

NGÔN NGỮ THIẾT KẾ XỬ LÝ LAYOUT KỸ THUẬT CAO

HYPER TECH MASK DESIGN PROCESSING

SVTH: LÊ THỊ VĨNH HẰNG
 TRƯƠNG HOÀNG ANH TUẤN
 TRẦN ANH MINH
 VÕ HOÀNG PHƯƠNG DUNG
 TRẦN THỊ MINH PHƯƠNG

Lớp 03T1, Khoa CNTT, Đại học Bách Khoa Đà Nẵng

CBHD: ThS. ĐẶNG BÁ LƯ

Khoa Công Nghệ Thông Tin – Đại học Bách Khoa Đà Nẵng

KS. VÕ TÙNG PHAN

GD Công ty SDS – Silicon Design Solutions – CN Đà Nẵng

TÓM TẮT

Xây dựng thư viện trong đó chứa những hàm cơ bản để có thể vẽ được các thiết bị cơ bản (Nmos, Pmos, Inverter) theo đúng định dạng GDSII, xây dựng công cụ trình biên dịch xử lý các thông tin từ thư viện mà người sử dụng muốn thực hiện và giao diện để hiển thị sơ đồ theo yêu cầu của người sử dụng.

- ✓ Thư viện là nơi chứa tất cả những hàm cần thiết cho việc vẽ các thiết bị cơ bản theo đúng luật thiết kế (tệp được đưa vào từ bên ngoài và có thể thay đổi được). Kết quả của việc xây dựng thư viện là đưa ra được tệp vẽ các thiết bị cơ bản theo đúng định dạng GDSII.
- ✓ Công cụ trình biên dịch được xây dựng theo một ngôn ngữ mới, có hỗ trợ việc xử lý các câu lệnh điều kiện, tính toán và các câu lệnh tương ứng với các hàm được xây dựng trong thư viện để người sử dụng vẽ. Đầu vào của trình biên dịch là câu lệnh người sử dụng muốn thực hiện, đầu ra là kết quả của câu lệnh đó được in ra ở màn hình điều khiển hay tệp định dạng gds.
- ✓ Công cụ Viewer- Editor để đưa ra được sơ đồ của các thiết bị cơ sở. Đầu vào của Viewer- Editor là tệp định dạng GDSII (gds)

ABSTRACT

The thesis is about building a library of all basic functions which can be used to draw basic devices (NMOS, PMOS, and Inverter) in the proper format of GDSII, a compiler tool which processes the required information from the library as well as an interface to display the required layout. The project would be divided into the following subsystems:

- ✓ Creating a library: includes all basic functions for drawing basic devices in design rule, which can be imported and changeable. The result of building the library is listing out all basic devices in the GDSII format.
- ✓ Compiler tool: creating by a new language in order to support all commands which are built in the library. The user command is known as the compiler's input, while its result will be known as the output and be displayed on screen as gds format.
- ✓ Viewer-Editor tool: to show the layout of basic devices. The input of the Viewer-Editor is known as gds file format

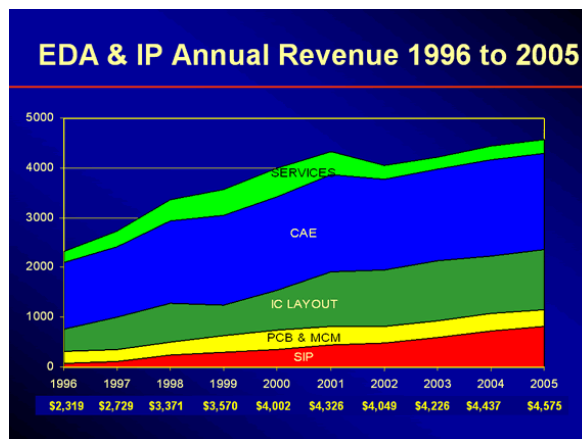
1. Sơ lược về CAD và ASIC

❖ CAD: Computer Aided Design:

Trong cuộc sống hiện đại ngày nay, có rất nhiều thứ được chế tạo bởi IC. Và để có được chúng, cần phải chế tạo theo thiết kế (design). Công cụ dùng để hỗ trợ cho việc thiết kế IC là EDA (**Electronic design automation**). EDA là một tập hợp các công cụ hỗ trợ cho việc thiết kế và sản xuất các hệ thống điện tử từ printed circuit boards (PCBs) đến integrated circuits. Đôi khi nó còn được gọi dưới tên khác là ECAD (electronic computer-aided design) hay gọi đơn giản là CAD.

EDA trong lĩnh vực điện tử đã phát triển rất nhanh cùng với công nghệ bán dẫn (semiconductor technology). Các nhà máy chế tạo sản phẩm với chất liệu bán dẫn và các công ty thiết kế sử dụng các phần mềm EDA để đánh giá các thiết kế dùng trong sản xuất.

Lợi nhuận của CAD:



Hình 1: Biểu đồ lợi nhuận hàng năm của EDA&IP

Nguồn: www.edac.org

Diễn hình như Cadence, một công ty làm trong lĩnh vực này số doanh thu là rất lớn. Chỉ một license của chương trình đã trị giá hàng ngàn đô la Mỹ

Hiện nay Việt Nam đang một thị trường rất tốt và mở cửa cho các nhà đầu tư nước ngoài vào lĩnh vực công nghệ cao. Intel cùng nhiều hãng khác đang ra sức thu hút nhân lực Việt Nam. Bên cạnh đó cũng có những công ty như SDS - Silicon Design Solutions là một trong những công ty đầu tiên ở Việt Nam từ gần 10 năm nay, chuyên thiết kế về bộ nhớ nhúng (Embedded Memory), và cung cấp các dịch vụ thiết kế vi mạch thích hợp (ASIC) trên các công nghệ đúc và thiết bị bán dẫn tiên tiến nhất hiện nay: 90nm, 65nm và 45nm.

Ngoài ra thiết kế vi mạch có tầm quan trọng lớn đối với một quốc gia. Đơn cử như trong lĩnh vực Quốc Phòng, vấn đề bảo mật là tối cần thiết. Ngoài ra, công nghệ ngày càng phát triển, vi mạch có mặt trong mọi lĩnh vực của đời sống nên sự phát triển của lĩnh vực thiết kế vi mạch cùng với CAD ở Việt Nam là điều hiển nhiên.

❖ ASIC : Application-Specific Integrated Circuit

ASIC được thiết kế dành cho một ứng dụng cụ thể. ASIC ngày nay được ứng dụng hầu như khắp mọi nơi, như vi xử lý của [điện thoại di động](#), chip xử lý trong các máy móc tự động, các phương tiện truyền thông, xe cộ, tàu vũ trụ, các hệ thống xử lý, các dây chuyền công nghiệp...

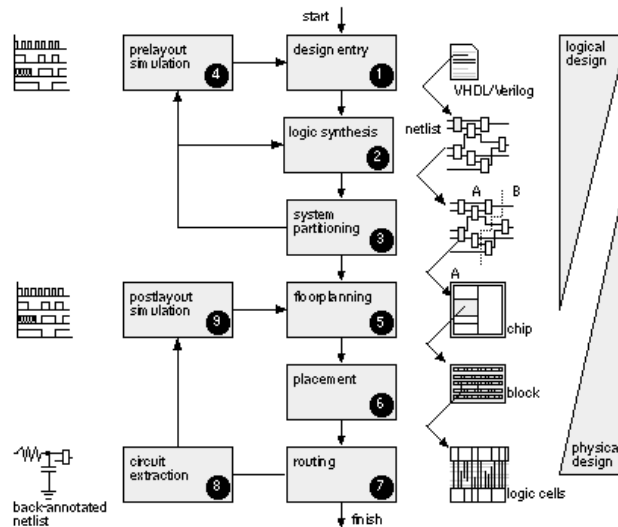
Thay vì phải xem xét nhiều khía cạnh trong khi thiết kế chế tạo IC chuẩn thì công nghệ ASIC cho phép tối ưu hóa thiết kế theo một mục đích cụ thể (specific task) nên sẽ cho hiệu suất cao hơn, cụ thể là cho phép lưu giữ lượng mạch logic lớn hơn so với các chip chuẩn cùng kích thước. Bên cạnh tính tin cậy cao thì IC đơn lẻ còn chiếm dụng ít không gian hơn trên bo mạch in, kéo theo giá thành rẻ hơn so với 1 hệ thống có cùng mục đích sử dụng nhiều IC chuẩn. Trình tự thiết kế ASIC theo các bước :

1. Thiết kế logic (logic design)
2. Chọn kỹ thuật thích hợp thiết kế mạch vật lý (physical design)
3. Chế tạo chip (fabrication) bởi công ty chuyên nghiệp.

Bản thiết kế ASIC thường được viết trên HDL ngôn ngữ mô tả phần cứng như Verilog, VHDL. Dựa theo công nghệ sản xuất và thiết kế có thể chia ASIC thành các loại cơ bản như sau:

- ASIC trên cơ sở thư viện phần tử logic chuẩn (*standard-cell-based ASIC*)
- ASIC dựa trên mảng logic (*gate-array-based ASIC*)
- ASIC đặc chế hoàn toàn (*full-custom ASIC*)
- ASIC tiền cấu trúc (*structured/platform design*)
- ASIC dùng thư viện phần tử logic và các phần tử thiết kế sẵn (*cell libraries, IP-based design*)

ASIC flow:

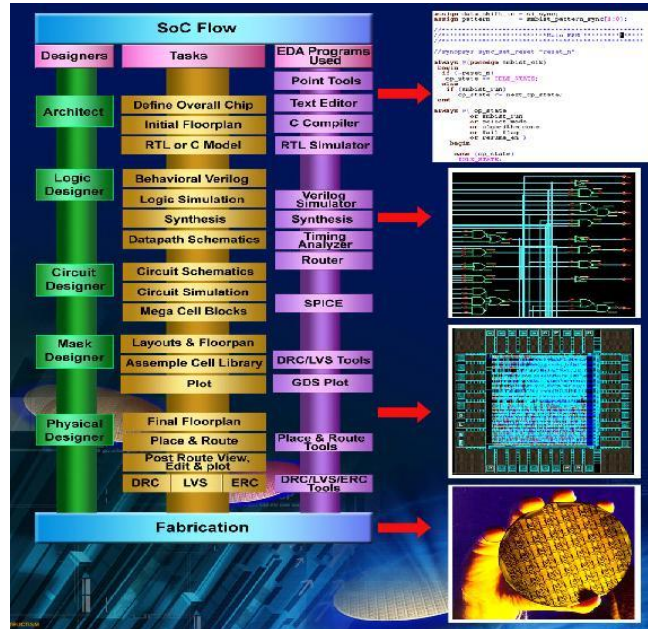


Hình 2: Lược đồ của ASIC

1. **Mô tả bài toán:** sử dụng ngôn ngữ mô tả phần cứng HDL (VHDL hay Verilog) (*VHDL by Department of Defense in 1980s and standardized by IEEE in 1993 - Verilog is created by Cadence in 1989 and standardized by IEEE in 1995*)
 2. **Tổng hợp logic:** dùng HDL và công cụ tổng hợp logic để xây dựng netlist – là sự mô tả các tế bào (cell), các khối (block) và kết nối (interconnect) giữa chúng
 3. **Phân chia hệ thống:** chia hệ thống lớn thành các phần thích hợp
 4. **Mô phỏng tiền layout:** kiểm tra tính đúng đắn của thiết kế (tiền layout = sơ đồ mạch logic – chỉ gần đúng với thực tế)
 5. **Sắp xếp các khối trên chip:** sắp xếp các khối của netlist trên chip. Nên xem xét cả khía cạnh vật lý và logic khi thiết kế bước này
 6. **Bố trí cell:** định vị cell bên trong khối
 7. **Thiết kế tuyến:** kết nối giữa các cell và các khối
 8. **Kiểm tra tính hợp lý của bước 7:** tính toán trở kháng và dung kháng lớp interconnect
 9. **Mô phỏng hậu layout:** kiểm tra khả năng làm việc ổn định của toàn bộ thiết kế trong trường hợp có thêm tải từ lớp interconnect (hậu layout = sơ đồ mạch thực tế)
- Các bước thiết kế 1 - 5: logic.
 Các bước thiết kế 5 – 9 : vật lý.
- ❖ **SOC : System On Chip.**

- Nhiều IC đơn lẻ được kết nối với nhau, thiết kế cho mục đích sử dụng riêng được gọi là SoC.
- Bao gồm bộ vi xử lý, bộ nhớ, IO, ... được đưa chung vào một package được gọi là chip.
- Có trong máy điều hòa nhiệt độ, xe hơi, tủ lạnh, máy giặt,...

SoC flow:



Hình 3: Quy trình của SOC

2. Phạm vi đề tài giải quyết: Mask design

Trước khi có EDA, IC được thiết kế hoàn toàn bằng tay. Một số công cụ sử dụng các phần mềm hình học để tạo ra các [tape](#) cho máy [photoplotter](#). Quá trình này là quá trình đồ họa cơ bản, thực hiện việc chuyển đổi bằng tay từ điện tử sang đồ họa ([with the translation from electronics to graphics done manually](#)). Công ty nổi tiếng nhất trong lĩnh vực này là Calma, sở hữu định dạng GDSII. Sau này tuy đã có nhiều thay đổi đáng kể, hỗ trợ nhiều hơn cho việc thực hiện layout nhưng vẫn còn một số bất cập, ví dụ như phải porting lại từ đầu khi muốn thay đổi kích thước của chip trong cùng một dòng sản phẩm.

Hai ứng dụng của đề tài trong mask design:

1. Porting:

Trong lĩnh vực thiết kế vi mạch, khi công nghệ càng phát triển thì kích thước của sản phẩm càng được thu nhỏ lại. Khi thu nhỏ kích thước chip, vẫn thực hiện theo quy trình cũ, chỉ có một số thay đổi nhỏ về chiều dài, phiên bản... Việc thay đổi này được gọi là porting.

Quy trình porting như sau:

- + Design architect (Thiết kế kiến trúc)
- + Simulation (Mô phỏng)
- + Resize (Thay đổi kích thước)
- + Redraw (Vẽ lại chip)
- + Tiếp tục thực hiện công việc theo quy trình cũ.

Trong đó design architect thường không có hoặc thay đổi rất ít, resize cũng chỉ thay đổi một số giá trị như độ rộng (width), độ lớn (length) theo design rule mới, còn lại các công việc đều không thay đổi. Nếu thực hiện từ đầu sẽ mất thời gian và dễ dẫn tới sai sót.

Nhu cầu đặt ra là xây dựng được một tool có thể tự động được quá trình porting để vẽ ra được chip theo kích thước của design rule mới, dựa vào đó kỹ sư layout có thể thay đổi một số giá trị để đưa ra được sản phẩm như mong muốn.

Đề tài đã thực hiện được việc porting tự động, tức chỉ thay đổi một số giá trị trong quá trình resize chứ không thực hiện lại toàn bộ quy trình. Làm như vậy, giai đoạn Layout Verification sẽ được bỏ qua, do chip mới được porting từ chip cũ, chip đã được kiểm tra DRC và LVS nên sẽ không cần thực hiện bước kiểm tra đối với chip được porting. Nhờ vậy sẽ giảm được các lỗi sinh ra do con người, giảm thời gian thực hiện do đó sẽ tăng chất lượng và hiệu quả công việc và giảm chi phí sản xuất.

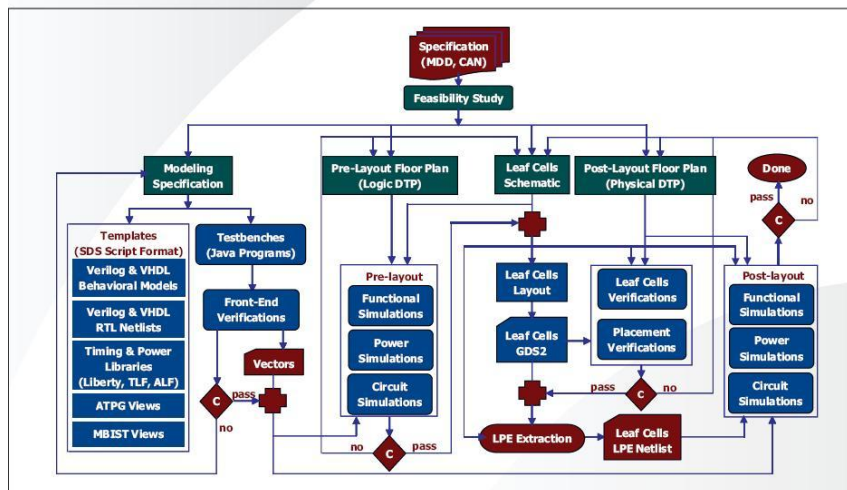
2. Memory Compiler:

Ứng dụng của Tool cho quy trình sản xuất Embedded Memory

➢ Tools HTMDP sẽ giúp cho việc tạo các leafcell một cách tự động. Đặc biệt nó có thể giúp lắp ghép các leafcell lại với nhau tùy thuộc các thông số đưa vào

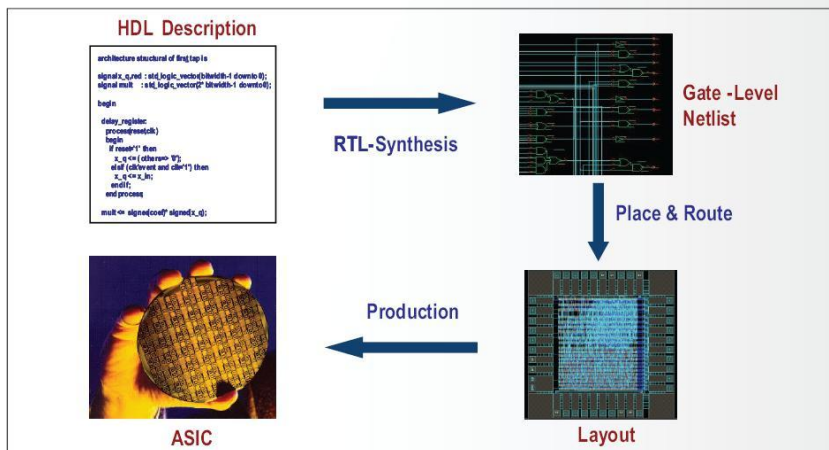
➢ Như vậy nó sẽ giúp tự động hóa một giai đoạn rất đặc biệt của quy trình tạo Embedded Memory

EMBEDDED MEMORY DESIGN FLOW

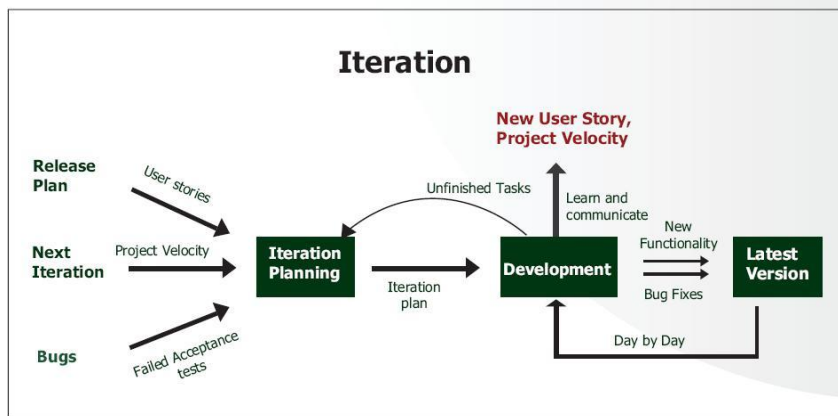


Hình 4: Lược đồ thiết kế bộ nhớ nhúng

SOC DESIGN FLOW



Hình 5: Lược đồ thiết kế hệ thống trên chip

SOFTWARE DESIGN FLOW

Hình 6: Lược đồ thiết kế phần mềm

3. Kết luận

Các công cụ sử dụng đã có đầy đủ các chứng năng của đề tài đặt ra: hỗ trợ câu lệnh cần thiết và chủ yếu của một ngôn ngữ lập trình. Đặc biệt cho phép người dùng tạo hàm và thủ tục cũng như phương thức xử lý chính của chương trình (phương thức main) một cách linh hoạt. Xây dựng được các hình vẽ layout cần thiết cho người kỹ sư thiết kế vi mạch thông qua các câu lệnh tạo tệp GDS từ thư viện. Sử dụng công cụ HTMDP giúp tự động hóa giai đoạn vẽ layout, giảm bớt các sai sót so với quá trình thao tác bằng tay.

Vì thời gian có hạn nên cú pháp của ngôn ngữ HD_Language chưa thân thiện với người sử dụng. Việc xây dựng thư viện chỉ mới cung cấp được một số chức năng cơ bản để vẽ được những thiết bị cơ bản, và chưa thực hiện được với những layout đòi hỏi độ phức tạp cao. Giao diện của TextEditor còn đơn giản, chưa hỗ trợ các chức năng cần thiết của một TextEditor chuyên dụng cho việc lập trình. Ngoài ra còn thiếu một số chức năng cần có của một trình biên dịch: như debug lỗi...

Nhưng đề tài cũng mở ra nhiều hướng phát triển mới trong tương lai, như việc xây dựng cú pháp thân thiện và phù hợp hơn. Cải thiện và nâng cấp thêm các tính năng cần thiết TextEditor, bổ sung chức năng debug. Phát triển ngôn ngữ thành ngôn ngữ hướng đối tượng, xây dựng thêm cây thư mục để quản lý các project dễ dàng hơn. Nâng cấp thư viện, cung cấp nhiều hàm hỗ trợ hơn, làm phong phú thêm cho ngôn ngữ HD_Language để hỗ trợ tạo ra file gds với nội dung phức tạp. Ngoài ra hướng xây dựng lâu dài và chiến lược là tích hợp các công cụ lại thành một thể thống nhất.

TÀI LIỆU THAM KHẢO

❖ Tài liệu của công ty SDS – Silicon Design Solutions:

- [1] An Introduction to MC.
- [2] SDSMC User Guide.
- [3] SDSScript

❖ Các tài liệu sách khác:

- [4] Trần Đức Quang, *Trình biên dịch thiết kế và cài đặt*, NXB Đại học Quốc gia TP Hồ Chí Minh, sản xuất 2002

❖ Các trang web cần thiết:

- [5] Định dạng của GDSII <http://www.iue.tuwien.ac.at/phd/minixhofer/node52.html>
- [6] Mã nguồn mở của công cụ ViewerEditor http://sourceforge.net/project/showfiles.php?group_id=121668
- [7] Cài đặt và sử dụng thư viện QT cho công cụ ViewerEditor <http://trolltech.com/>