Tema 7
Disseny Microelectrònic: Metodologies, tècniques i eines CAD

Sebastià Bota

Sistemes Microelectrònics
Assignatura Optativa

Què és disseny microelectrònic

- Conjunt de processos necessaris per passar d'una idea a un producte
- Com en la major part dels camps de l'enginyeria, el disseny VLSI compren:
  - Definició
  - Construcció
  - Simulació
  - Elaboració de prototipus
  - Test
  - Validació

Metodologies de disseny

- La complexitat dels sistemes microelectrònics creix any darrera any
- La productivitat creix més lentament
  - Noves eines CAD
  - EDA (*Electronic Design Automation*)

– Cal seguir metodologies de disseny estructurades

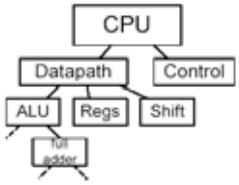
Metodologies de disseny

- Una metodologia consisteix en una seqüència ordenada de tasques (realitzades amb l'ajuda d'eines CAD) que s'empren per completar el disseny d'un circuit.
- Diferents grups/empreses poden emprar les mateixes eines però efectuar una seqüència de tasques diferent.
- D'altres poden introduir eines pròpies (no comercials) per completar aquest flux

Metodologies de disseny

2 grans families....

- Top-down
  - Qué necessit per fer aquest circuit?



Bottom-Up

- Que tenc per dissenyar aquest circuit?

El disseny es monta com un Lego®

Metodologies de disseny: Top-down

- Selecció de l'algoritme a implementar (optimització)
- Selecció de l'arquitectura (optimització)
- Definició dels moduls funcionals
- Definició de la jerarquia del disseny
- Dividir en blocs més simples – dividir en blocs més simples – dividir en blocs més simples
- Definir les unitats necessàries (sumadors, màquines d'estat, etc.)
- Floor-planning
- Mapejar en la tecnologia seleccionada (sintesis, esquema, layout)
  (canviar algoritme/arquitectura en funció dels resultats: velocitat o àrea)
- Eines de simulació a nivell de comportament

Metodologies de disseny: Bottom-up

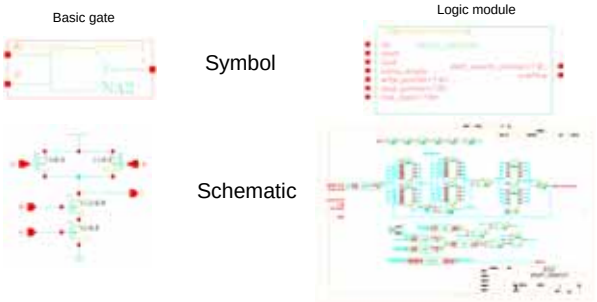
- Build gates in given technology
- Build basic units using gates
- Build generic modules of use
- Put modules together
- Hope that you arrived at some reasonable architecture
- Gate level simulation tools

Old fashioned design methodology like discrete logic

Comment by one of the main designers of a Pentium processor
The design was made in a typical top - down , bottom - up , inside - out design methodology

Schematic based

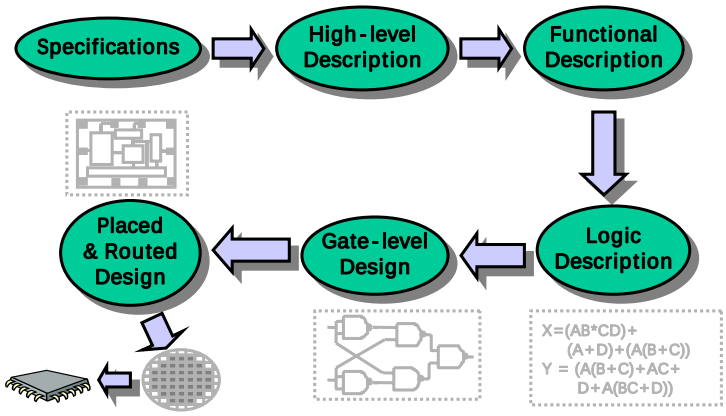
- Symbol of module defines interface
- Schematic of module defines function
- Top - down: Make first symbol and then schematic
- Bottom - up: Make first Schematic and then symbol



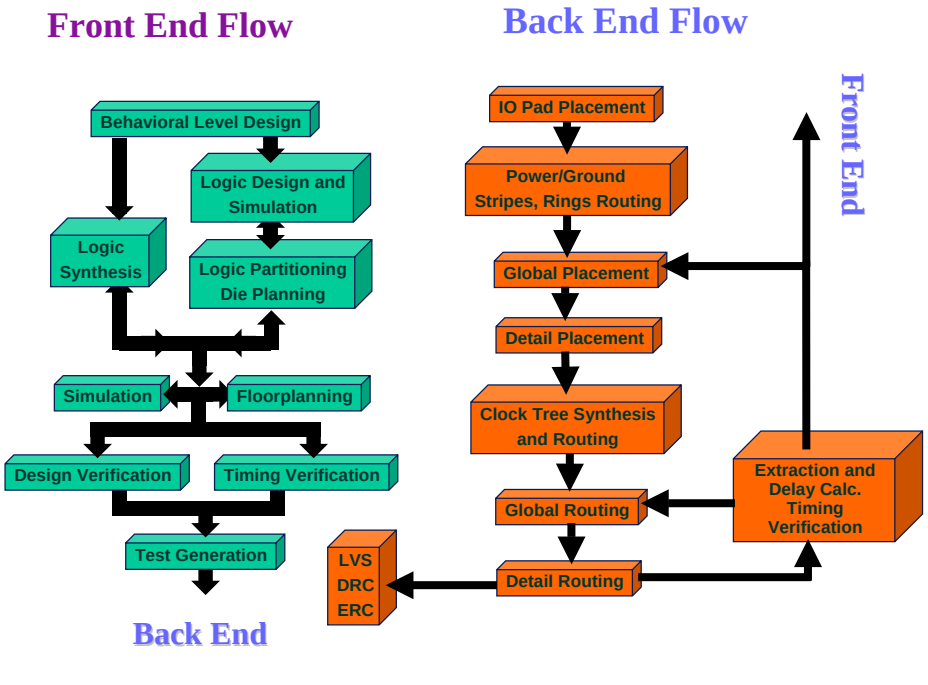
Metodologies de disseny

- Procés en cascada

  - Transformacions entre diferents nivells d'abstracció: de més abstracció a menys



- Idealment el procés es realitzaria sense iteracions



Estratègies de disseny

Disseny Jeràrquic

- Consisteix en dividir sistemes complexos en sub-sistemes més simples
- Es pot aplicar als diferents nivells de representació
  - Comportamiento: Subrutinas
  - Estructural: Bloques
  - Físico: Módulos

Disseny Regular

- S'aprofiten les propietats que poden ser comuns
  - Camins de dades (busos)
  - Intentar utilitzar un nombre limitat de components
- Simplifica la simulació
- Simplifica la detecció i correcció d'errors

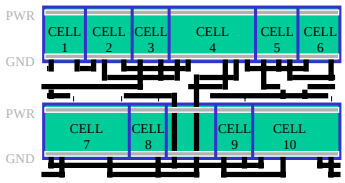
Disseny Modular

- Defineix les interfaces entre blocs diferents
- Cada bloc es pot tractar com una caixa negra
- Diferents dissenyadors sols s'han d'ocupar del seu mòdul

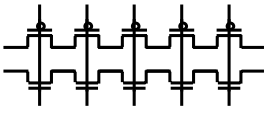
Alternatives de disseny

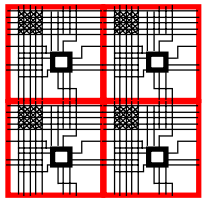
Full Custom

Standard Cells



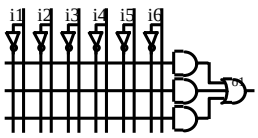
Gate Arrays





Field-Programmable Gate Arrays (FPGAs)

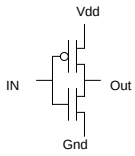
Programmable Logic Devices

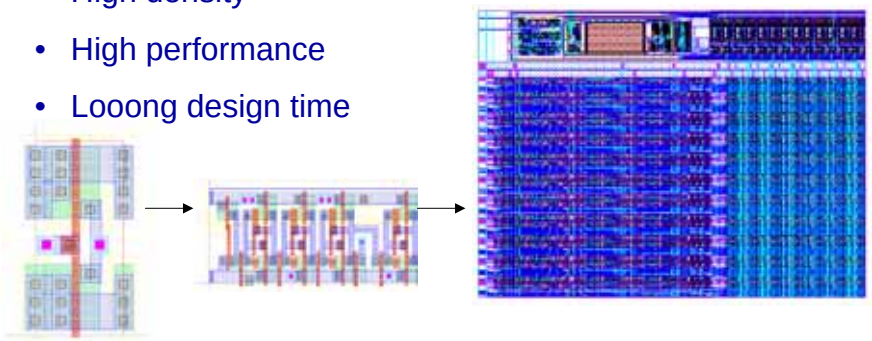


Discrete Components

Full custom

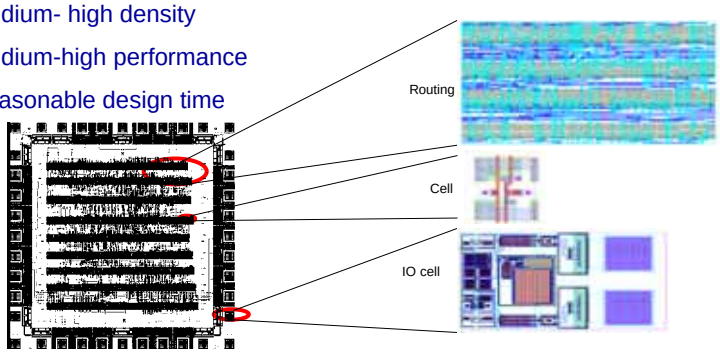
- Hand drawn geometry
- All layers customized
- Digital and analog
- Simulation at transistor level (analog)
- High density
- High performance
- Looong design time





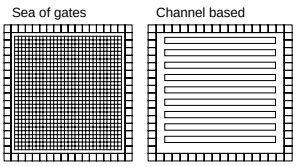
Standard Cells

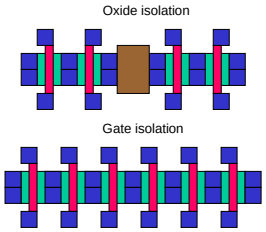
- Standard cells organized in rows (and, or, flip-flops,etc.)
- Cells made as full custom by vendor (not user).
- All layers customized
- Digital with possibility of special analog cells.
- Simulation at gate level (digital)
- Medium- high density
- Medium-high performance
- Reasonable design time

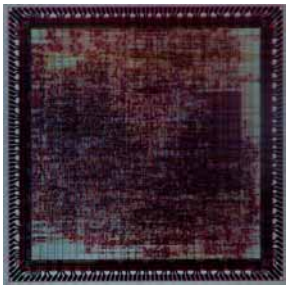


Gate Arrays

- Predefined transistors connected via metal
- Two types: Channel based, Sea of gates
- Only metal layers customized
- Fixed array sizes (normally 5-10 different)
- Digital cells in library (and, or, flip-flops,etc.)
- Simulation at gate level (digital)
- Medium density
- Medium performance
- Reasonable design time

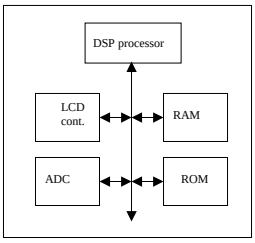






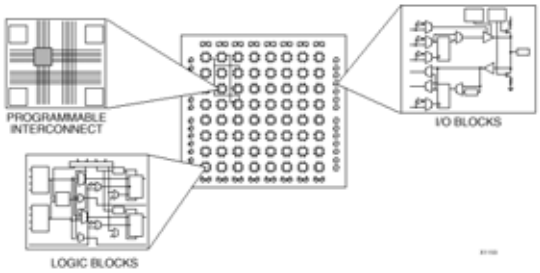
Macro cell

- Predefined macro blocks (Processors, RAM,etc)
- Macro blocks made as full custom by vendor
  (Intellectual Property blocks = IP blocks)
- All layers customized
- Digital and some analog (ADC)
- Simulation at behavioral or gate level (digital)
- High density
- High performance
- Short design time
- Use standard on-chip busses
- “System on a chip” (SOC)



FPGA

- Programmable logic blocks
- Programmable connections between logic blocks
- No layers customized (standard devices)
- Digital only (analog types also exist)
- Low - medium performance (up to a few hundred MHz)
- Low - medium density (< 1M gates)
- Programmable: SRAM,
- Easy and quick design changes
- Cheap design tools
- Low development cost
- High device cost
- NOT a real ASIC
  (Application Specific Integrated Circuit)

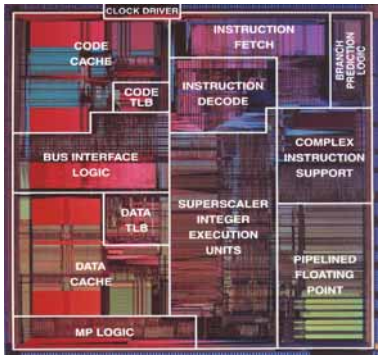


Comparison

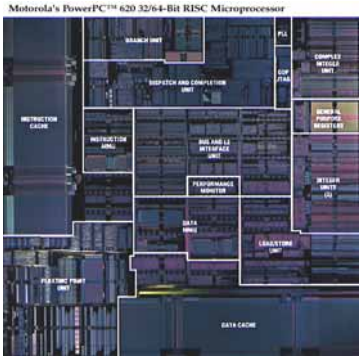
	FPGA	Gate array	Standard cell	Full custom	Macro cell
Density	Low	Medium	Medium	High	High
Flexibility	Low (high)	Low	Medium	High	Medium
Analog	No	No	No	Yes	Yes
Performance	Low	Medium	High	Very high	Very high
Design time	Low	Medium	Medium	High	Medium
Design costs	Low	Medium	Medium	High	High
Tools	Simple	Complex	Complex	Very complex	Complex
Volume	Low	Medium	High	High	High

High performance devices

- Mixture of full custom, standard cells and macro's
- Full custom for special blocks: Adder (data path), etc.
- Macro's for standard blocks: RAM, ROM, etc.
- Standard cells for non critical digital blocks

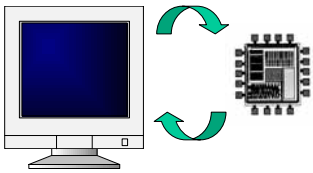


Pentium



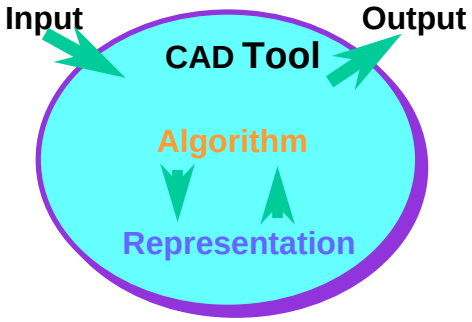
Power PC

Eines CAD



- Disseny procés complicat

- Eines informàtiques de suport al disseny
- C.A.D. : Computer Aided Design



- Estructura i llenguatge representació determinats

Tipus d'eines CAD

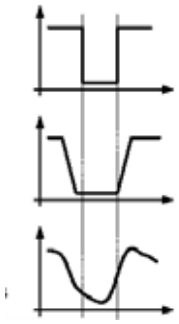
- Eines de simulació
  - Predicció del comportament del circuit que se representa
  - Elèctrics, digital o mixte
- Eines de síntesis
  - Eines de síntesis d'alt nivell
  - Eines de síntesis física: Placement & Routing
    - possibilitat de guiar en funció de paràmetres (àrea, dissipació,velocitat, cost

Tipus d'eines CAD

- Eines d'extracció
  - Exemple: extracció de l'esquema elèctric a partir del layout
- Eines de verificació
  - Comparen resultats de simulació d'un nivell de descripció amb el superior
  - Asseguren el cumpliment de certes regles (layout-regles de disseny)
- Eines de Test
  - Exemple : generació de vectors de test (ATPG)

Simuladors

- Simuladors físics
  - Nivell físic de descripció
  - Equacions físiques (Maxwell)
  - Mètodes numèrics
    - FEM
    - PEEC
    - BEM
- Simuladors elèctrics
- Simuladors lògics
- Simuladors d'alt nivell



Els simuladors també treballen
En diferents nivells d'abstracció

Descripció Spice

- Llenguatge descripció


```
# File name: LDCISE_DF8_schematic.S.
# Subcircuit for cell: DF8.
# Generated on Sep  1 17:33:51 1999.

XCIN_3 ci net18 net6 cn clinvr_1
XCIN_2 ci net9 net18 cn clinvr_1
XCIN_4 cn net13 net6 ci clinvr_1
XCIN_1 cn D net18 ci clinvr_1
XIN_4 net6 net13 invr_2
XIN_3 net18 net9 invr_2
XIN_5 net13 Q invr_3
XIN_6 net6 QN invr_3
XIN_2 cn ci invr_2
XIN_1 C cn invr_2

.SUBCKT invr_3 in out
MN1 out in 0 0  modn  L=0.6e-6 W=4.0e-6
MP1 out in vdd! vdd!  modp  L=0.6 W=7.0e-6
.ENDS invr_3

.SUBCKT invr_2 in out
MN1 out in 0 0  modn  L=0.6e-6 W=2.0e-6
MP1 out in vdd! vdd!  modp  L=0.6 W=3.5e-6
.ENDS invr_2

.SUBCKT clinvr_1 clk in out xclk
MN2 out clk net18 0  modn  L=0.6e-6 W=2.0e-6
MN1 net18 in 0 0  modn  L=0.6e-6 W=2.0e-6
MP1 net10 in vdd! vdd!  modp  L=0.6 W=3.5e-6
MP2 out xclk net10 vdd!  modp  L=0.6 W=3.5e-6
.ENDS clinvr_1

# Transistor models

.include MOS_models.inc

.END
```

Logic Simulation

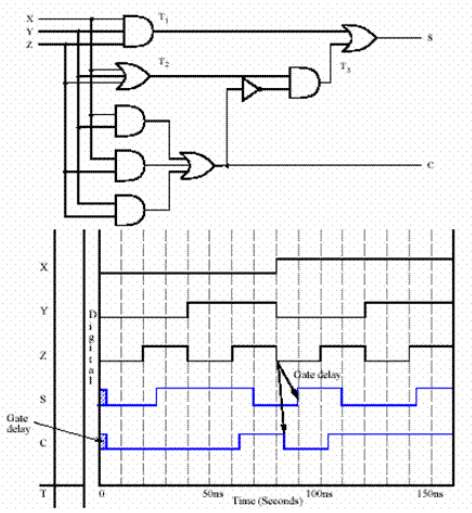
- A simulator interprets the HDL description and produces a readable output, such as a timing diagram, that predicts how the hardware will behave before its is actually fabricated.
- Simulation allows the detection of functional errors in a design without having to physically create the circuit.

Logic Simulation (2)

- The stimulus that tests the functionality of the design is called a test bench.
- To simulate a digital system
  - Design is first described in HDL
  - Verified by simulating the design and checking it with a test bench which is also written in HDL.

Logic Simulation

- Logic simulation is a fast, accurate method of analyzing a circuit to see its waveforms



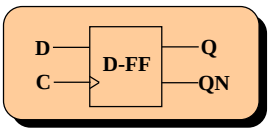
Models i simuladors lògics

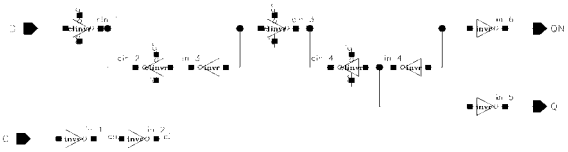
- Descripció lògica d'un FFD incloent especificacions adicionals

```
// Biestable D descripción funcional
// incorporando retardos y otros parámetros
module dflipflop(q,d,c);
  output q;
  input  d,c;

  reg q;  // nodo con memoria

  always    // descripcion de la funcion lógica
    @(posedge c) q=d;

  specify   // descripcion de parámetros adicionales
      specparam area = 787.095;
      specparam capacidad_C = 0.022;
      specparam capacidad_D = 0.020;
      specparam t_C_a_subida_Q = 0.01567
      specparam t_C_a_bajada_Q = 0.01433
      (c => q) = (t_C_subida_Q,t_C_bajada_Q);
      endspecify
endmodule
```

Models i simuladors lògics

- Descripció Verilog estructural del D-FF





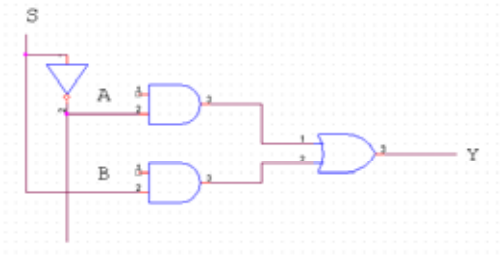
```
//Biestable D descripción estructural
module dflipflop(q,qn,d,c);
  output q, qn;
  input  d,c;

  tri    n1,n2;    // nodos tercer estado
  wire   cn,ci,n3,n4;  // nodos normales

  not in_1(cn,c), in_2(ci,cn);
   otfif1  cin_1(n1,d,cn);
   ot  in_3(n3,n1);
   otfif1  cin_2(n1,n3,ci);
   otfif1  cin_3(n2,n1,ci);
   ot  in_4(n4,n2);
   otfif1  cin_4(n1,n4,cn);
   ot in_6(qn,n2), in_5(q,n4);
endmodule
```

Models i simuladors lògics

- Descripció Verilog funcional del D-FF

```
// Biestable D descripción funcional
module dflipflop(q,qn,d,c);
  output q, qn;
  input  d,c;

  reg q,qn;  // nodos con memoria

  always
    @(posedge c)
      #10 q=d,qn=!d;
endmodule
```

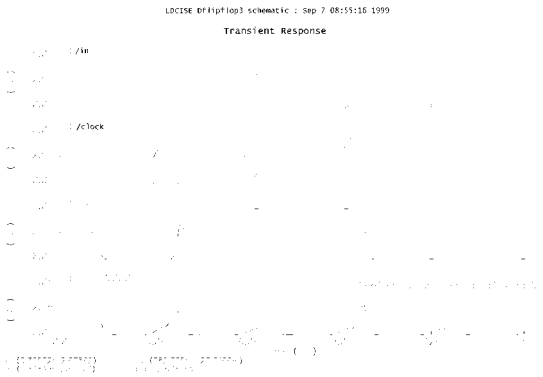
Descr. VHDL : struct. & behav.




```
ENTITY mux IS
PORT(a, b, s: in bit;
          y: out bit);
END mux;
ARCHITECTURE dataflow OF mux IS
BEGIN
     y<= (NOT s AND a) OR (s AND b);
END dataflow;
```

Comportament

Simulació Elèctrica



Simulació lògica

Getting it right - Simulation

- Simulate the design at all levels (transistor, gate, system)
- Analog simulator (SPICE) for transistor level
- Digital gate level simulator for gate based design
- Mixed mode simulation of mixed analog-digital design
- Behavioral simulation at system/module level (Verilog, VHDL)
- All functions must be simulated and verified.
- Worst case data must be used to verify timing
- Worst - Typical - Best case conditions must be verified
  Process variations, Temperature range, Power supply voltage
  Factor two variation to both sides (speed: ½ : 1 : 2)
- Use programming approach to verify large set of functions
  (not looking at waveform displays)

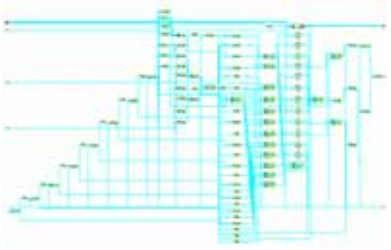
Automatització del disseny: Síntesis

- Síntesis d'alt nivell
  - Pas de llenguatge d'alt nivell HDL a operadors
  - Scheduling : determinar ordre com es fan les operacions
  - Binding : repartir les operacions entre els recursos diponibles
- Síntesis i optimització lògica
  - Blocs a alt nivell = conjunt d'equacions booleanes
  - Implementació lògica combinacional + registre d'estat
  - lògiques a dos nivells i lògiques multinivell
- Síntesis a nivell físic
  - Partició i planificació
  - Placement
    - Cel.les, Pads, Macrocel.les i Canals
  - Routing

Synthesis based

- Define modules and their behavior in a proper language (also used for simulation)
- Use synthesis tools to generate schematics (netlists)

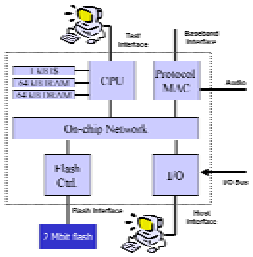
```
always @(posedge clk)
        begin
        if (set) coarse <= #(test.ff_delay) offset;
        else if (coarse == count_roll_over)
                coarse <= #(test.ff_delay) 0;
        else coarse <= #(test.ff_delay) coarse + 1;
        end
```

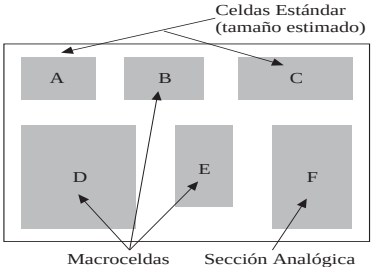


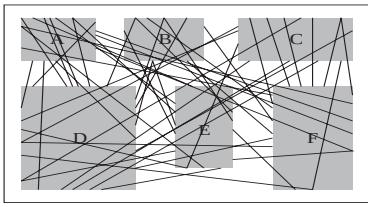
Only possible way to make designs with millions of gates

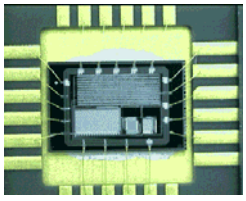
Automatització : síntesis a nivell físic

- Particionat i planificació



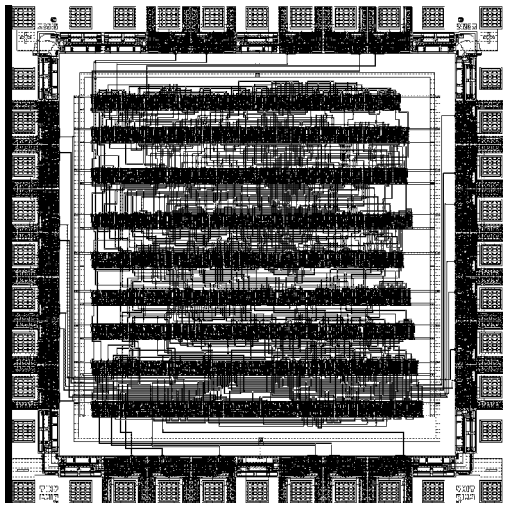






Automatització : síntesis a nivell físic

- Placement & Routing



Automatització : síntesis a nivell físic

- PADS & Core