Linzhi Semiconductors

Linzhi Working Papers
No 9

ASIC Feasibility Study in 5 Steps.

by Chen Min

March 2019

Keywords: LWP9, ASIC Design, ASIC Feasibility, Trade-off

This publication is available on https://linzhi.io
Telegram discussion https://t.me/LinzhiCorp

ASIC Feasibility Study in 5 Steps

Linzhi ASICs
Mar 11 · 3 min read

> *Can you make an ASIC for algorithm X?*
> *How fast can it be?*
> *Can it be better than Y?*

Today we want to document the 5 steps that we typically think through, roughly, to answer such questions.
We intentionally leave out considerations for platform, programmability and software stack, industry trends or competition.
This article is about the technical feasibility of a chip design for a given algorithm.

. . .

1. Math First

If we have an algorithm's math description, we start from that. If we only have source code, we need to convert it to a math description first.
The reason for going to math first is that hardware may use a fundamentally different way to compute an algorithm.
Example task: Search how many data items X exist within a database P and locate them. Software may need a data structure, maybe a hash table, and many memory accesses. Hardware may be able to produce all results in one clock cycle using a CAM (content-addressable memory).
Many things like search and sort can have a different implementation in hardware.

. . .

2. Optimization Target

We need to clarify the goal of the optimization. Optimize for high throughput? For low latency? Low power? Low energy? Low cost? Fast

time-to-market? Security? High reliability?
You cannot get all of them at once, so you have to pick priorities. For example in a Bitcoin miner, 1st is low energy, 2nd high throughput, 3rd low cost.
After this phase you have constraints, such as budget, target performance, maximum power, target delivery time, etc.

. . .

3. Hardware-Software Boundary

An algorithm may not benefit from a full hardware implementation. We need to decide which part to run in software, and which to run in hardware, after careful algorithm study.
Normally hardware will handle the major data flow for high-performance. Software can handle complex protocols. However if analysis shows the protocol processing to be the performance bottleneck, that part will also move to hardware.

. . .

4. Building Blocks

There are three major kinds of building blocks (called IP): logic, memory, IO. Each has nearly infinite types of them available.

For example let's look at one of the smallest IP's: an integer adder, to sum up two numbers. For smallest area we choose a Ripple-Carry adder, for lowest latency we choose Kogge-Stone. There are many others: carry-skip, carry-lookahead, Brent-Kung, Han-Carlson, conditional sum, Sklansky, Ling, Jackson, etc. You need to carefully choose because each one has a different PPA (power, performance, area).
The algebraic logic on non-critical paths can normally be optimized automatically by EDA tools.

Same for memory: First there is on-chip memory called SRAM. SRAM is fastest but also most expensive, and there are many kinds of SRAM. eDRAM is much lower cost but very limited. eFlash is cheap but very slow, there is MRAM, RRAM, Regfile, etc.

Off-chip memory: PSRAM, DDR, LPDDR, QDR, RLDRAM, GDDR, HBM, HMC, NAND Flash, NOR Flash, FRAM, PCM, …

Based on project contraints, you can easily rule out most of them. For the rest you need to contact the vendor to sign an NDA, get detailed information, and price quote. Then you decide what IP and peripheral to use, based on trade-off of cost, performance, power, etc.
Like the Nvidia CEO said “It’s much more expensive for consumer goods. But storage prices will fall. Everything is fine with HBM. I love HBM. But I love GDDR6 (graphics memory) so much more.”
https://optocrypto.com/ceo-of-nvidia-says-gddr6-is-cheaper-than-hbm-for-choosing-next-gen-gpu-memory/
They made a trade-off for cost.

If no good IP can be found for the target application, we may ask a vendor to customize an IP for our application. Or we can design it ourselves from the transistor level. In Bitcoin miners you find a large ratio of fully custom designed circuit blocks such as dynamic flip-flops.

. . .

5. Physical Implementation

- Foundry
- Process node
- Floor plan
- Simulation
- Clocking
- Power distribution network
- Timing corner
- Signal integrity
- Temperature range
- Process variation
- Aging

- Pinout
- Packaging
- Thermal management
- Mechanical reliability
- ESD protection
- Verification
- Tapeout
- Process Tuning
- Tester
- Binning/Grade
- PCB
- Assembly
- Software integration
- Burn-in
- Shipping

. . .

Done :)

Chen Min
Linzhi Shenzhen
Telegram: https://t.me/LinzhiCorp