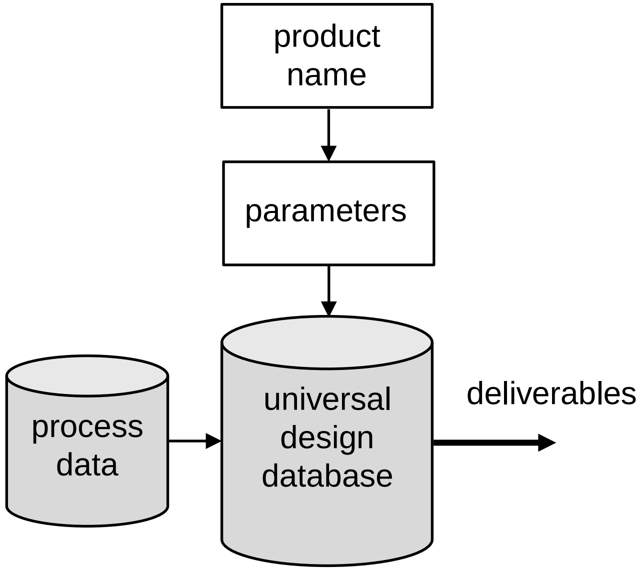
- Standardize everything
- Single database: all data in one place
  - Use parameters to describe everything
  - Tools build from universal templates
  - Target design type, process, metal, Vts, etc.
- Outgoing simulation and QA to insure quality
- Porting effort focuses on key challenges at the cutting edge
- Customize at the margins, with automation

Overall Strategy

- Circuit strategy
  - Understand the key challenges at the new node
  - Address these challenges with robust and flexible circuits
  - Automated sizing and retargeting tools simplify the job
- CAD environment
  - Custom and highly integrated
- Simulation, layout and lab characterization follow suit
- Personnel
  - Requires versatile designers with programming skills
  - Automate repeatable tasks and design “at the margins”

Why the CAD Environment is Key

- I have a passion for CAD
- Controlling the tools is key to supporting multiple designs and processes
- Custom CAD used for
  - Describing, specifying, modeling and simulating designs
  - Creating front and back-end views
  - Packaging designs and documentation
  - Lab testing and report generation
- Parameters drive everything
  - All tools and flows
  - Models and documentation
  - Linked to product names
- IP is “correct by construction”



Descriptions and Flows

- Languages used to describe things allow embedded C to
  - Support parameterization
  - Minimize descriptions for repetitive items
  - Maximize reuse of description elements
- Everything can be generated from command line
- Flows built on one another to allow complete designs to be created from product names
- Example: Configurable ring oscillator

```
xr1 vo[num:1] vdd vss ring(size)

.subckt ring(s=1) vo[n:1] vdd vss
:int i, j;
:for (i = 1, j = n; i <= n; j = i++) {
    xi[i] vo[j] vo[i] vdd vss inv(s)
:}
.ends ring
```

Custom vs. Commercial CAD Tools

- Having custom CAD tools helps, but is not essential to our strategy
- Many of the things that I will describe can be accomplished with commercial CAD tools
- The key is in the scripting, added layers of abstraction, and standardization applied to the CAD tools
- Some tools will make it easier than others
- We chose to use custom tools for most functions to give us
  - More flexibility
  - Access to core behavior
  - Equivalent of site licenses

Robust Base IP Foundation

- Use rich library of process insensitive circuits to achieve
  - Adaptive bandwidth
  - Self training
  - Matched timing by construction
- Design to operate over wide ranges
- Pin programmable for customer flexibility

Design Libraries

- Unified database containing
  - Basic gates
  - Programmable datapath blocks (adders, multipliers, shifters, etc.)
  - Programmable higher-level structures (fractional dividers, etc.)
  - Complete IP blocks
- Process independent
- Options for different design targets (speed/power)
- Customization done with reusable sub-blocks
  - Can control features with parameters
- Can represent any design shipped (all versions)

Front-End Design Tools

- Universal database representation
  - Parameter driven, process independent
  - Incorporates information for automated characterization
- Powerful netlist processing, including synthesis
- Generate many views: circuit, logic, characterization, placement
- Parameter driven characterization environment
  - Setup to perform any measurements that can be imagined
  - Uses large number of processors in server farm
  - Reduces data for easy analysis and creates reports

Front-End Design Tools (Cont.)

- Simulation tool supports
  - Mixed mode simulations (logic, circuit, behavioral, Verilog A)
  - Delay modeling with back annotated parasitics
  - Transient noise analysis
- Timing verification tool supports
  - Statistical timing analysis
  - Arbitrary clock domains
  - Structural checks
  - Interactive environment
  - Automatic characterization of cell libraries

Back-End Design Tools

- Berkeley Magic-based layout editor with powerful additions
- Directed synthesis allows equations of signals and buses
- Place and route (analog and digital)
  - Uses placement hints coded in netlist
  - Placement and routing are repeatable, independent of technology
- Layout re-mapping
  - Allows structures to be mapped to a different set of design rules
- Global routing
- Universal layout manipulation tools for block assembly
  - Perform logic operations on layers or create special layers
  - Perform options and re-sizing
  - Generate structures (equivalent to P-cells, etc.)

Lab Testing

- Fully automated lab characterization environment
  - Setup to drive all equipment (DUT, sources, scopes, environment)
  - Controlled by scripts that are parameter driven from chip ID
- Can be used interactively or in batch mode
- Can be run remotely
- Lab characterization software leverages simulation environment
  - Common data representation format
  - Common characterization functions (e.g. measuring jitter)
- Setup to automatically generate test reports

Maintenance

- Goal is to make it very easy to maintain designs
  - Because knowledge is in design databases and tools, it is very easy for user to perform operations without specific knowledge of the particular design
- Unified design representation makes it easy to incorporate fixes, improvements, etc. on all related designs

Conclusion

- The business goals
  - IP suppliers: amortize design investments
  - IP customers: reduce risk and save time
- Technical goals
  - Maximize reuse using a custom CAD environment
- How it’s done
  - Use robust and flexible circuits
  - Incorporate design and characterization information in design
  - Make maintenance easier using unified design specifications
- Challenges
  - Controlling the CAD tools
  - Finding personnel who can design at the margin