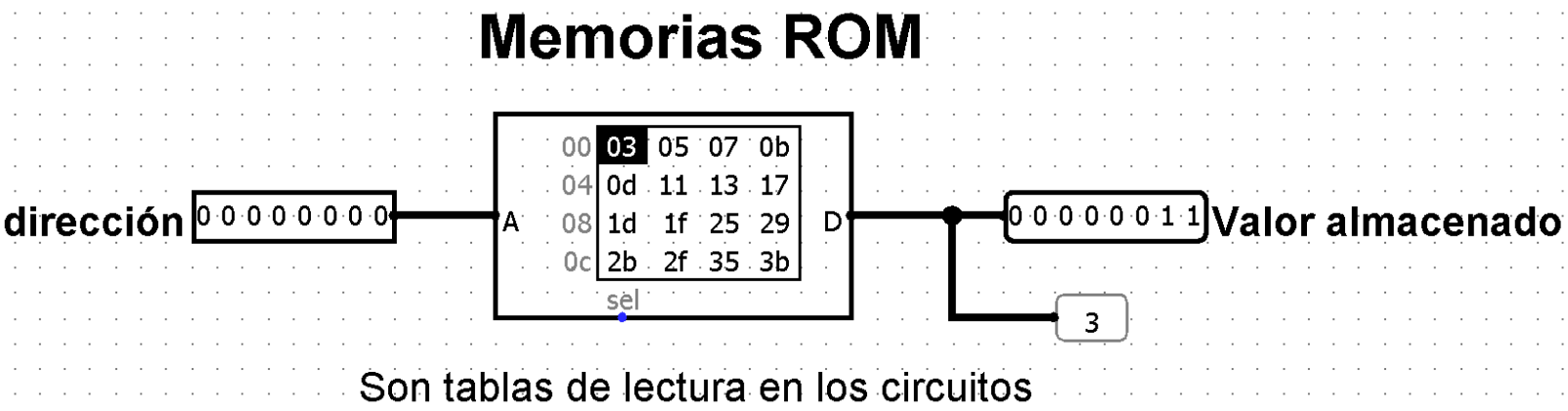
CC3301
Programación de software de sistemas

Memorias ROM y RAM, lenguaje Verilog, implementación de compuertas básicas, la ley de Moore, litografía, los desafíos de las fábricas de chips

Memorias ROM

- Una memoria ROM (*read only memory*) es una memoria que solo se puede leer
- Su contenido viene grabado de fábrica
- Se coloca la dirección en *A* y en *D* aparecen los datos
- Si *sel* es 0, *D* se desconecta



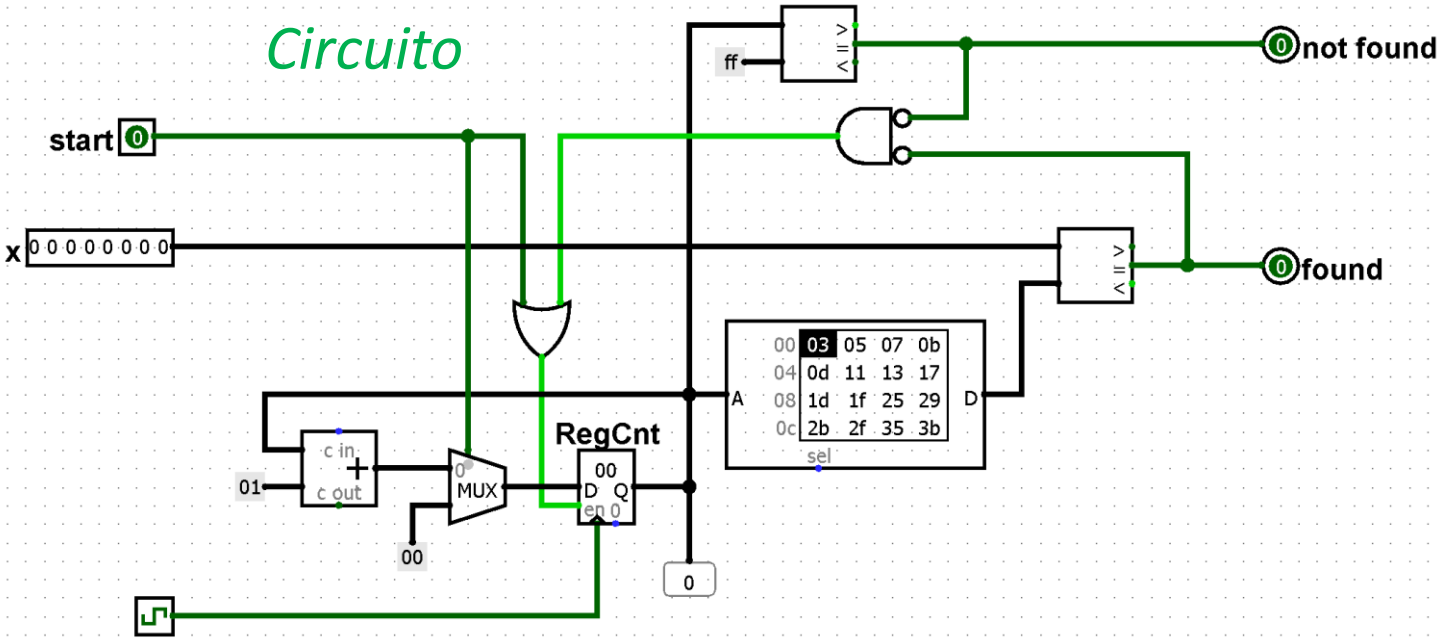
Ejemplo

Circuito que busca si un entero está presente en alguna dirección de memoria

```
int buscar(int x) {
  int cnt= 0;
  while (cnt!=255) {
      if (rom[cnt]==x) return 1;
  }
  return 0;
}
```

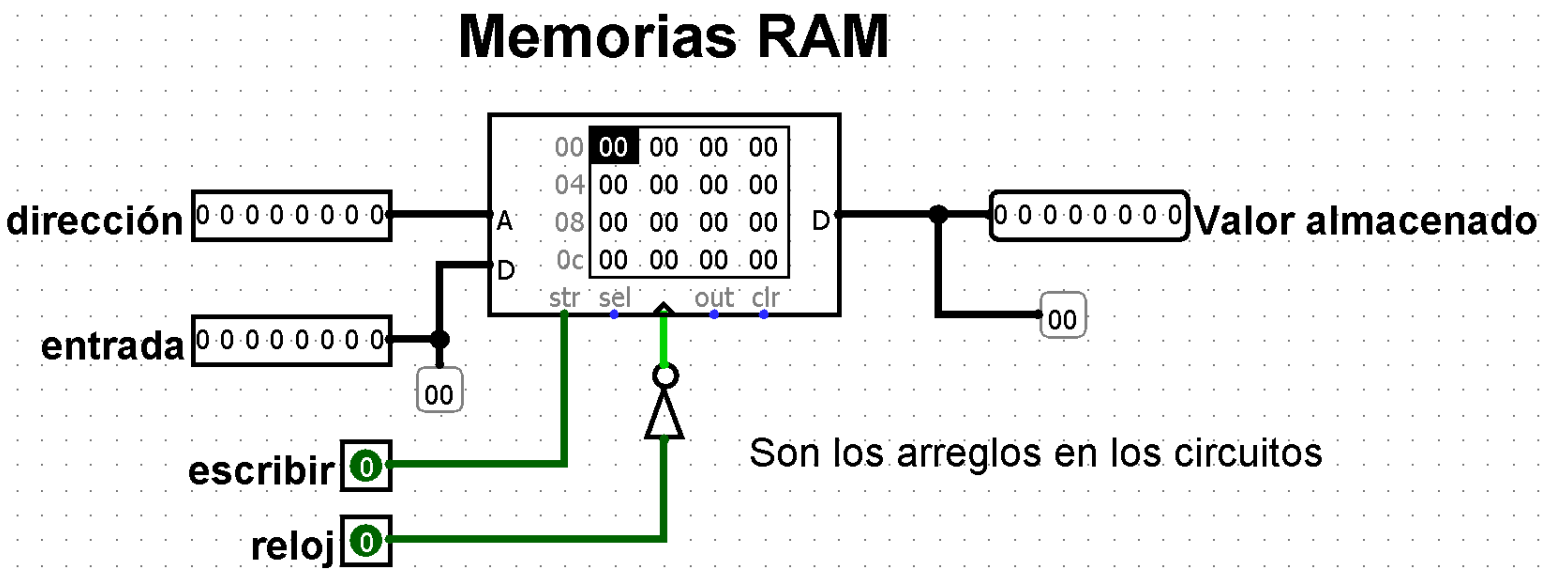
Algoritmo

Circuito



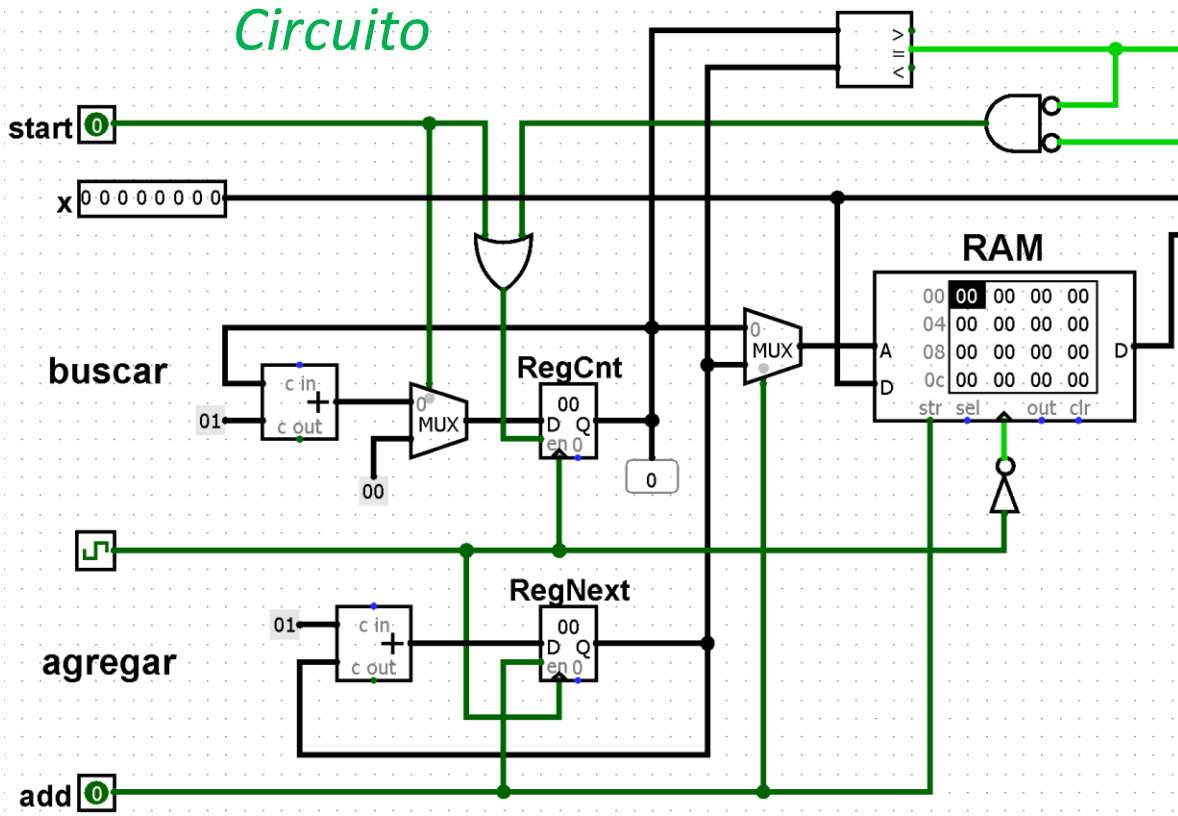
Memorias RAM

- Una memoria RAM (*random access memory*) es una memoria que sí se puede escribir
- El tiempo de lectura y escritura es el mismo para cualquier dirección de la memoria
- Para escribir se coloca un 1 en *str*
- El cambio ocurre en el pulso de bajada del reloj



Ejemplo

Circuito como el anterior pero que además permite agregar un entero

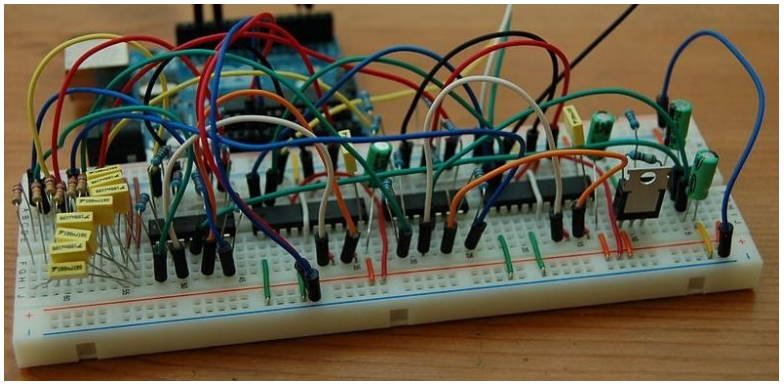



```
int next= 0;
int ram[256];
int buscar(int x) {
   int cnt= 0;
   while (cnt!=next) {
      if (ram[cnt]==x) return 1;
   }
   return 0;
}
int agregar(int x) {
   ram[next++]= x;
}
```

Algoritmo

¿Como hacer sus propios circuitos?

- Protoboards



- FPGAs

  Field Programmable

  Gate Arrays

- ASICs

  Application Specific

  Integrated Circuit

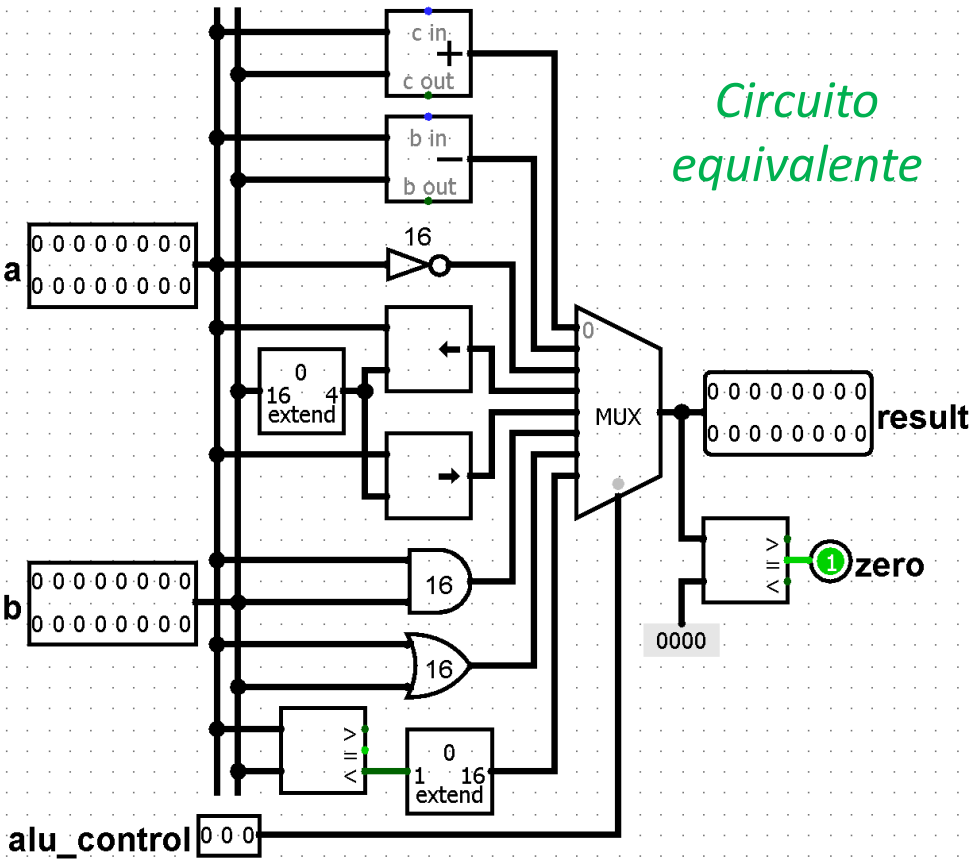
  (muchos millones de dólares)

hardware

software

Hardware Description Language: Verilog

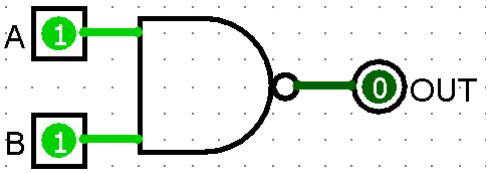
```
// Verilog code for ALU
module ALU(
 input  [15:0] a,  //src1
 input  [15:0] b,  //src2
 input  [2:0] alu_control,

 output reg [15:0] result
 output zero
    );

always @(*)
begin
 case(alu_control)
 3'b000: result = a + b;
 3'b001: result = a - b;
 3'b010: result = ~a;
 3'b011: result = a<<b;
 3'b100: result = a>>b;
 3'b101: result = a & b;
 3'b110: result = a | b;
 3'b111: begin if (a<b) result = 16'd1;
                else    result = 16'd0;
         end
 default:result = a + b; // add
 endcase
end
assign zero = (result==16'd0) ? 1'b1: 1'b0;

endmodule
```



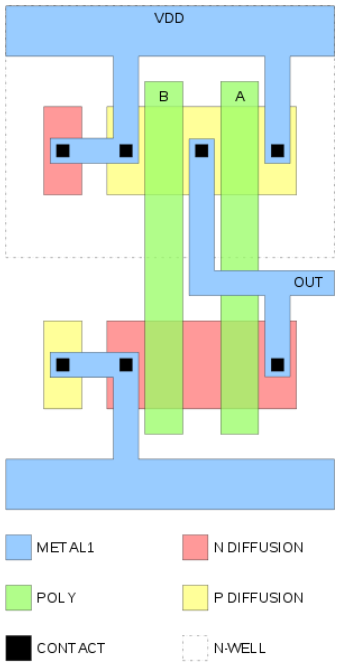
Circuito equivalente

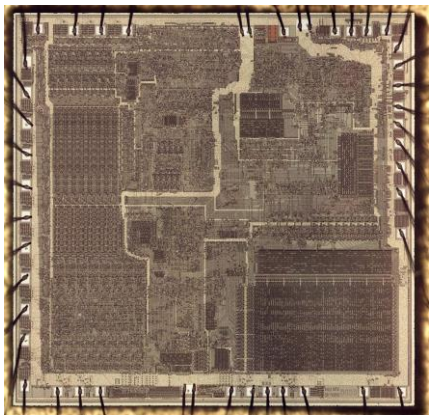
Otros HDLs: VHDL, Chisel, MyHDL

Implementación de las compuertas básicas (video)



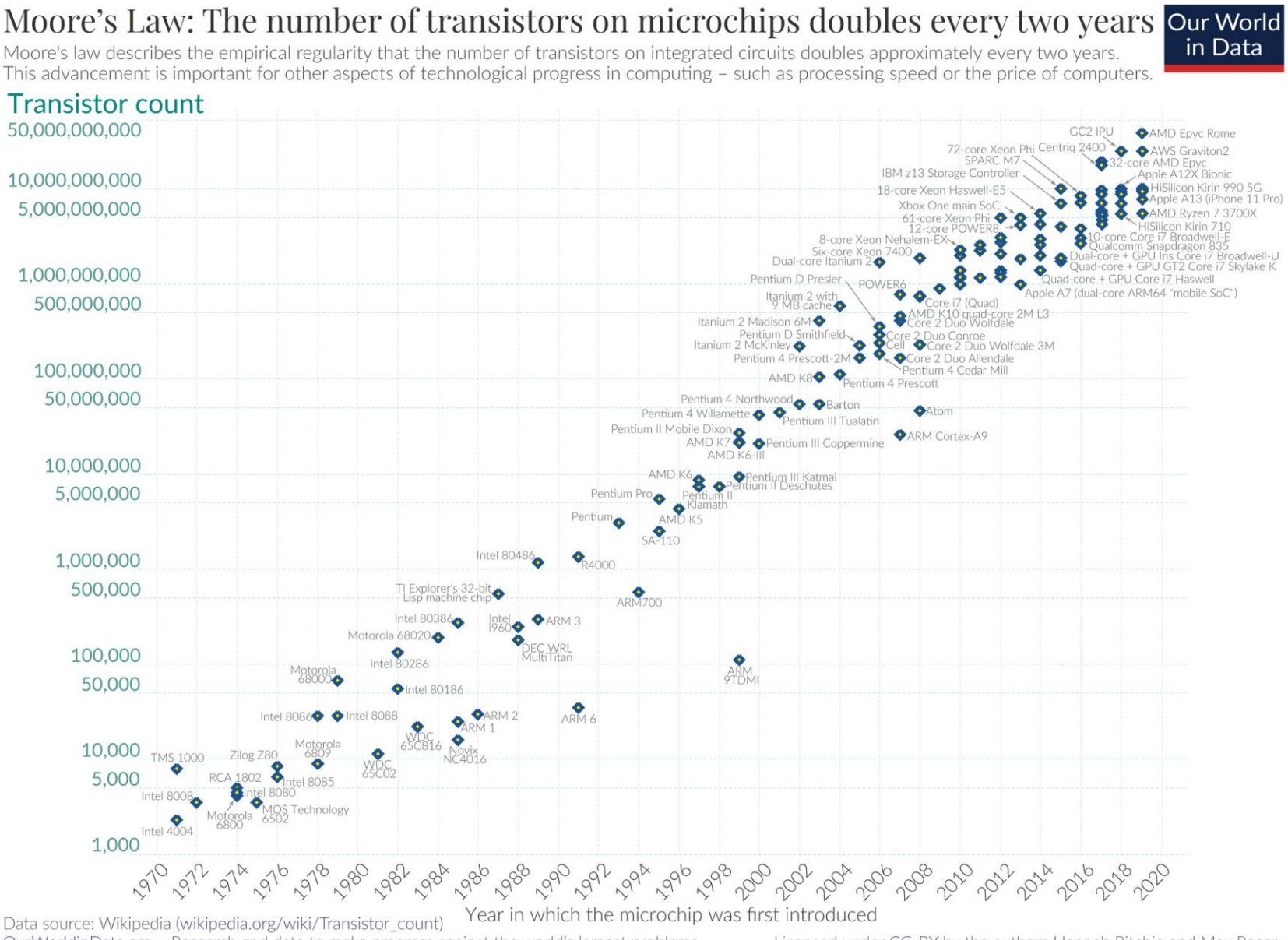
- Para la compuerta más simple, el *nand*, se usan 2 transistores por cada entrada
- Se construyen grabando capas de distintos materiales en el chip de silicio
- *Ley de Moore*: cada ~ 2 años se duplica la cantidad de transistores que se pueden grabar en la misma superficie de un chip
- *Al achicar los transistores se hacen más rápidos y consumen menos energía*
- Desde 1963 hasta ?
- ¡A un precio marginalmente mayor hasta 2010!
- Procesos: 10000 nanómetros en 1971, 1500 nm en 1985, 130 nm en 2003, 5 nm en 2020, …
- Cuidado: los nanómetros ya no dicen nada



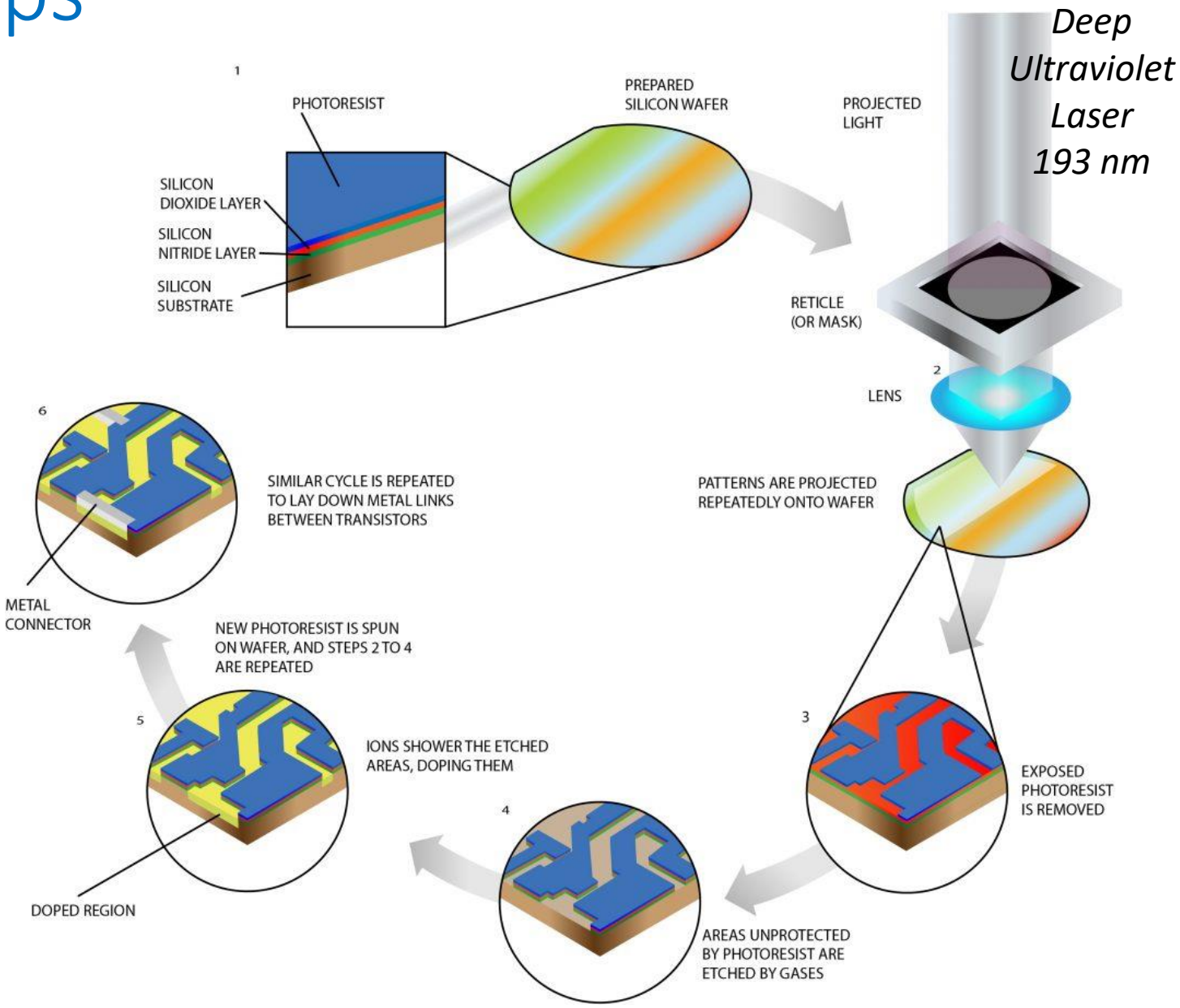


Intel 8086 (1978)
29000 transistores

Ley de Moore


Moore's Law: The number of transistors on microchips doubles every two years
Moore's law describes the empirical regularity that the number of transistors on integrated circuits doubles approximately every two years.
This advancement is important for other aspects of technological progress in computing – such as processing speed or the price of computers.
Our World in Data
Transistor count
50,000,000,000
10,000,000,000
5,000,000,000
1,000,000,000
500,000,000
100,000,000
50,000,000
10,000,000
5,000,000
1,000,000
500,000
100,000
50,000
10,000
5,000
1,000
GC2 IPU
AMD Epyc Rome
72-core Xeon Phi
Centriq 2400
AWS Graviton2
SPARC M7
32-core AMD Epyc
IBM z13 Storage Controller
Apple A12X Bionic
18-core Xeon Haswell-E5
HiSilicon Kirin 990 5G
Apple A13 (iPhone 11 Pro)
Xbox One main SoC
AMD Ryzen 7 3700X
61-core Xeon Phi
12-core POWER8
HiSilicon Kirin 710
8-core Xeon Nehalem-EX
10-core Core i7 Broadwell-E
Six-core Xeon 7400
Qualcomm Snapdragon 835
Dual-core Itanium 2
Dual-core + GPU Iris Core i7 Broadwell-U
Quad-core + GPU GT2 Core i7 Skylake K
Pentium D Presler
POWER6
Quad-core + GPU Core i7 Haswell
Apple A7 (dual-core ARM64 "mobile SoC")
Itanium 2 with 9 MB cache
Core i7 (Quad)
AMD K10 quad-core 2M L3
Itanium 2 Madison 6M
Core 2 Duo Wolfdale
Pentium D Smithfield
Core 2 Duo Conroe
Itanium 2 McKinley
Cell
Core 2 Duo Wolfdale 3M
Pentium 4 Prescott-2M
Core 2 Duo Allendale
Pentium 4 Cedar Mill
AMD K8
Pentium 4 Prescott
Pentium 4 Northwood
Barton
Atom
Pentium 4 Willamette
Pentium III Tualatin
Pentium II Mobile Dixon
ARM Cortex-A9
AMD K7
Pentium III Coppermine
AMD K6-III
AMD K6
Pentium III Katmai
Pentium II Deschutes
Pentium Pro
Pentium II Klamath
Pentium
AMD K5
SA-110
Intel 80486
R4000
TI Explorer's 32-bit Lisp machine chip
ARM700
Intel 80386
Intel i960
ARM 3
Motorola 68020
DEC WRL MultiTitan
Intel 80286
ARM 9TDMI
Motorola 68000
Intel 80186
Intel 8086
Intel 8088
ARM 2
ARM 1
ARM 6
WDC 65C816
Novix NC4016
Motorola 6809
TMS 1000
Zilog Z80
WDC 65C02
RCA 1802
Intel 8085
Intel 8080
Intel 8008
MOS Technology 6502
Motorola 6800
Intel 4004
1970
1972
1974
1976
1978
1980
1982
1984
1986
1988
1990
1992
1994
1996
1998
2000
2002
2004
2006
2008
2010
2012
2014
2016
2018
2020
Year in which the microchip was first introduced
Data source: Wikipedia (wikipedia.org/wiki/Transistor_count)
OurWorldinData.org – Research and data to make progress against the world's largest problems.
Licensed under CC-BY by the authors Hannah Ritchie and Max Roser.

Litografía para la fabricación de chips



https://www.gallagherseals.com/blog/semiconductor-manufacturing-process/

Las fábricas (Foundries)

- *TSMC* en Taiwan
- *Intel* en Estados Unidos y otros países
- *Samsung* en Corea del Sur
- *Global Foundries* en Alemania y Estados Unidos (Nueva York)
- *Hynix*, *Toshiba*, *Micron* y otros: fabricantes para propósitos especiales como memorias RAM, flash, sensores, etc.
- *SMIC* en China

Desafíos

- La miniaturización de los transistores se ha ralentizado últimamente
- ¡El costo por transistor aumenta!
- El más barato se fabrica a 14 nanómetros con DUV
- Las fábricas cuestan miles de millones de dólares y aumentando
- A partir de ~ 7 nm se necesita EUV: monopolio de *ASML*