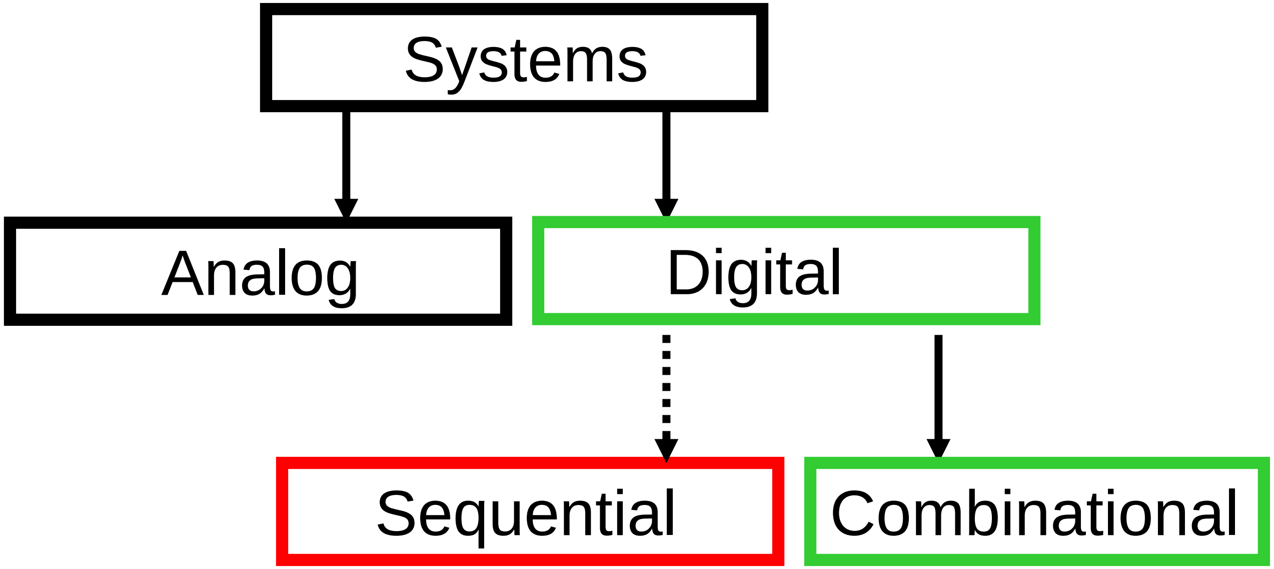
Sequential Logic Circuits

REGISTERS and RAM (Memory)



Systems

Analog

Digital

Sequential

Combinational

Combinational Logic Circuits

- The logic circuits [adders, decoders, encoders and multiplexers …] that we used or designed **up to now** are referred as “combinational logic circuits”

- In the combinational logic, time is not involved

- Output = *f* (current inputs)

Logic circuits with **no** memory = Memoryless

Memory

- In computer systems, memory is also a major component.
- The need to store prior information (data) is a must …
- Computers store data in a variety of memories (RAM, Flash, ROM, ….)
- Logic circuits that store data dynamically are referred as “Sequential Logic Circuits”.

Sequential Logic Circuits

- Time is involved …
- The output(s) depends on **past as well as present inputs.**
- Output = *f* (current inputs, past inputs, past outputs)
- The output(s) are known as **states.**

Logic circuits **with** memory

Sequential Logic Circuits

- The basic component of a sequential logic circuit is a logic device that is referred as **Latch/ Flip-Flop**
- Latches and Flip-Flops store a single bit of data until the stored value is overwritten.

Latches, Flip Flops

- Latch: [level sensitive storage element]
- Flip-Flop: [edge triggered storage element]

Types of Latches:

- S-R latch, J-K latch, D latch (= gated D latch)

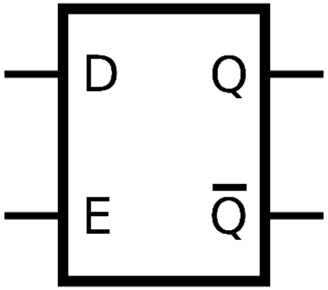
Types of Flip-Flops (FF):

- D-FF, D-FF with enable, Scan-FF, JK-FF, T-FF.

D-Flip Flop (D-FF)

- A logic device that stores one bit of binary data.
- The value of the stored bit is "Flipped" back and forth between the two binary states (1 and 0).
- Note that the D type Flip-Flop is the easiest basic sequential device that can be "understood" and used in design.
- D = Data of Delay

D-FF **(Truth Table)**

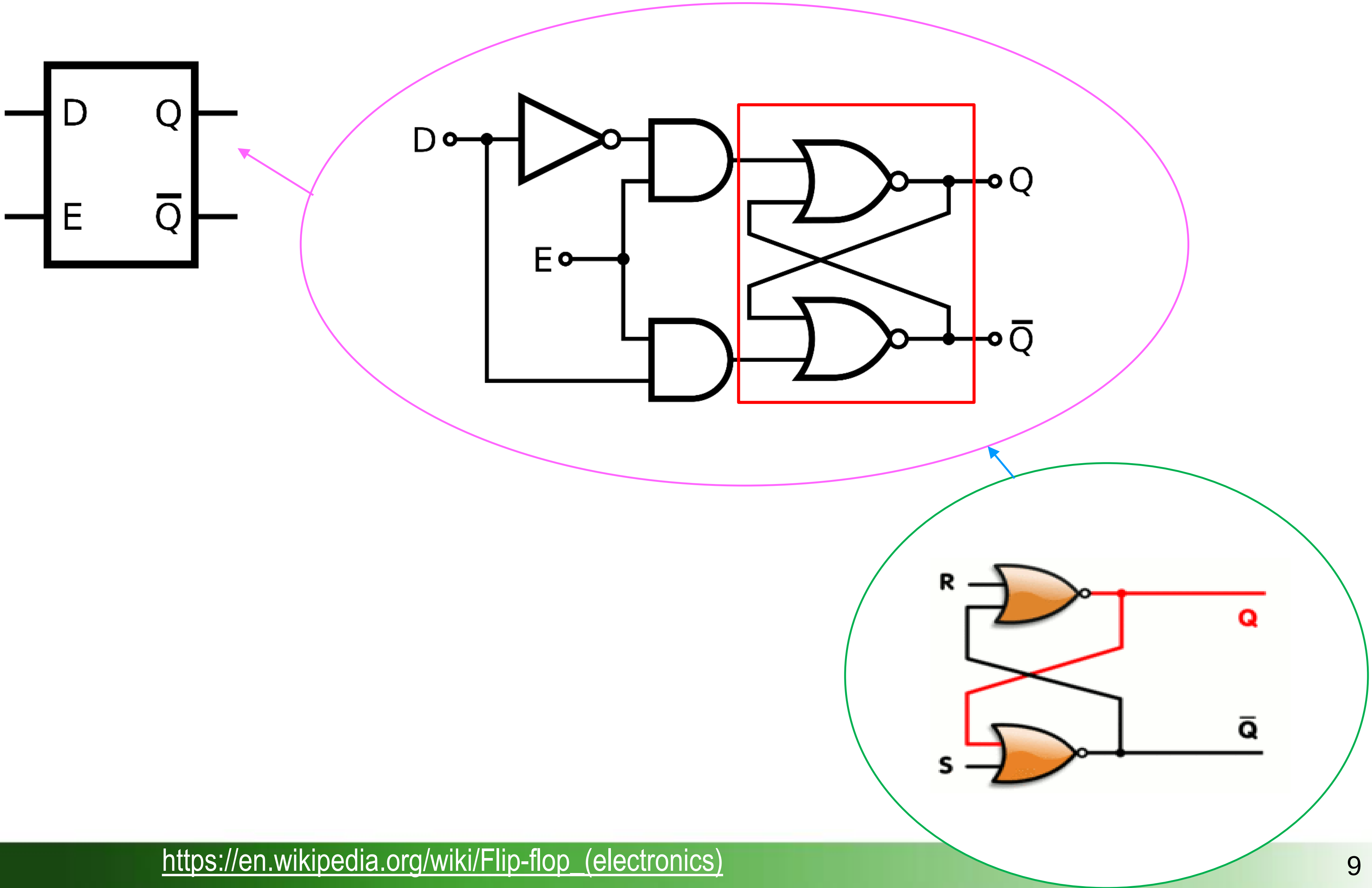


D	Q	State
0	0	reset
1	1	set

Q = D

D = Input
Q = Output
E = Enable or Clock (CLK)

Inside a D-FF …



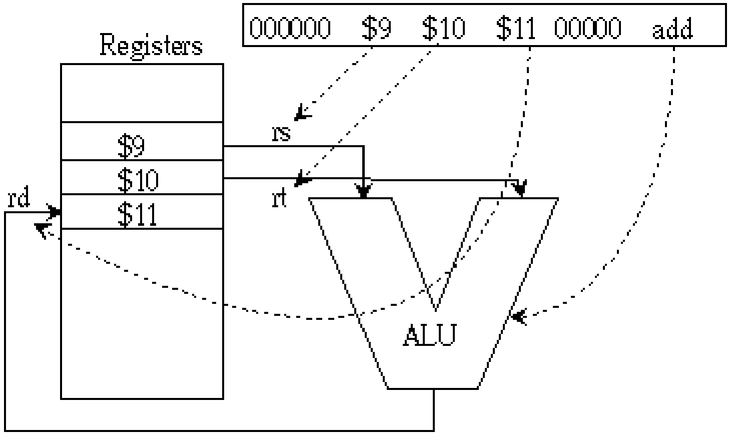
https://en.wikipedia.org/wiki/Flip-flop_(electronics)

Flip Flops are “Tools”

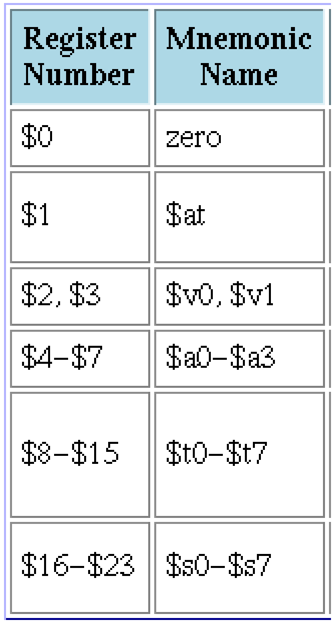
- The Flip-Flops are the “Tools” to (understand) design …

  - Registers
  - Digital (Binary) Counters
  - RAM

Registers



MIPS

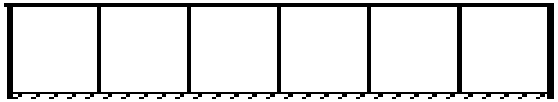


Register Number	Mnemonic Name
$0	zero
$1	$at
$2, $3	$v0, $v1
$4–$7	$a0–$a3
$8–$15	$t0–$t7
$16–$23	$s0–$s7

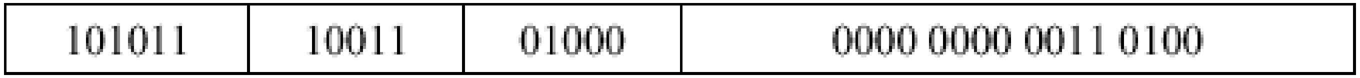
Register

Dynamic logic elements capable of storing an array of bits.

6-bit Register



32-bit Register

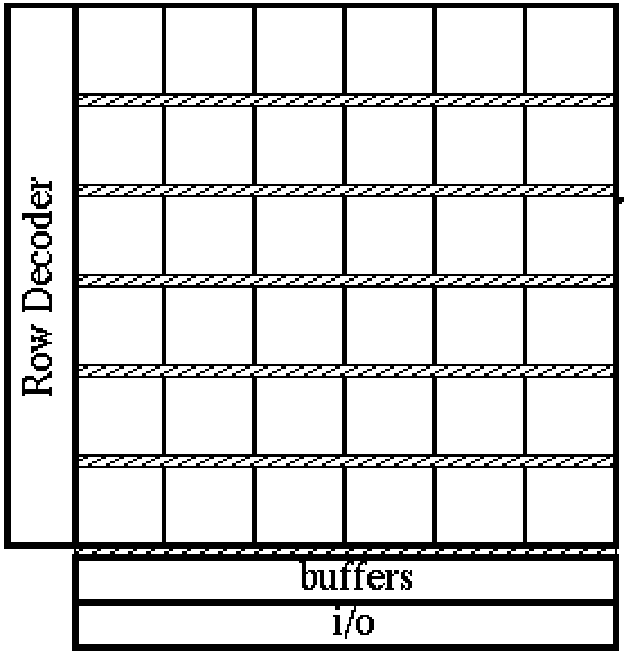


101011	10011	01000	0000 0000 0011 0100

Memory system

Dynamic logic elements of storing a matrix of bits.

36-bit Memory



Register

- A temporary storage logic component
- It is completely transparent to user, and even to programmer (except Assembly Language)
- A n-bit register is capable of storing n-bit numbers in the range:

1	1	1	…	1	1	High
0	0	0	…	0	0	Low

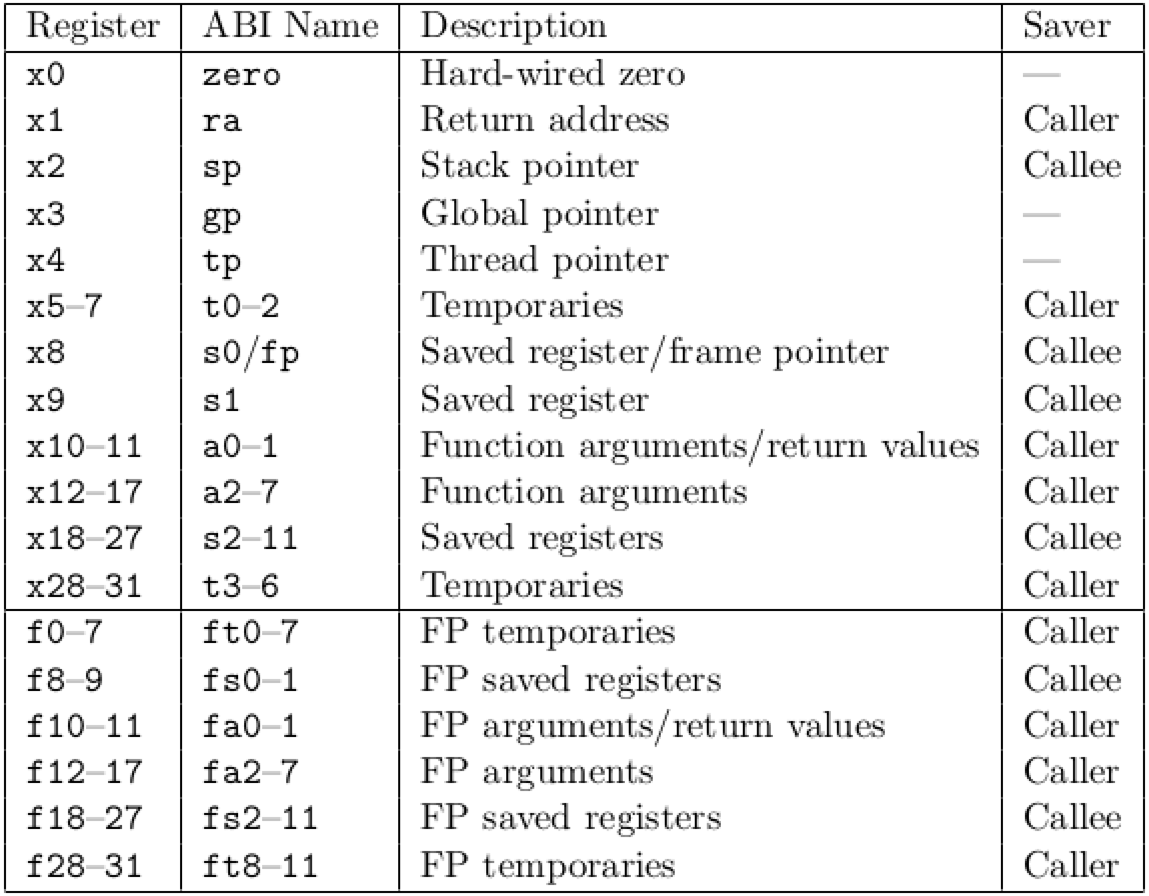
Therefore

Latch (Flip-Flop): Can store **a single bit**

Register: Can store an **array of bits**

Memory: Can store a **matrix of bits**

Registers: RISC-V CPUs



Register	ABI Name	Description	Saver
x0	zero	Hard-wired zero	—
x1	ra	Return address	Caller
x2	sp	Stack pointer	Callee
x3	gp	Global pointer	—
x4	tp	Thread pointer	—
x5–7	t0–2	Temporaries	Caller
x8	s0/fp	Saved register/frame pointer	Callee
x9	s1	Saved register	Callee
x10–11	a0–1	Function arguments/return values	Caller
x12–17	a2–7	Function arguments	Caller
x18–27	s2–11	Saved registers	Callee
x28–31	t3–6	Temporaries	Caller
f0–7	ft0–7	FP temporaries	Caller
f8–9	fs0–1	FP saved registers	Callee
f10–11	fa0–1	FP arguments/return values	Caller
f12–17	fa2–7	FP arguments	Caller
f18–27	fs2–11	FP saved registers	Callee
f28–31	ft8–11	FP temporaries	Caller

https://www.arm.com/products/processors/instruction-set-architectures/index.php

Types of Registers

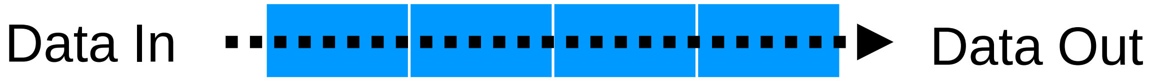
- Processor registers
- Data registers
- Address registers
- Conditional registers
- Special registers
- Control and status registers (PC & IR)
- Floating point registers
- …..
- Shift registers.

Shift Register

- A register that its contents are shifted …
- How?

4-bit Register

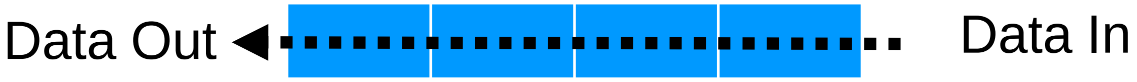
Shift right and left (4-bit register)



Data In

Data Out

Serial-in/Shift Right/Serial-out

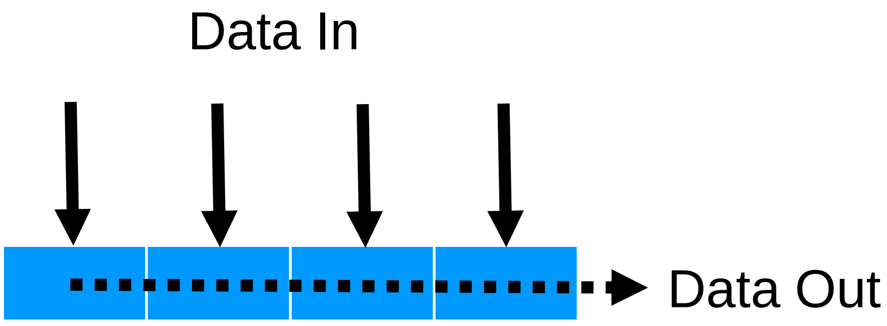


Data Out

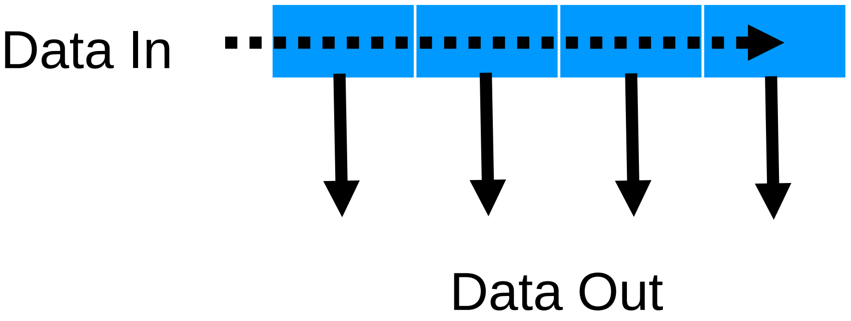
Data In

Serial-in/Shift Left/Serial-out

Parallel/Serial (4-bit register)

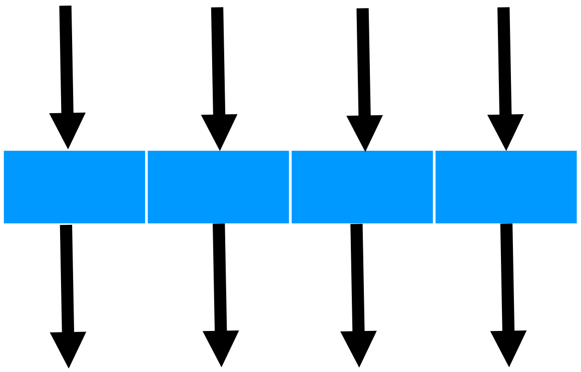


Data Parallel-in/ Data Serial-out



Data Serial-in/ Data Parallel-out

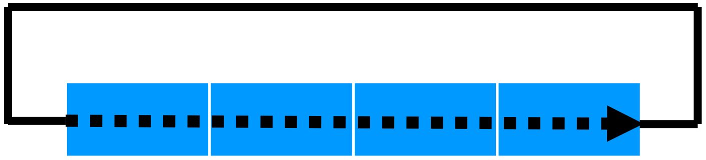
Buffer (Data Parallel-in/ Data Parallel-out)



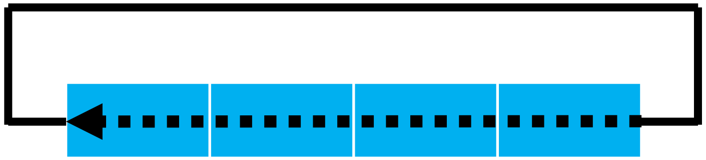
Data In

Data Out

Data movement



Rotation-right

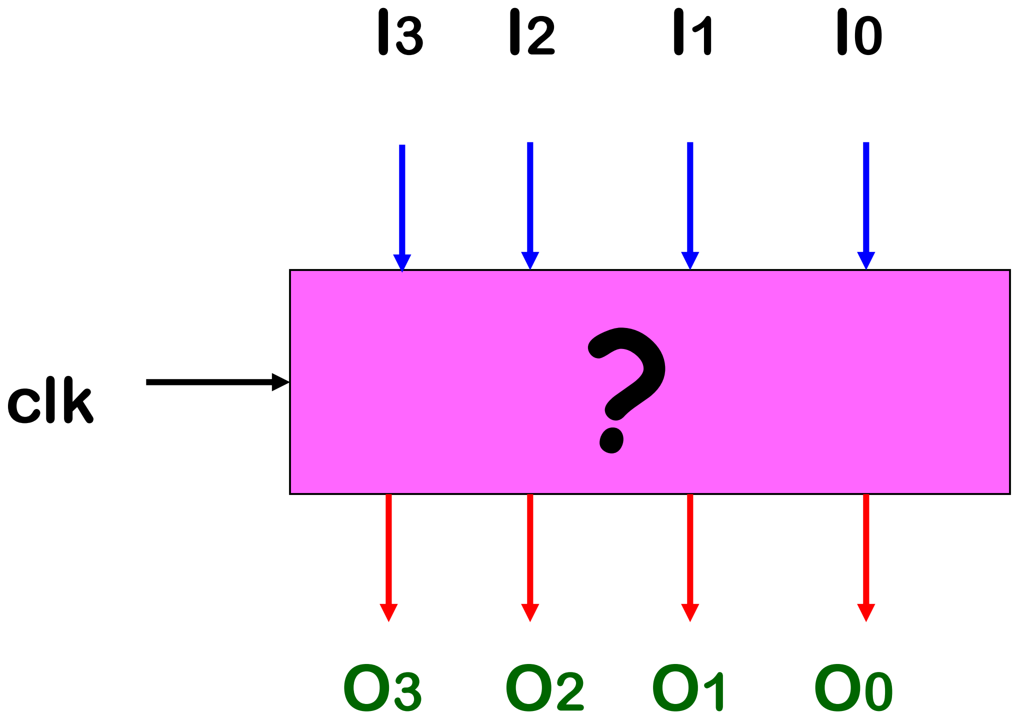


Rotation-left

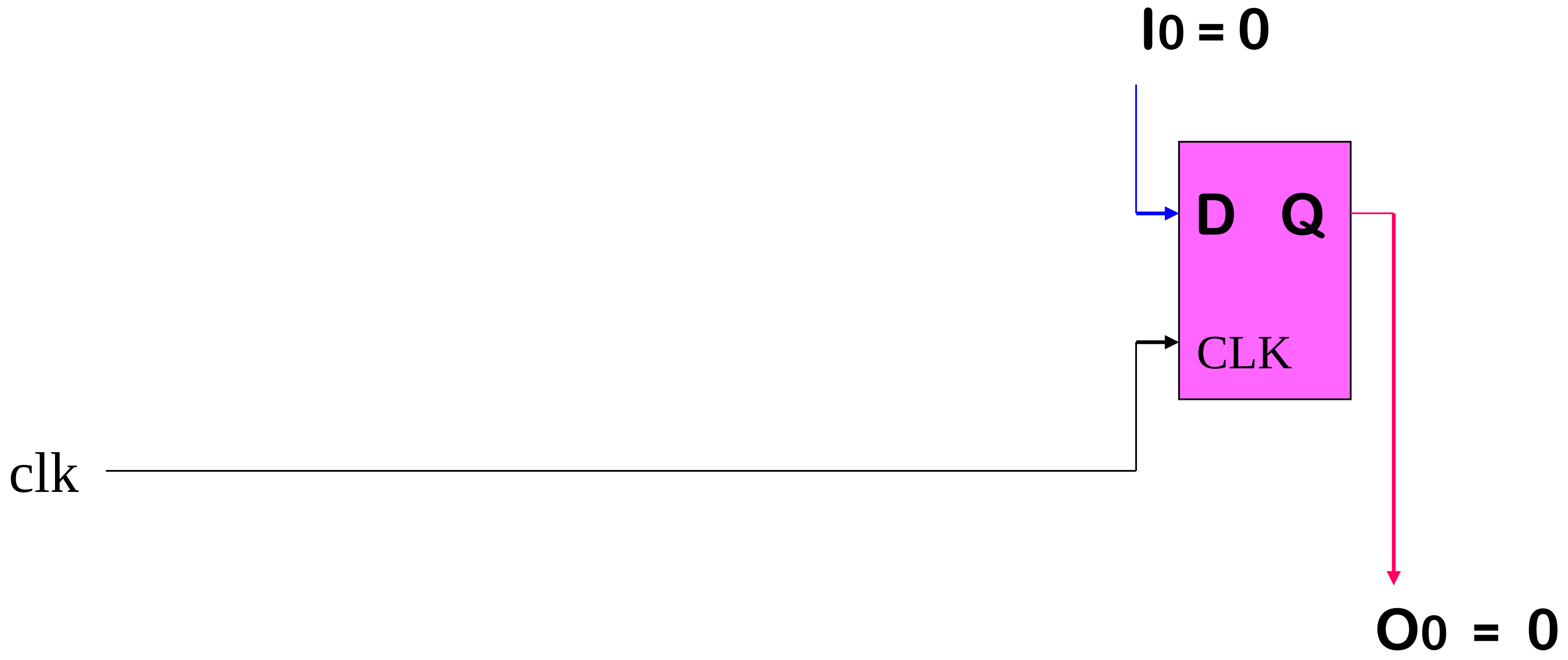
Registers (Examples)

1. Data Parallel-in / Data Parallel-out
2. Data Serial-in / Data Parallel-out
3. Data Serial-in / Data Serial-out

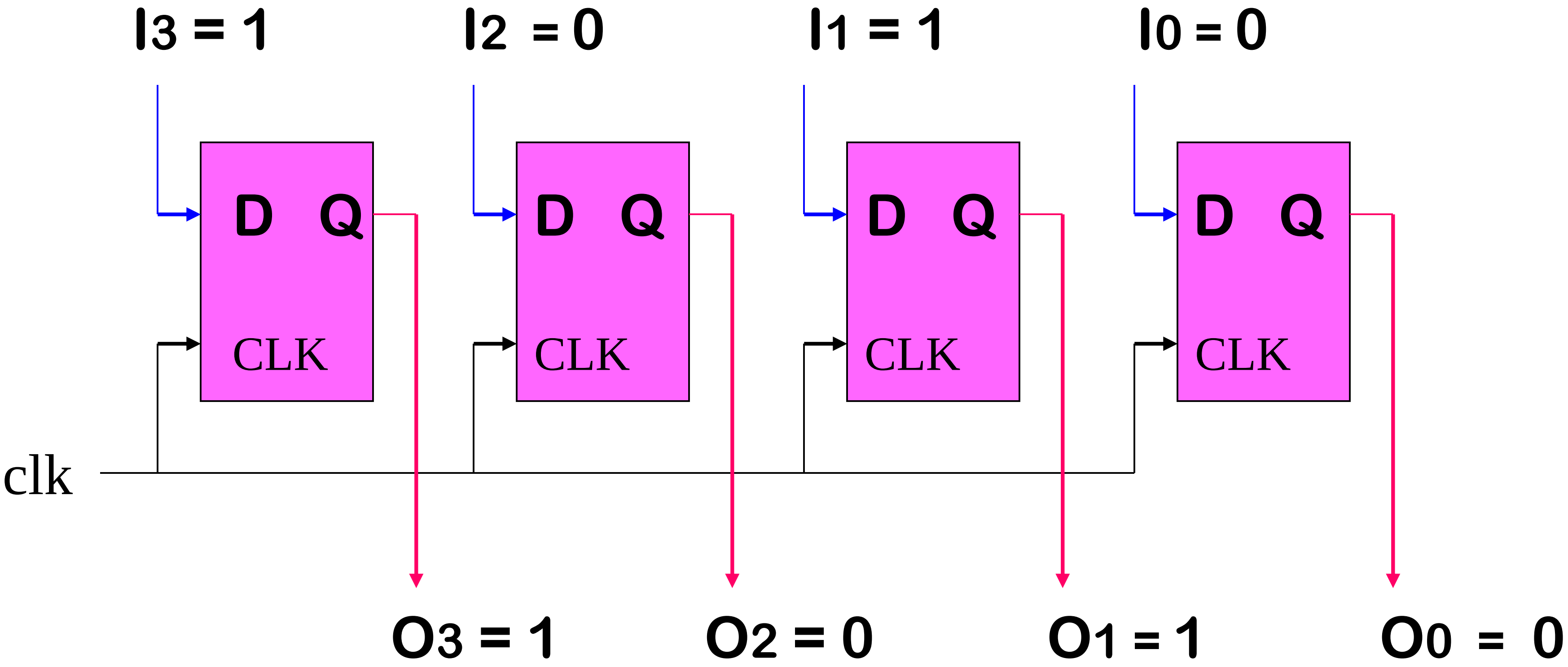
1) 4-bit Parallel-in Parallel-out Register



1) 4-bit Parallel-in Parallel-out Register

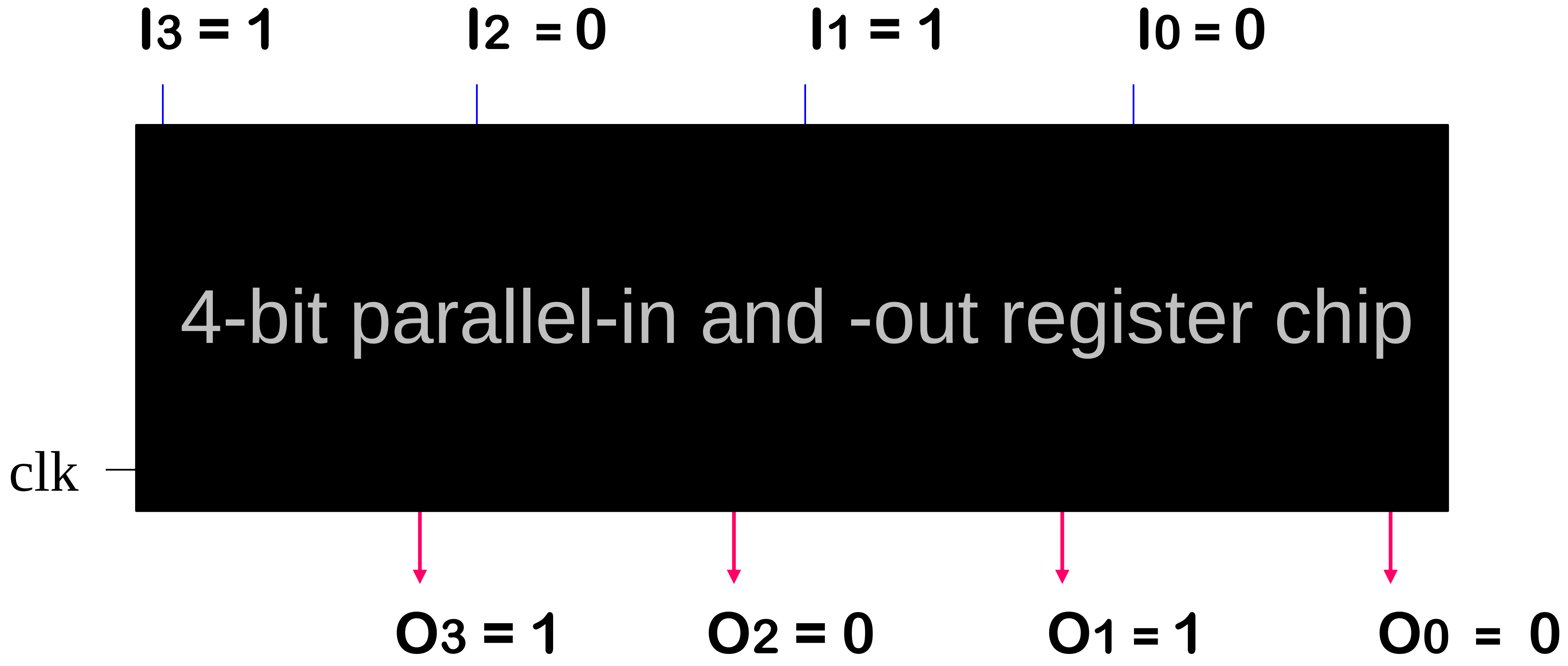


1) 4-bit Parallel-in Parallel-out Register

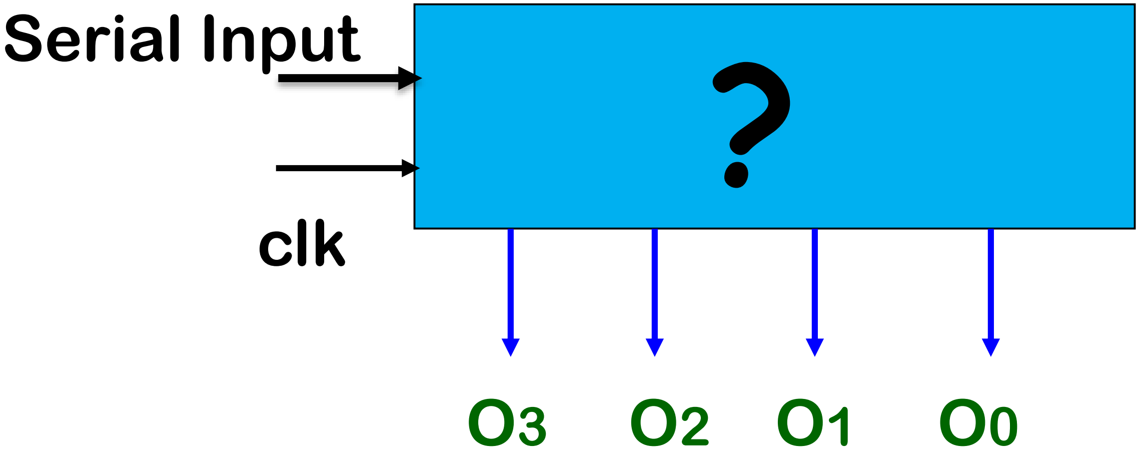


The data stored in the register are available at all time at the output lines

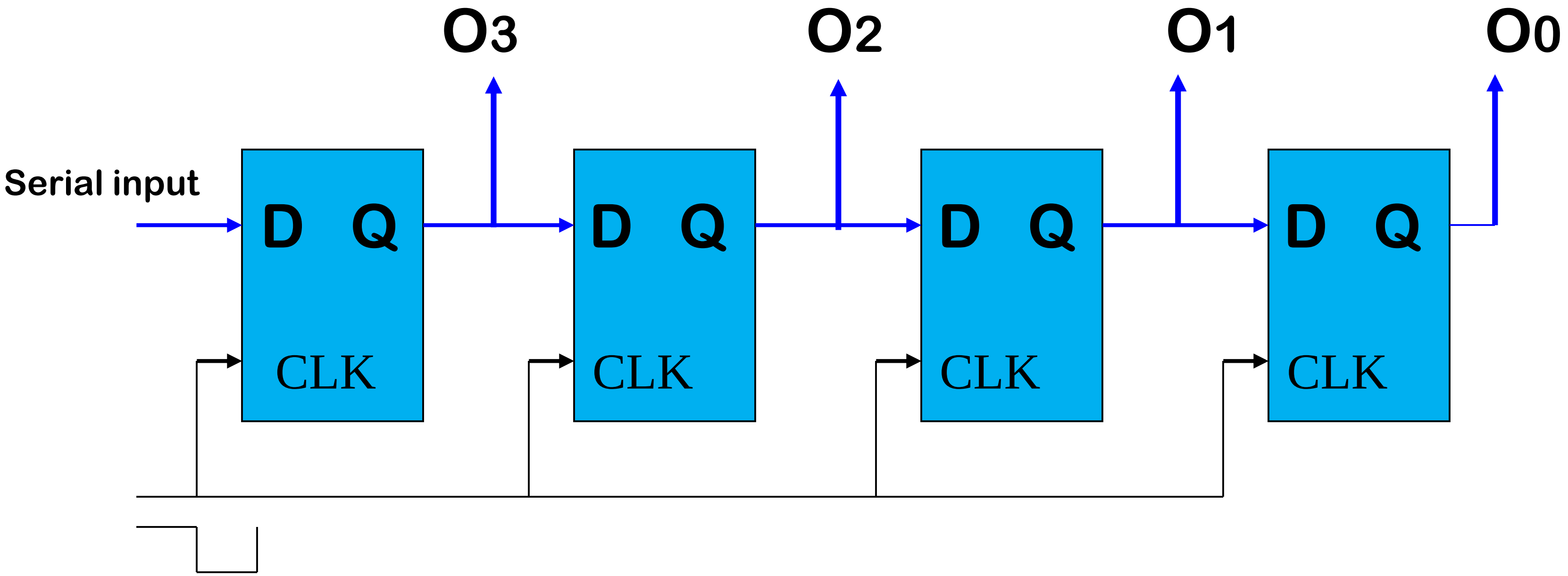
1) 4-bit Parallel-in Parallel-out Register



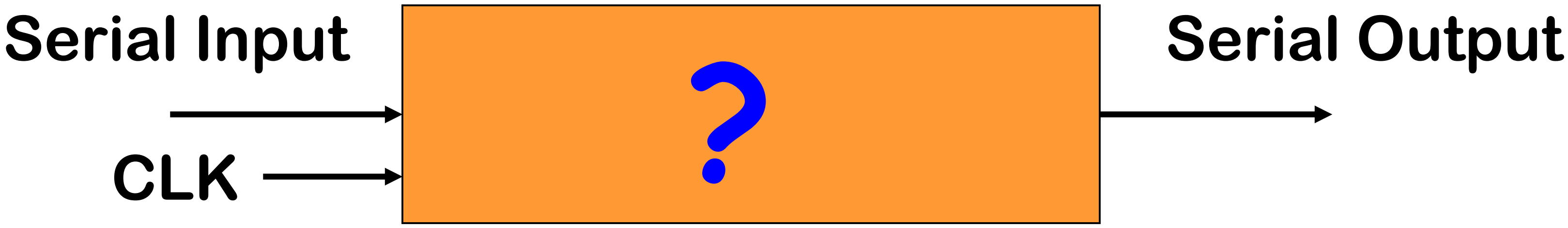
2) Serial-in parallel-out shift register



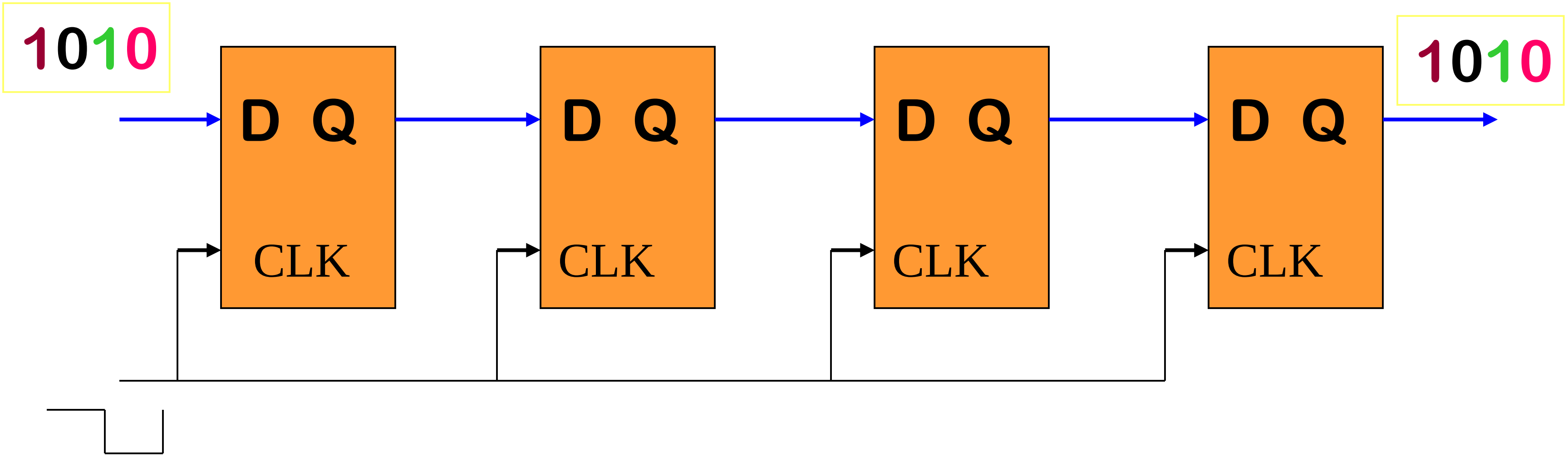
2) 4-bit serial-in parallel-out shift register

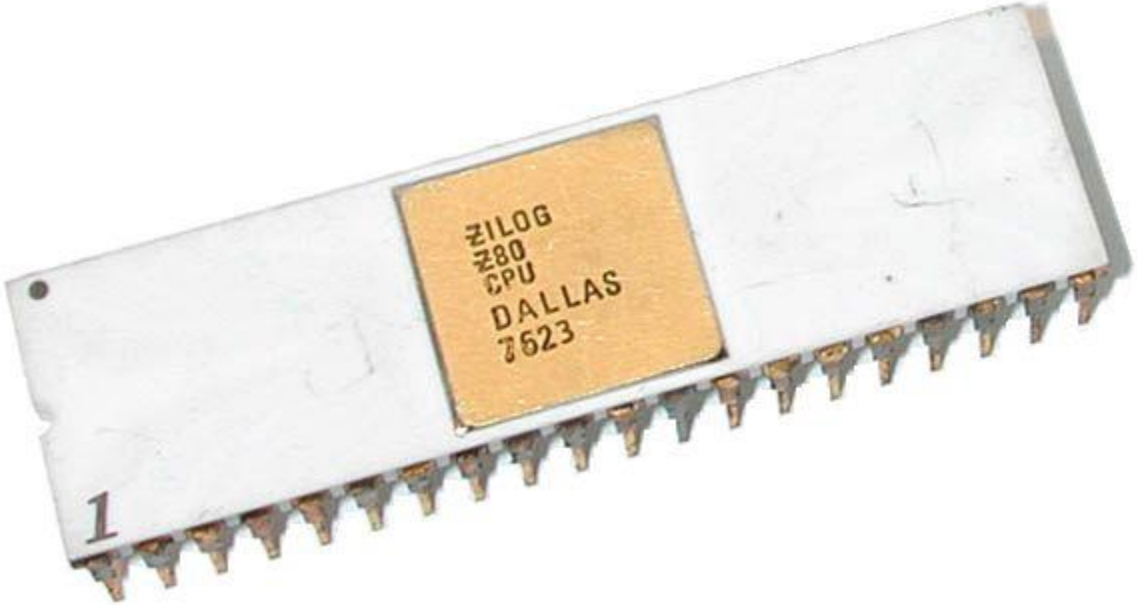


3) Serial-in Serial-out shift Register



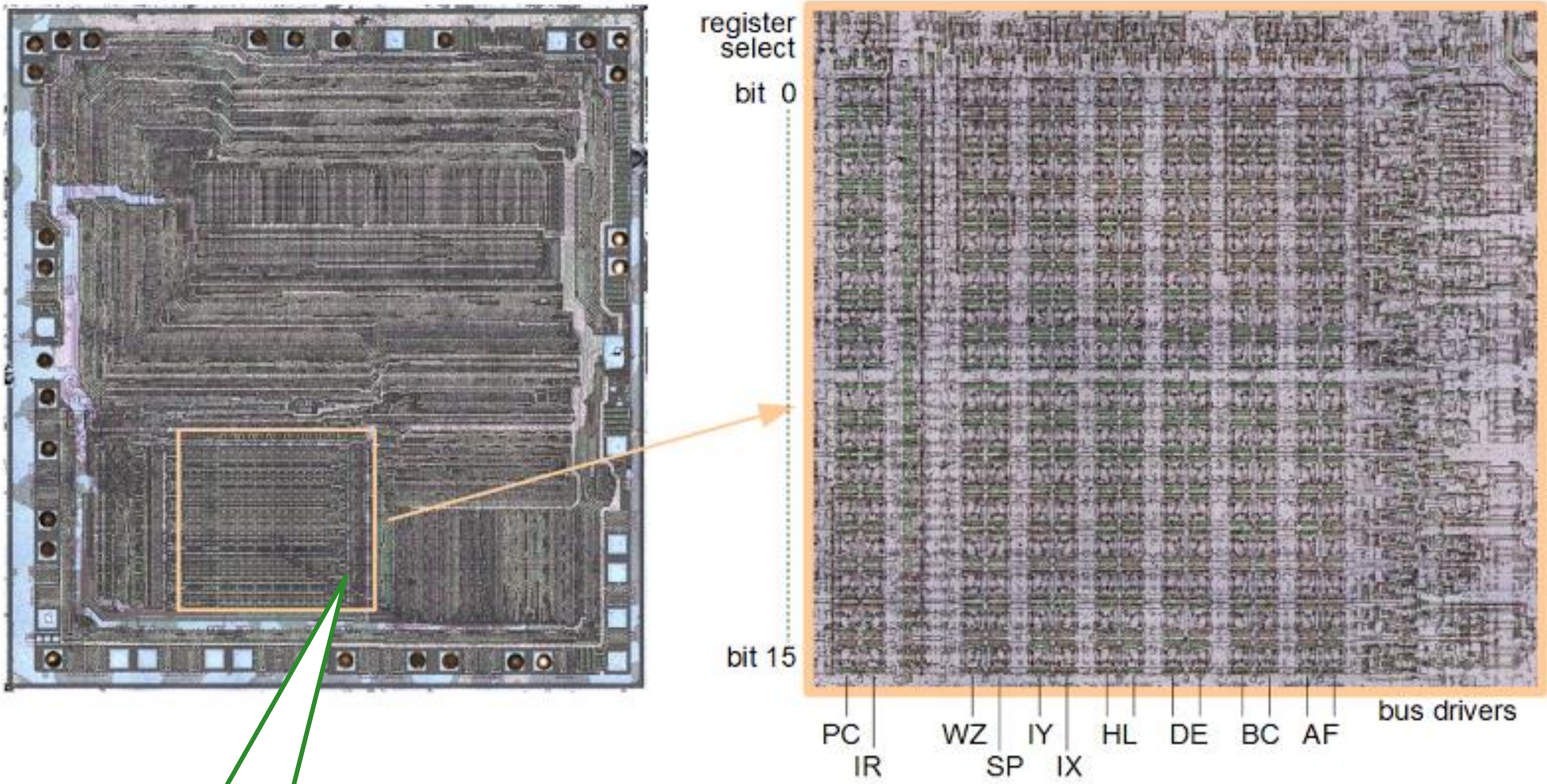
3) 4-Bit Serial-in Serial-out shift Register





Z80 CPU

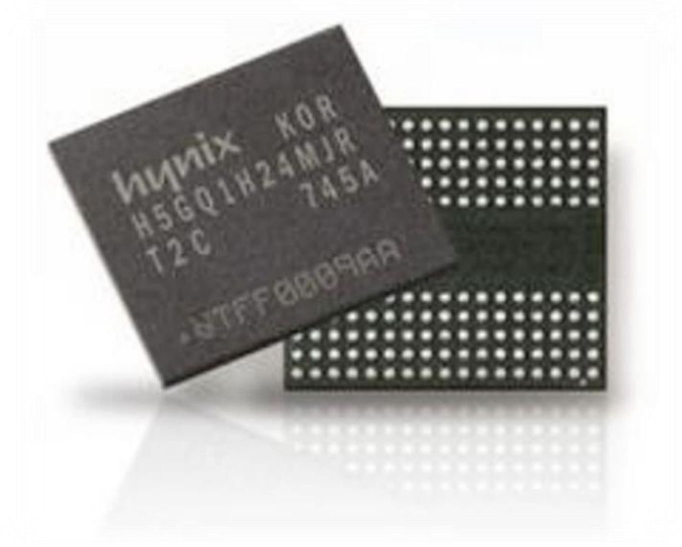
Die of the Z80 microprocessor (CPU)



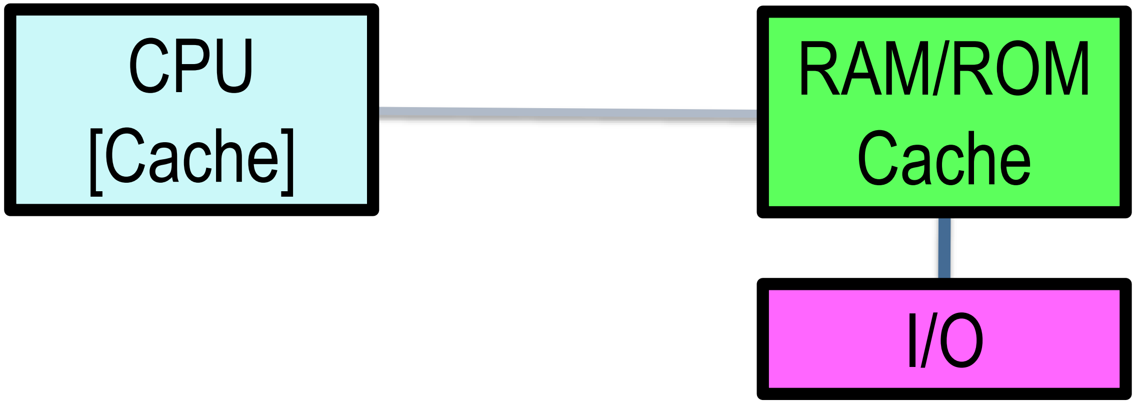
Register File

http://www.righto.com/2014/10/how-z80s-registers-are-implemented-down.html

Memory



ROM…RAM…Cache (Memories)



Types of Memory

- ROM
- RAM
- Cache (?)
- Flash (?)

Types of Memory

- ROM (read-permanent)
- RAM (read/write-temporary)
- Cache (very fast RAM-temporary)
- Flash (new fast RAM-permanent)

Types of Memory

- ROM
- RAM
- Cache
- Flash
- Magnetic disc (Hard, Floppy, Zip, …)
- Tape
- Optical (CD-ROM, CD-R, CD-RW, WORM, DVD-ROM)

ROM++

- PROM (Programmable ROM)
- EPROM (Erasable Programmable ROM)
- EEPROM (Electrically EPROM)

RAM

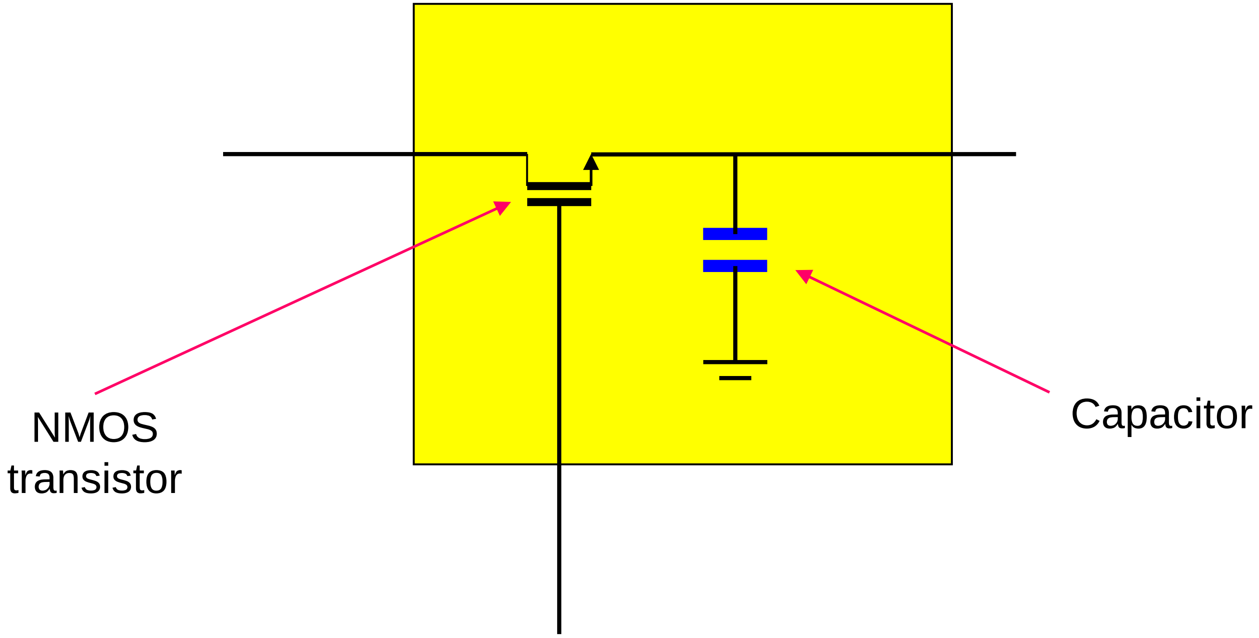
- Dynamic → D-RAM
- Static → S-RAM
- Video → V-RAM

- How do use D-RAM and S-RAM ?

D-RAM

- Good for large memory systems
- Low cost: Requires 1 transistor for storing
- Requires refreshing (recharging), thousands of times per second, of the stored information (electric charge in a capacitor) as it loses the data.

D-RAM basic memory cell



NMOS (N-type metal-oxide-semiconductor)

Refreshing

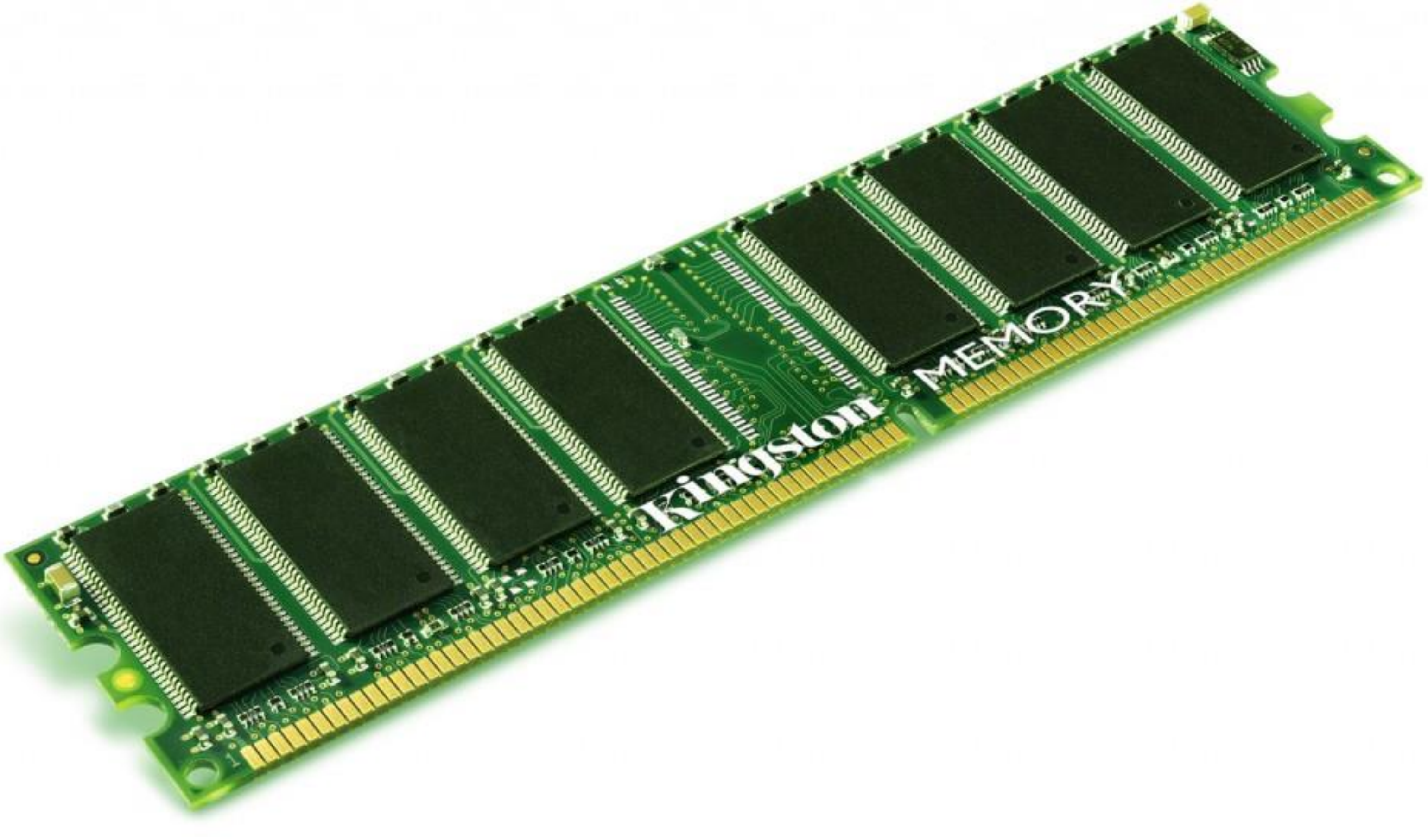
- Typical refresh rates: 7.8, 15.6, …,128 µsec
- Refreshing is done…
  - Counters, Voltage pulses
  - “Self-refresh mode”
  - [Memory chip controller (MCC)]

D-RAM is the …

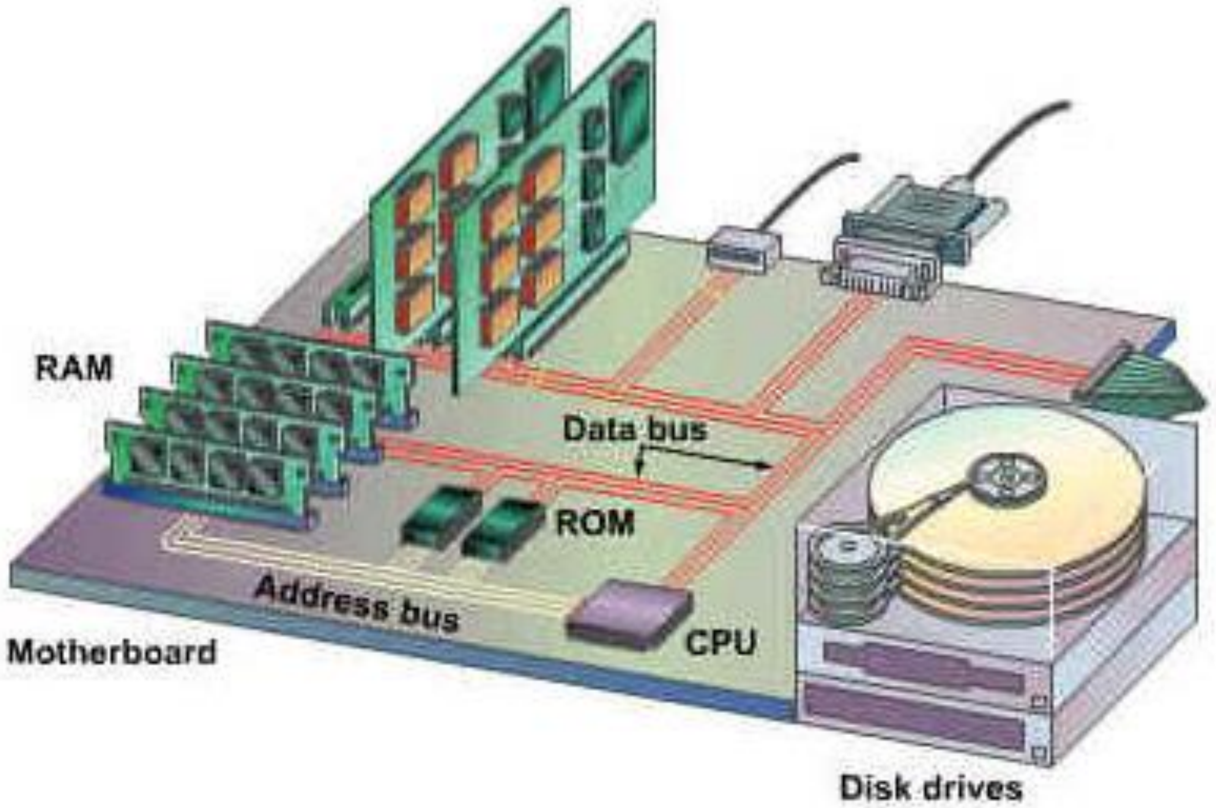
- ?...? Memory

D-RAM is the …

- Main (system) Memory



RAM



In general the computer memory…

- The memory is build in powers of 2.

- Example:
  - 1Kb = 2^{10} = 1,024 bits
  - 64 Kb = 2^{16} bits

Main operations of the memory

- Write = Store information to RAM
- Read = Load information from RAM

S-RAM

- Stores binary information without the necessity of periodic refreshing
- High cost: Requires 4 or 6 transistors
- Good for small memory systems

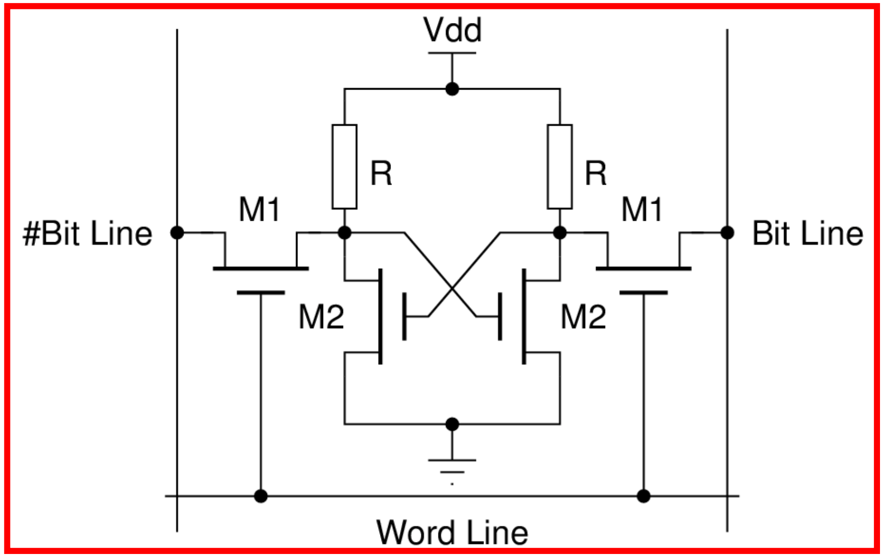
S-RAM is the …

- ?...? Memory

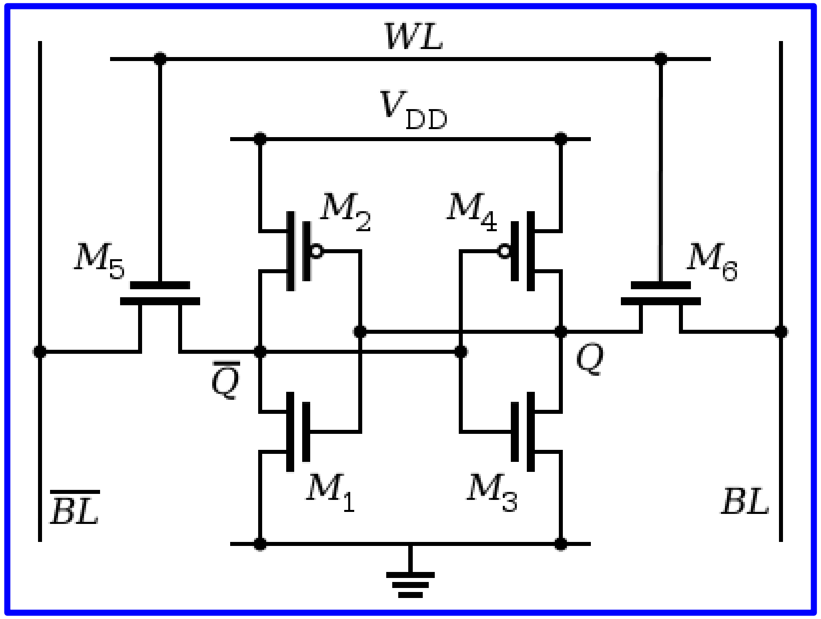
S-RAM is the …

- Cache Memory

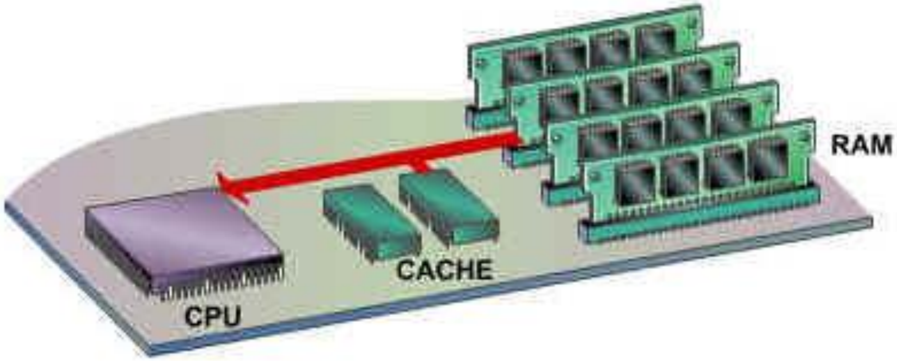
4 to 6 Transistors



"SRAM Cell (6 Transistors)" by Inductive load - Own work. Licensed under Public Domain via Wikimedia Commons - http://commons.wikimedia.org/wiki/File:SRAM_Cell_(6_Transistors).svg#/media/File:SRAM_Cell_(6_Transistors).svg



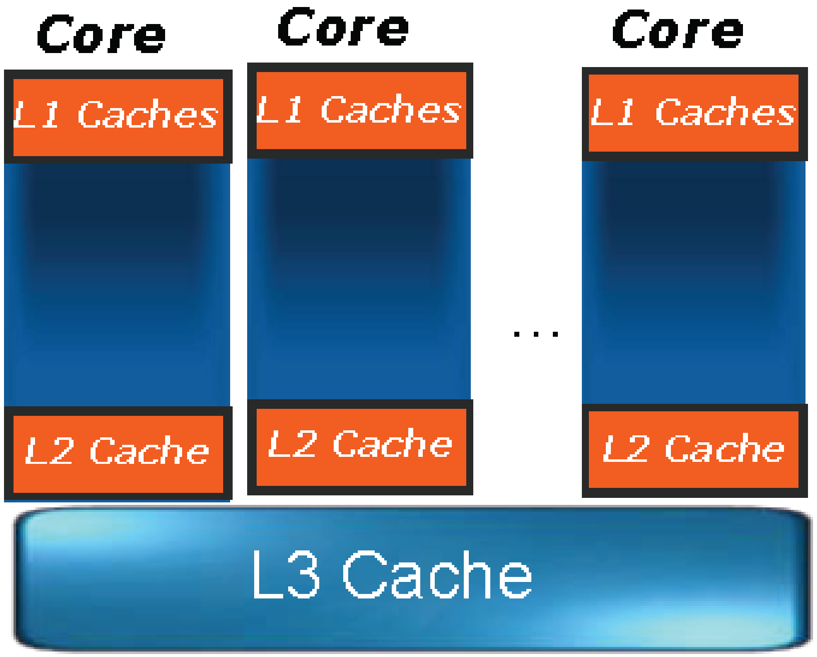
S-RAM is the …Cache Memory



http://www.alf.sd83.bc.ca/courses/It12/using_it/processor_speed.htm

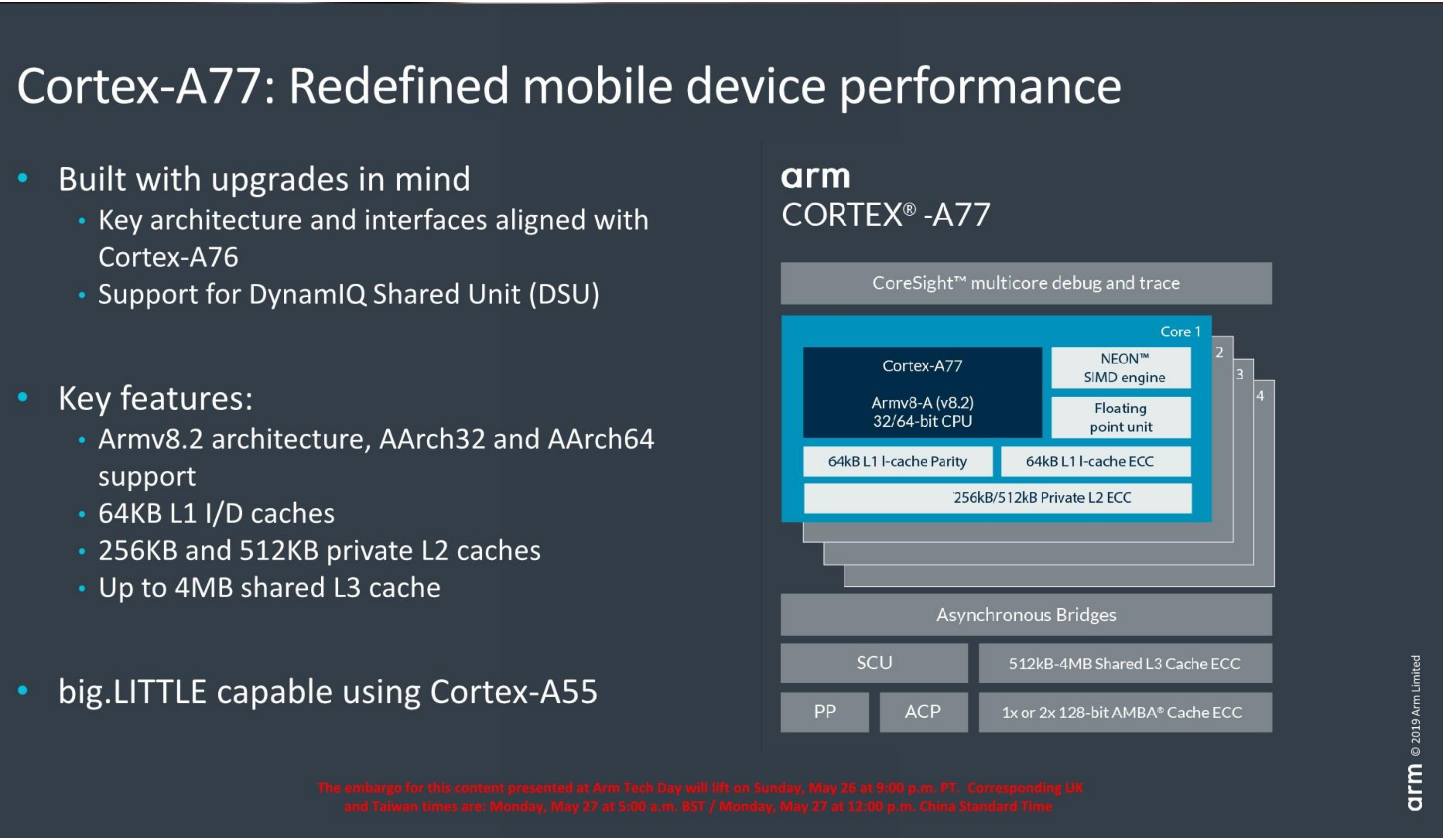
Cache Memory; Today ... 2 or 3 levels

- On chip (L1) cache memory
- On/off chip (L2) cache memory
- On/off chip (L3) cache memory



Cache memory = Shared memory between the available cores

Today CORTEX –A77 (2020)



Cortex-A77: Redefined mobile device performance

- Built with upgrades in mind
  - Key architecture and interfaces aligned with Cortex-A76
  - Support for DynamIQ Shared Unit (DSU)
- Key features:
  - Armv8.2 architecture, AArch32 and AArch64 support
  - 64KB L1 I/D caches
  - 256KB and 512KB private L2 caches
  - Up to 4MB shared L3 cache
- big.LITTLE capable using Cortex-A55

arm
CORTEX® -A77

CoreSight™ multicore debug and trace

Core 1
2
3
4

Cortex-A77
Armv8-A (v8.2)
32/64-bit CPU

NEON™
SIMD engine

Floating
point unit

64kB L1 I-cache Parity

64kB L1 I-cache ECC

256kB/512kB Private L2 ECC

Asynchronous Bridges

SCU

512kB-4MB Shared L3 Cache ECC

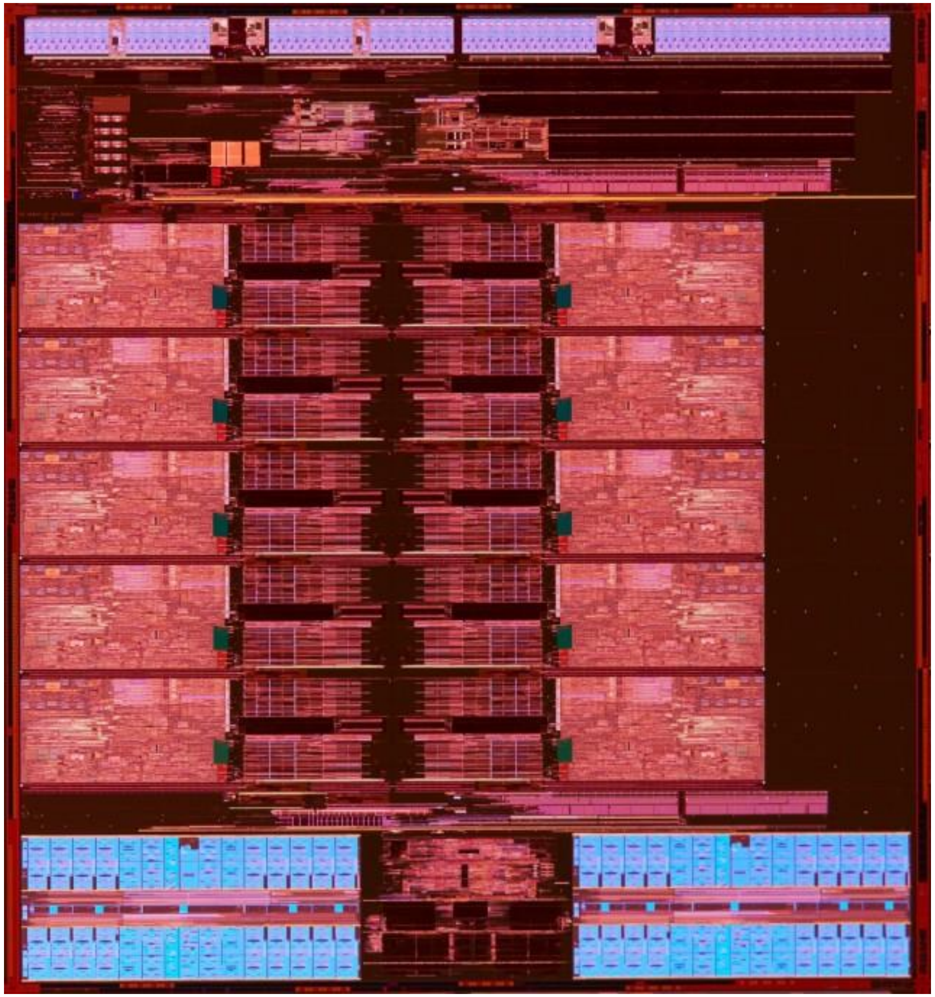
PP

ACP

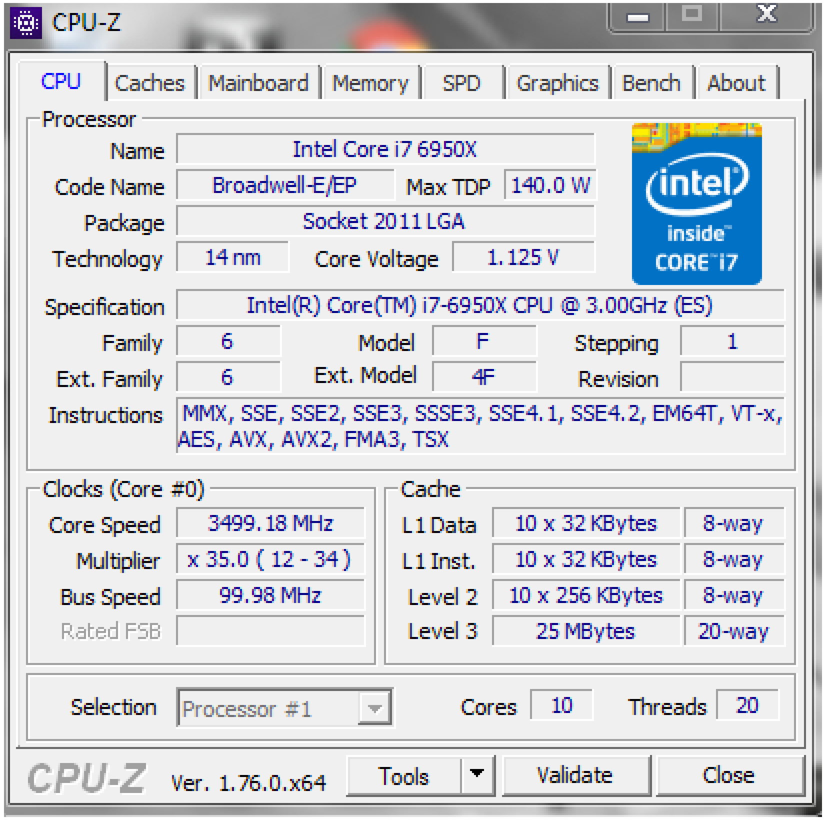
1x or 2x 128-bit AMBA® Cache ECC

The embargo for this content presented at Arm Tech Day will lift on Sunday, May 26 at 9:00 p.m. PT. Corresponding UK and Taiwan times are: Monday, May 27 at 5:00 a.m. BST / Monday, May 27 at 12:00 p.m. China Standard Time

10-Core CPU with Cache Memory (i7-6950X)



10-Core CPU with Cache Memory (i7-6950X)



Processor

Name	Intel Core i7 6950X				
Code Name	Broadwell-E/EP	Max TDP	140.0 W		
Package	Socket 2011 LGA				
Technology	14 nm	Core Voltage	1.125 V		
Specification	Intel(R) Core(TM) i7-6950X CPU @ 3.00GHz (ES)				
Family	6	Model	F	Stepping	1
Ext. Family	6	Ext. Model	4F	Revision	
Instructions	MMX, SSE, SSE2, SSE3, SSSE3, SSE4.1, SSE4.2, EM64T, VT-x, AES, AVX, AVX2, FMA3, TSX				

Clocks (Core #0)

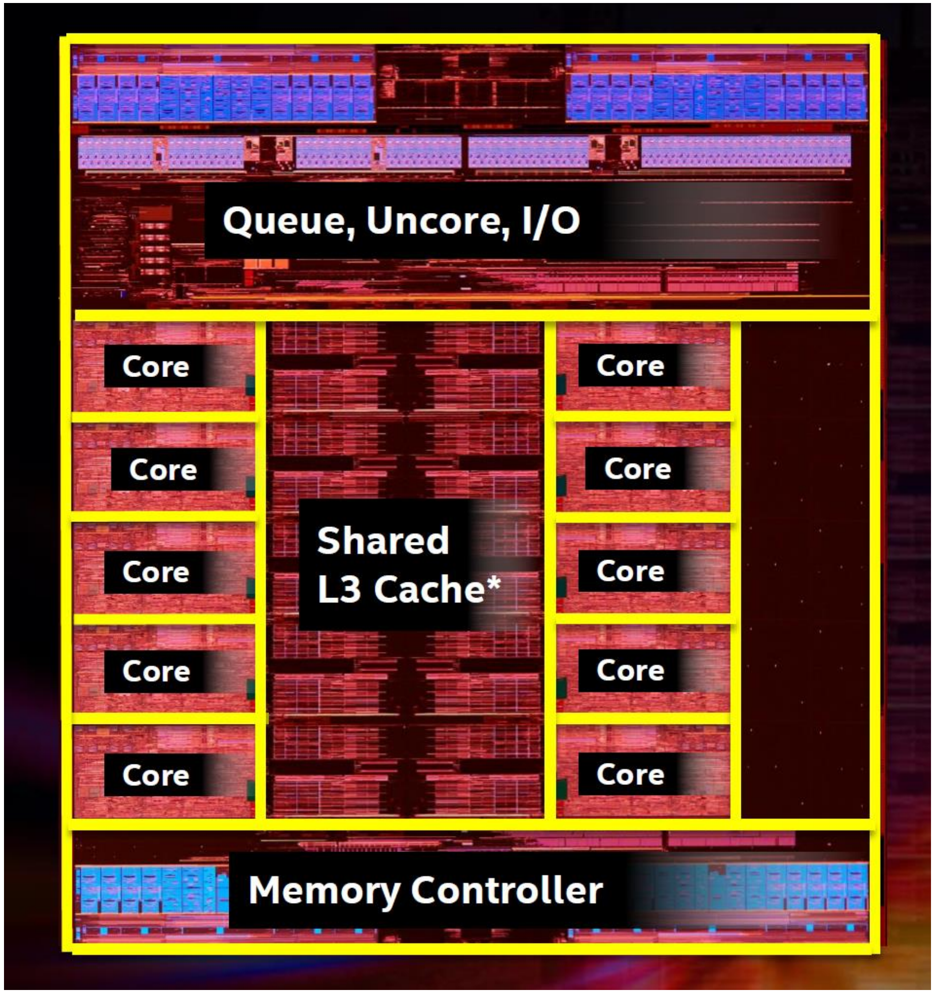
Core Speed	3499.18 MHz
Multiplier	x 35.0 (12 - 34)
Bus Speed	99.98 MHz
Rated FSB	

Cache

L1 Data	10 x 32 KBytes	8-way
L1 Inst.	10 x 32 KBytes	8-way
Level 2	10 x 256 KBytes	8-way
Level 3	25 MBytes	20-way

Selection	Processor #1	Cores	10	Threads	20

Ver. 1.76.0.x64



The Apple A7/Cyclone core

- CPU Codename--------------- Cyclone,
  ARM ISA------------------------- ARMv8-A(32/64),
  Issue Width---------------------- 6 micro-ops,
  Reorder Buffer Size----------- 192 micro-ops,
  Branch Mispredict Penalty---16 cycles (14 – 19),
  Integer ALUs-------------------- 4,
  Load/Store Units--------------- 2,
  Load Latency------------------- 4 Cycles,
  Branch Units-------------------- 2,
  Indirect Branch Units--------- 1,
  FP/NEON ALUs----------------3,
  L1 Cache-—---------------------- 64KB I$ + 64KB D$,
  L2 Cache------------------------ 1MB,
  L3 Cache------------------------ 4MB.

https://www.top500.org/

- Summit - IBM Power System AC922

- More than 2,414,592 cores

https://en.wikichip.org/wiki/supercomputers/summit

Oak Ridge National Laboratory's next High Performance Supercomputer.

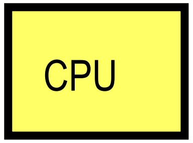
Typical values (2019)

- DRAM = up to ? GB
- SRAM = up to ? MB (level-3)

Typical PC values (2019)

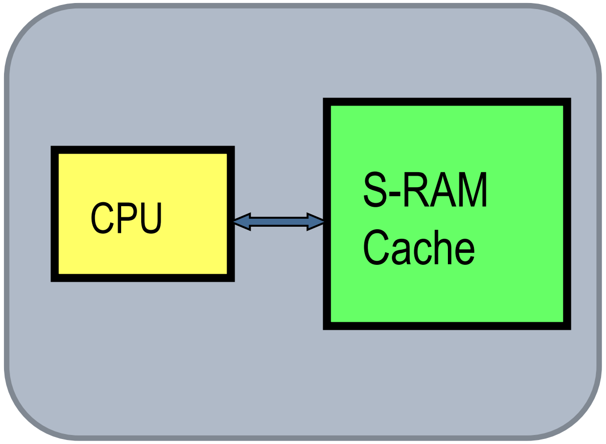
- DRAM = up to 8-128 GB
- SRAM = up to 8 MB (level-3).

CPU

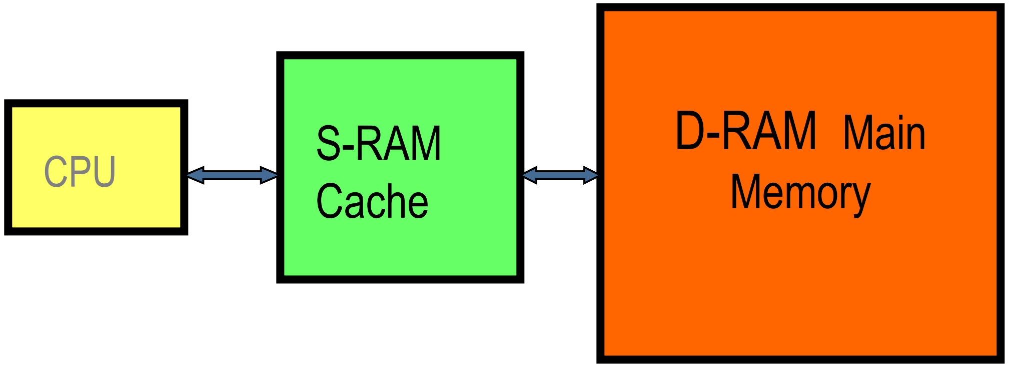


CPU

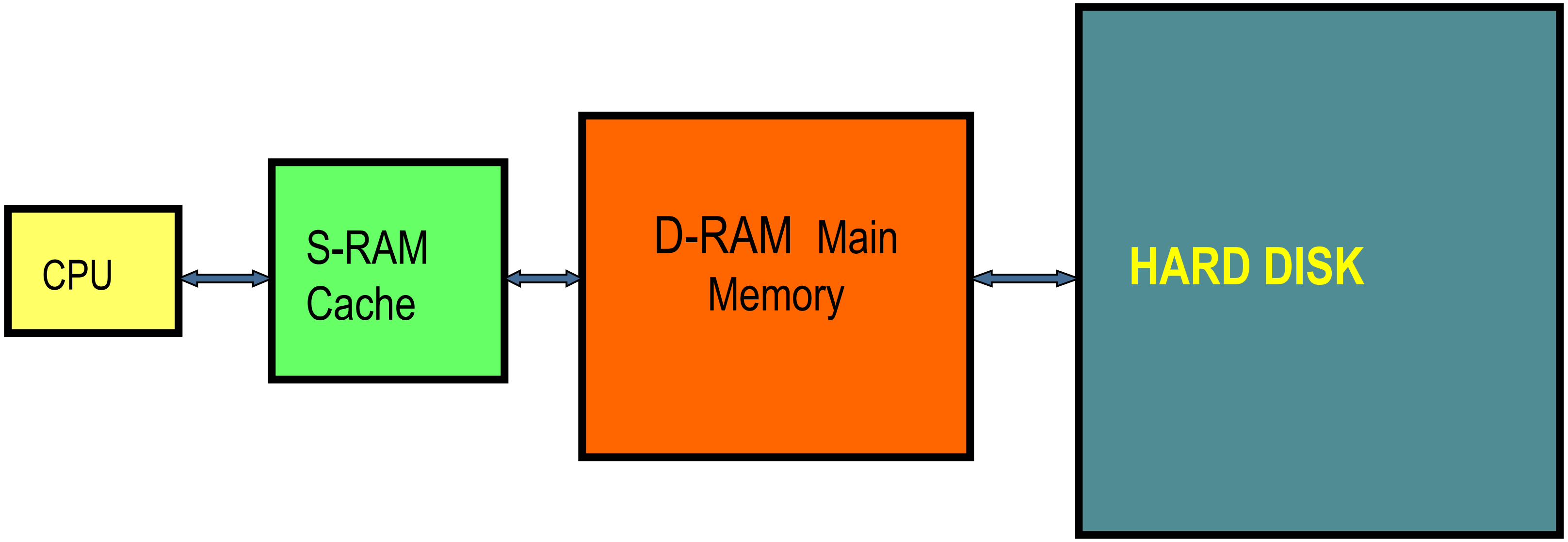
Cache Memory



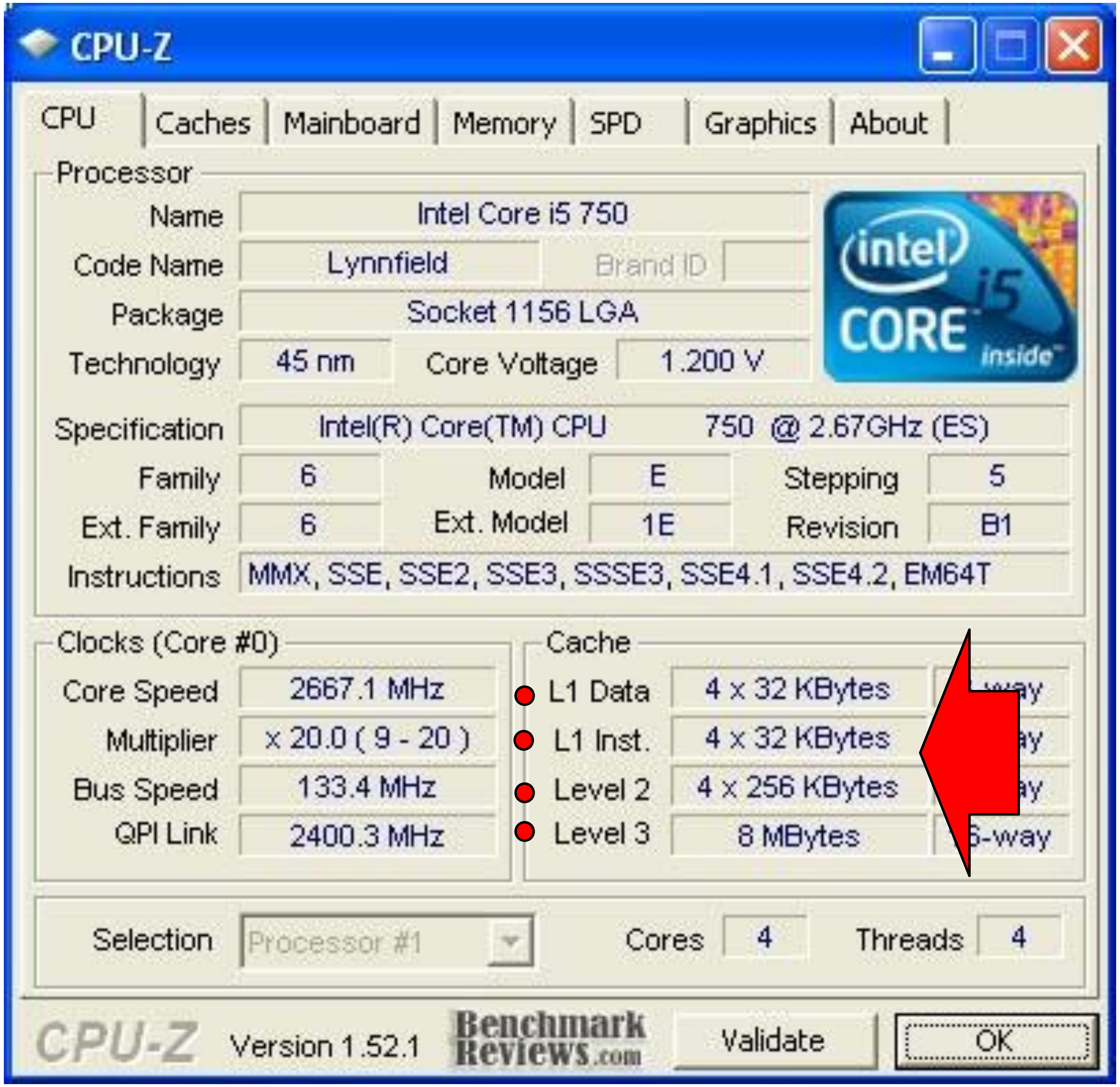
Main Memory



Hard disk



Three Levels of Cache Memory



Processor

Name: Intel Core i5 750
Code Name: Lynnfield
Brand ID:
Package: Socket 1156 LGA
Technology: 45 nm
Core Voltage: 1.200 V
Specification: Intel(R) Core(TM) CPU 750 @ 2.67GHz (ES)
Family: 6
Model: E
Stepping: 5
Ext. Family: 6
Ext. Model: 1E
Revision: B1
Instructions: MMX, SSE, SSE2, SSE3, SSSE3, SSE4.1, SSE4.2, EM64T

Clocks (Core #0)

Core Speed: 2667.1 MHz
Multiplier: x 20.0 (9 - 20)
Bus Speed: 133.4 MHz
QPI Link: 2400.3 MHz

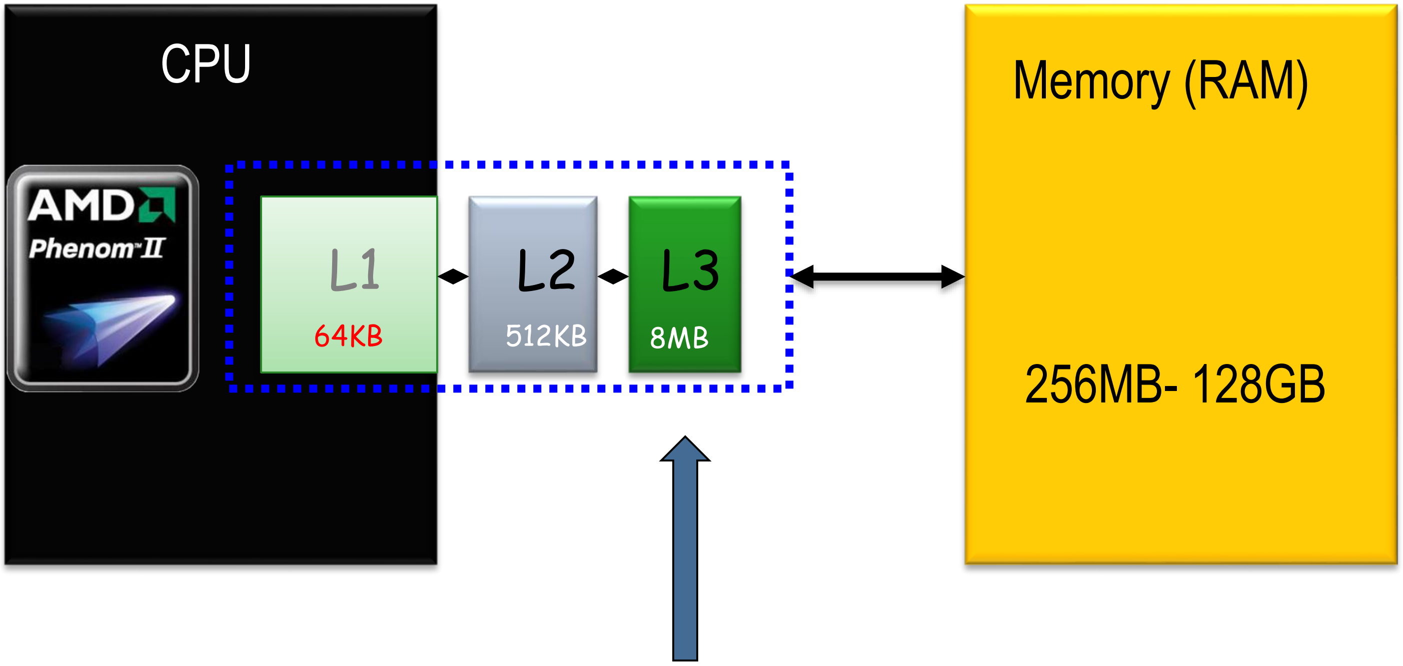
Cache

- L1 Data: 4 x 32 KBytes -way
- L1 Inst.: 4 x 32 KBytes ay
- Level 2: 4 x 256 KBytes ay
- Level 3: 8 MBytes 6-way

Selection: Processor #1
Cores: 4
Threads: 4

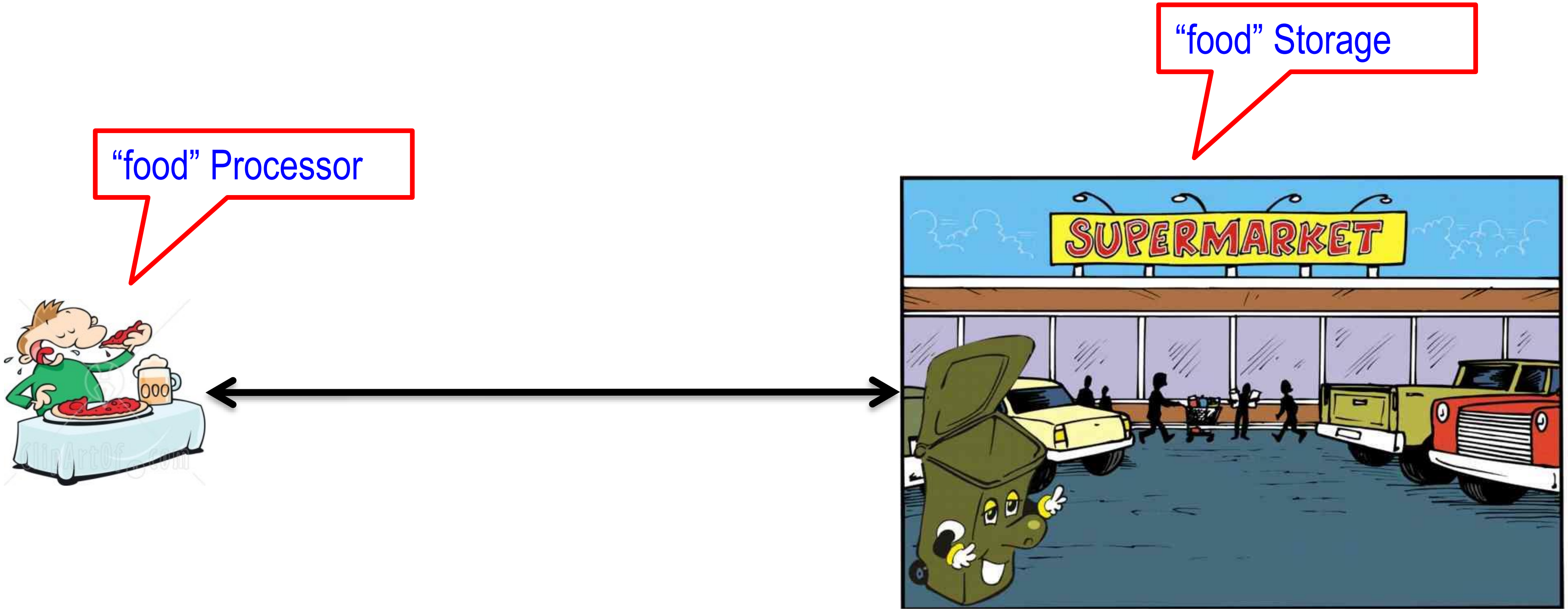
Version 1.52.1

Cache Memory ... RAM



Multicore – shared memory

CPU and System Memory



“food” Storage

“food” Processor

CPU

System memory

Cache Memory

Local “food” Storage

↔

↔

CPU

Cache memory

System memory

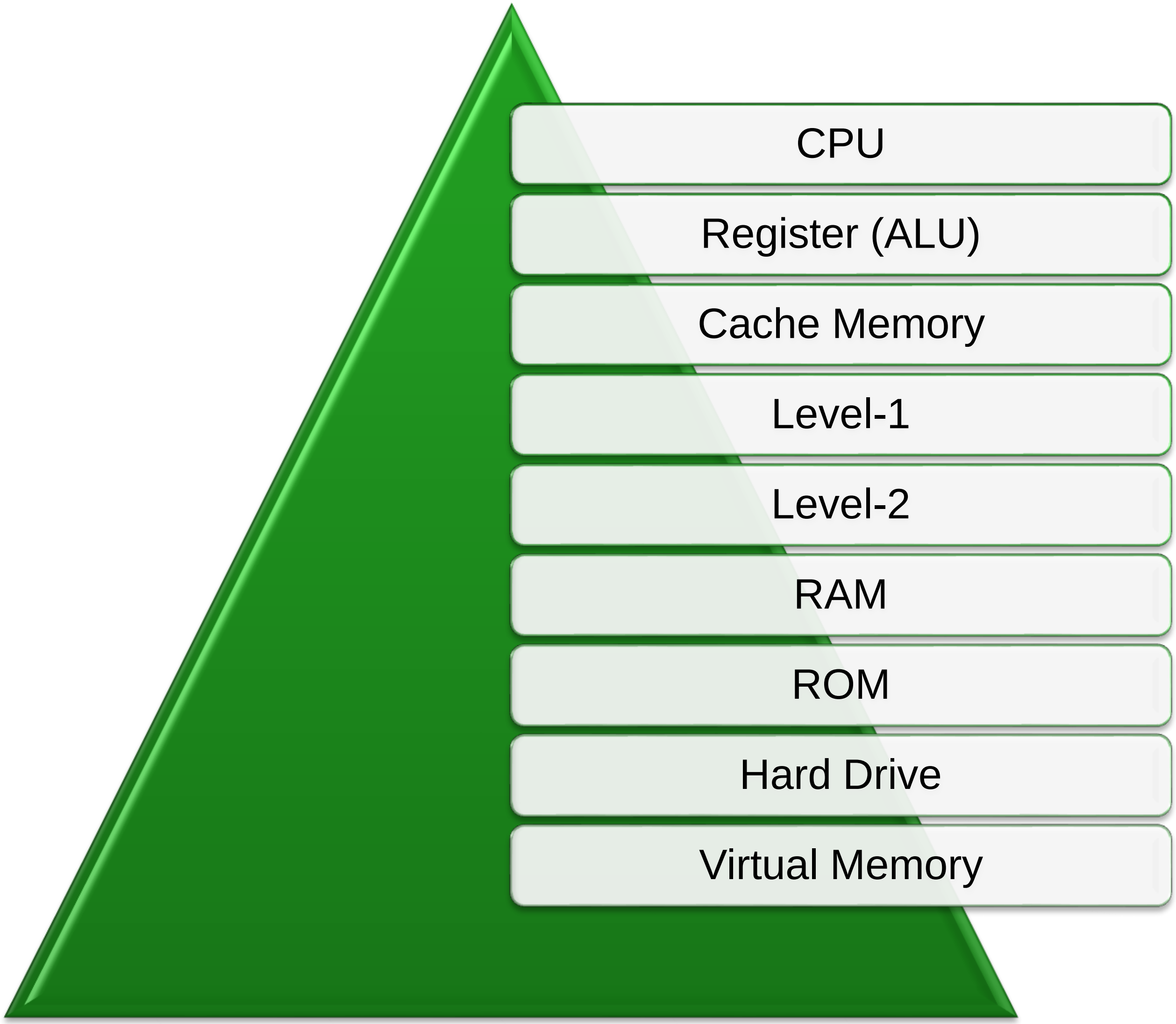
RAM

S-RAM (Cache)	D-RAM (Main)
Small & Fast	Slow & Large
Complex	Simple
No refresh	Refresh needed
More expensive	Less expensive

Specialized RAM is the VRAM

- VRAM = Video RAM
- Dual ported Dynamic RAM
- Can write while someone else reads.

The Memory Hierarchy



CPU

Register (ALU)

Cache Memory

Level-1

Level-2

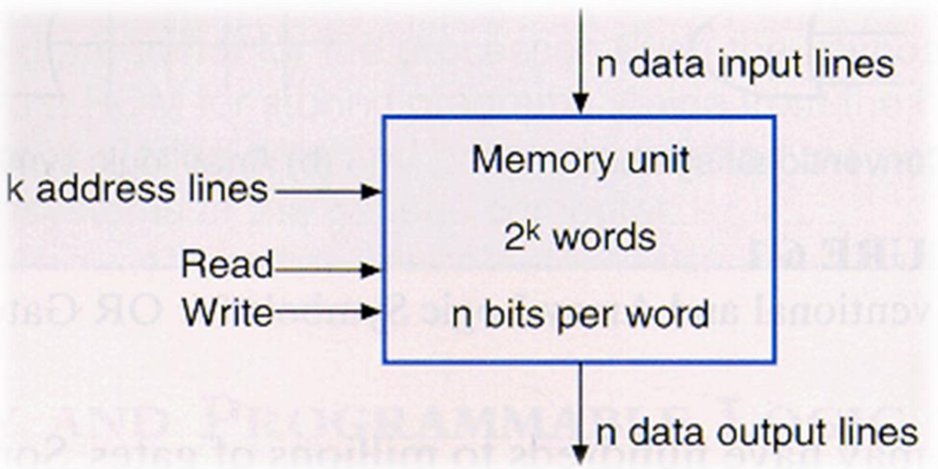
RAM

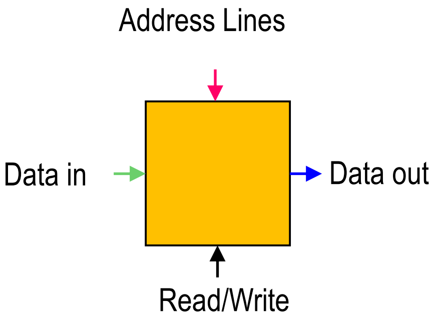
ROM

Hard Drive

Virtual Memory

Memory cell



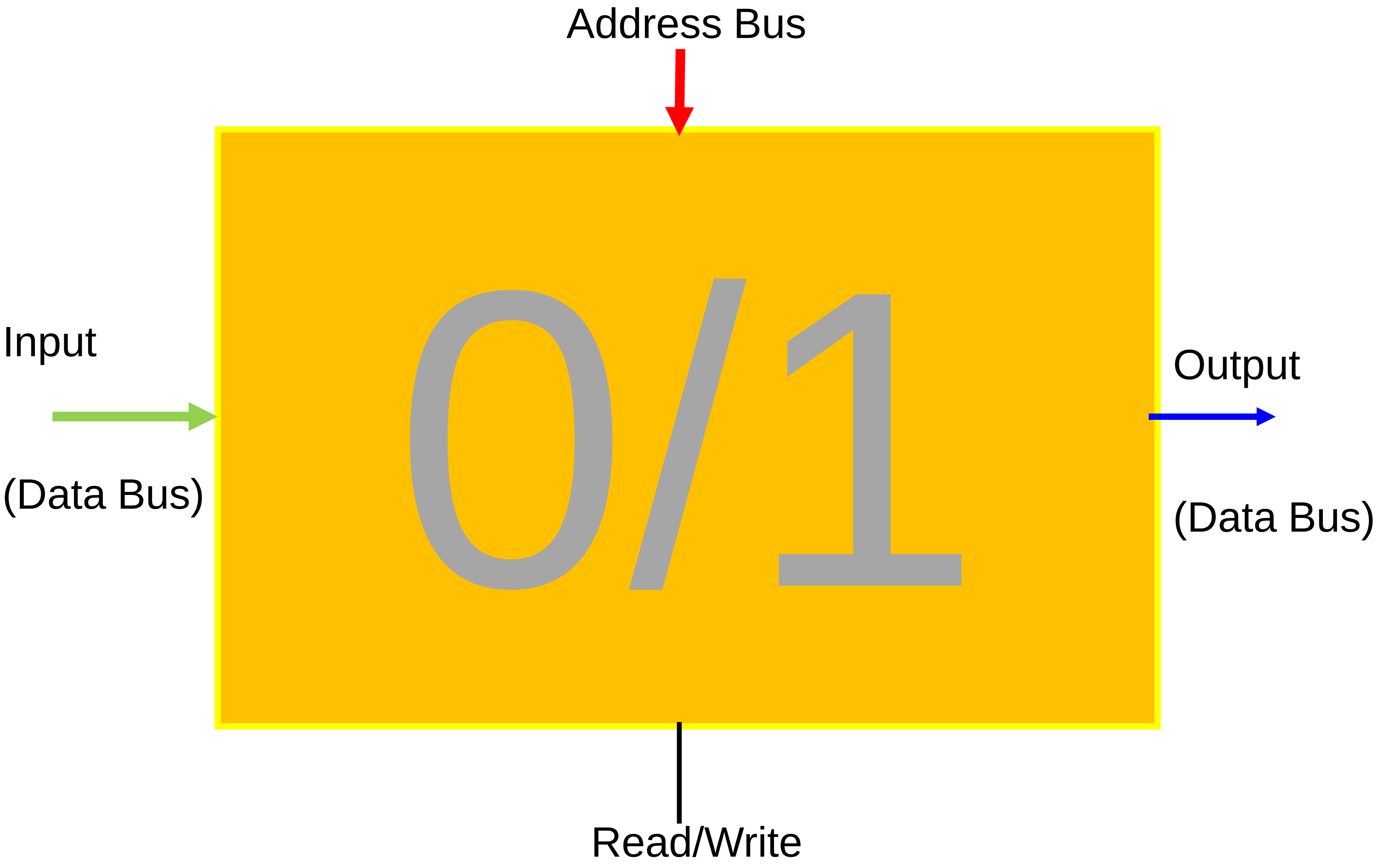

Address Lines

Data in

Data out

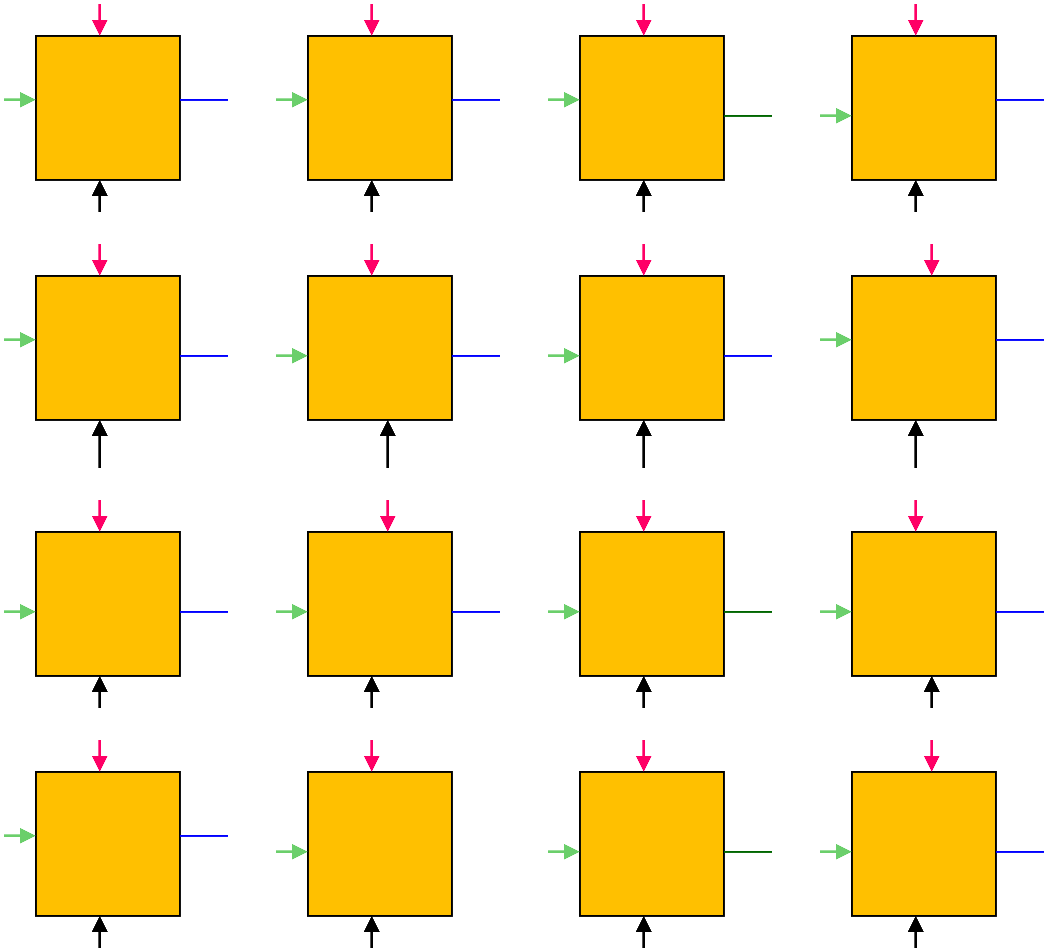
Read/Write

Basic memory cell

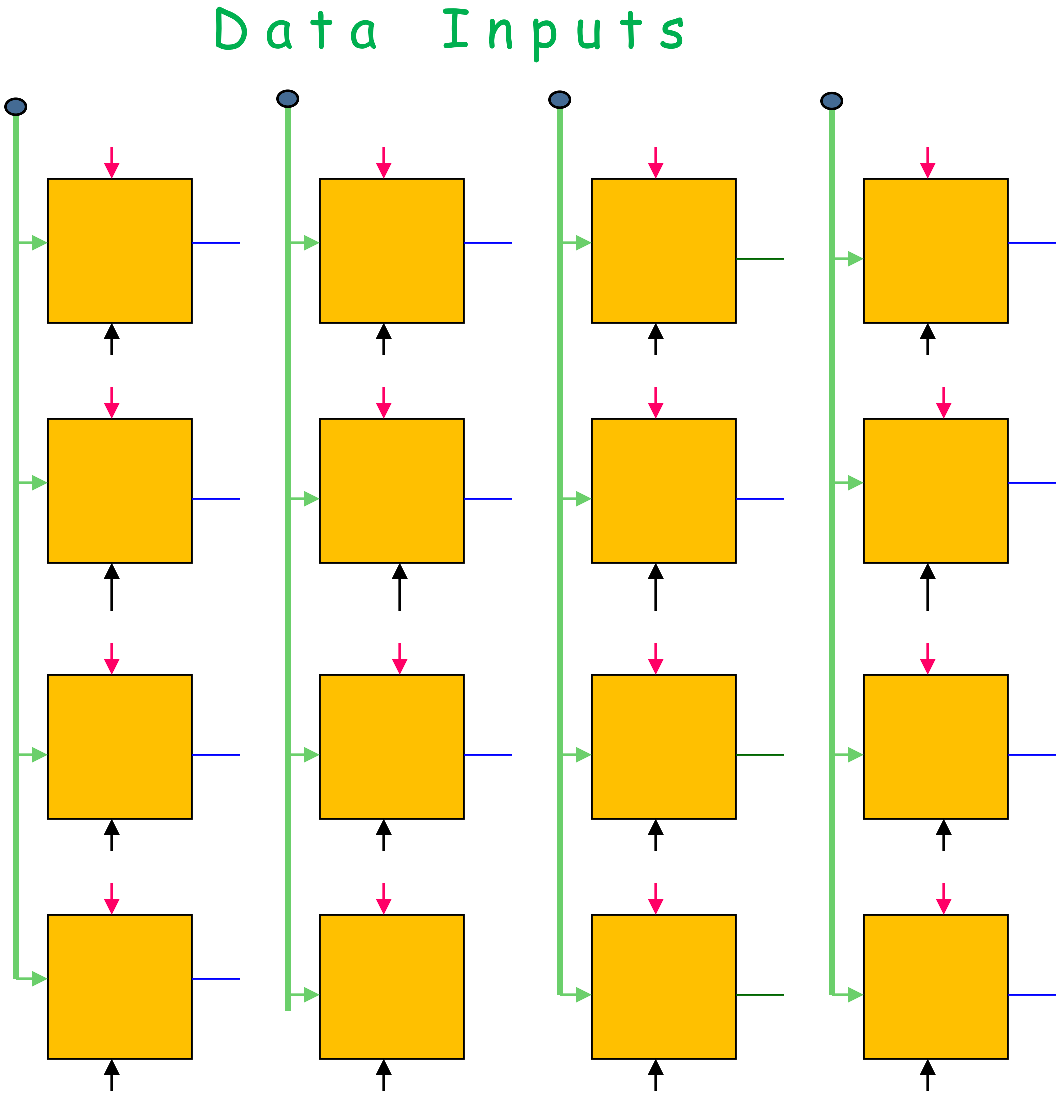


Design a 16x4 bits RAM memory

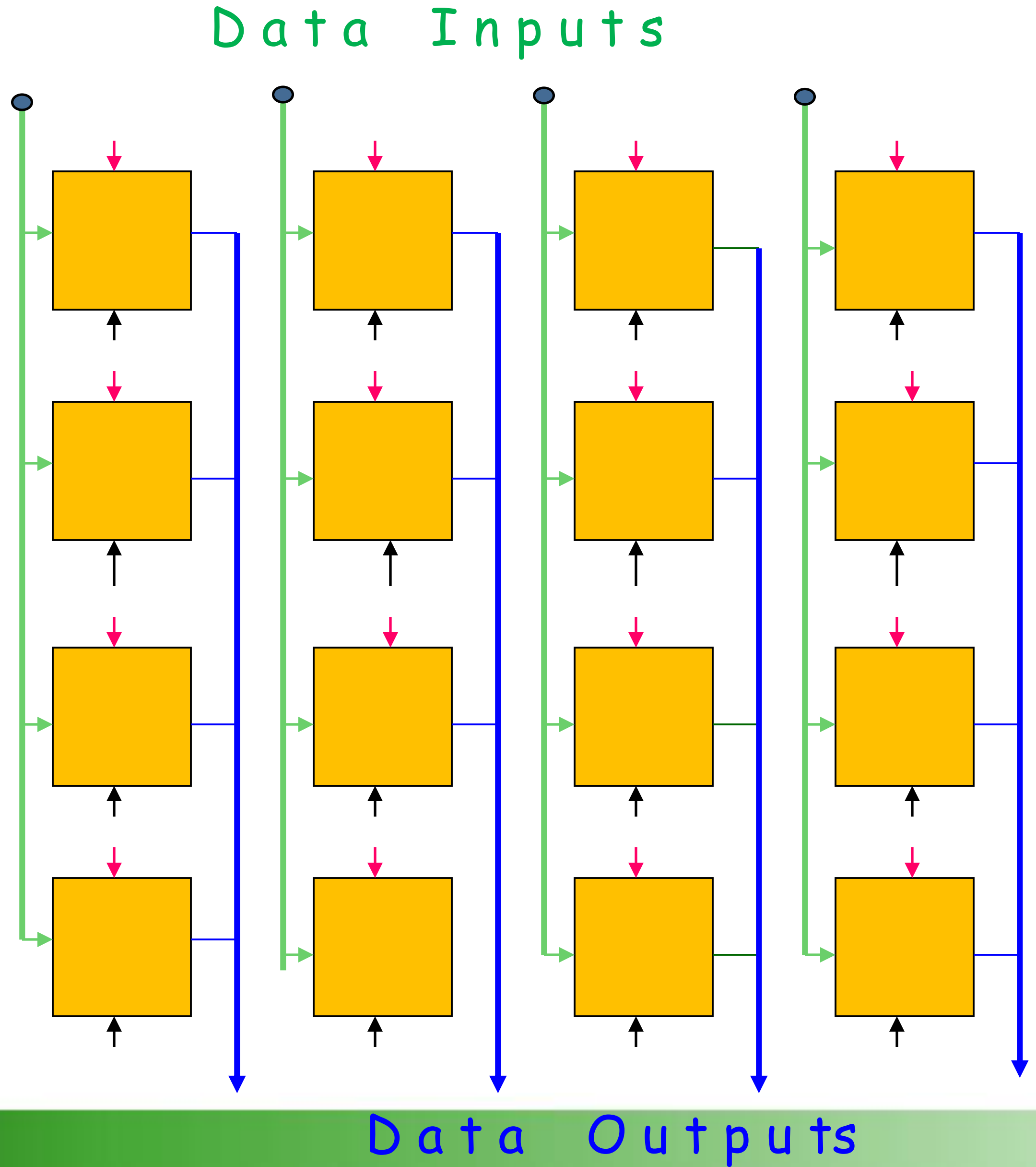
Memory system: 16 x 4 bits



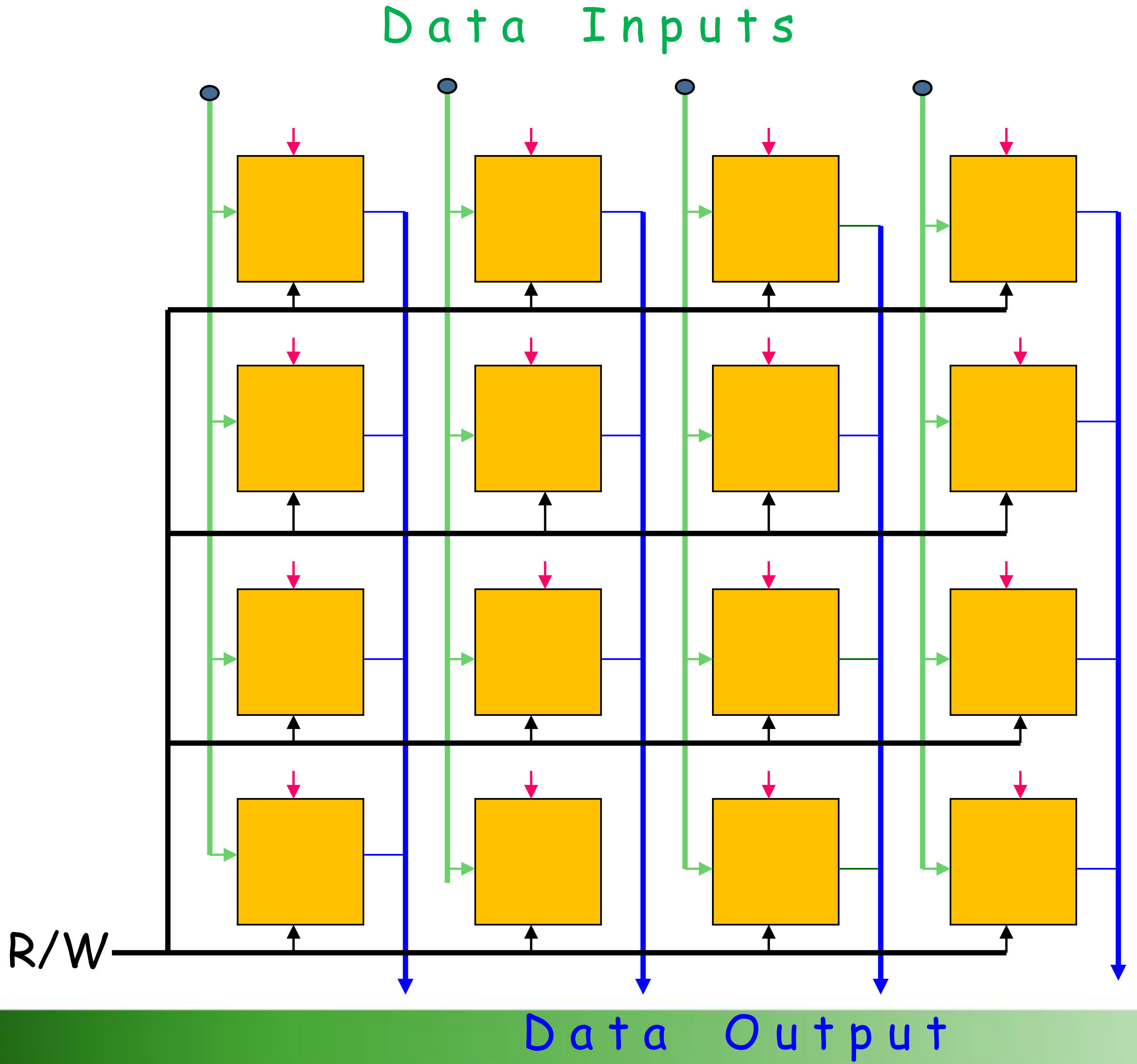
Data inputs



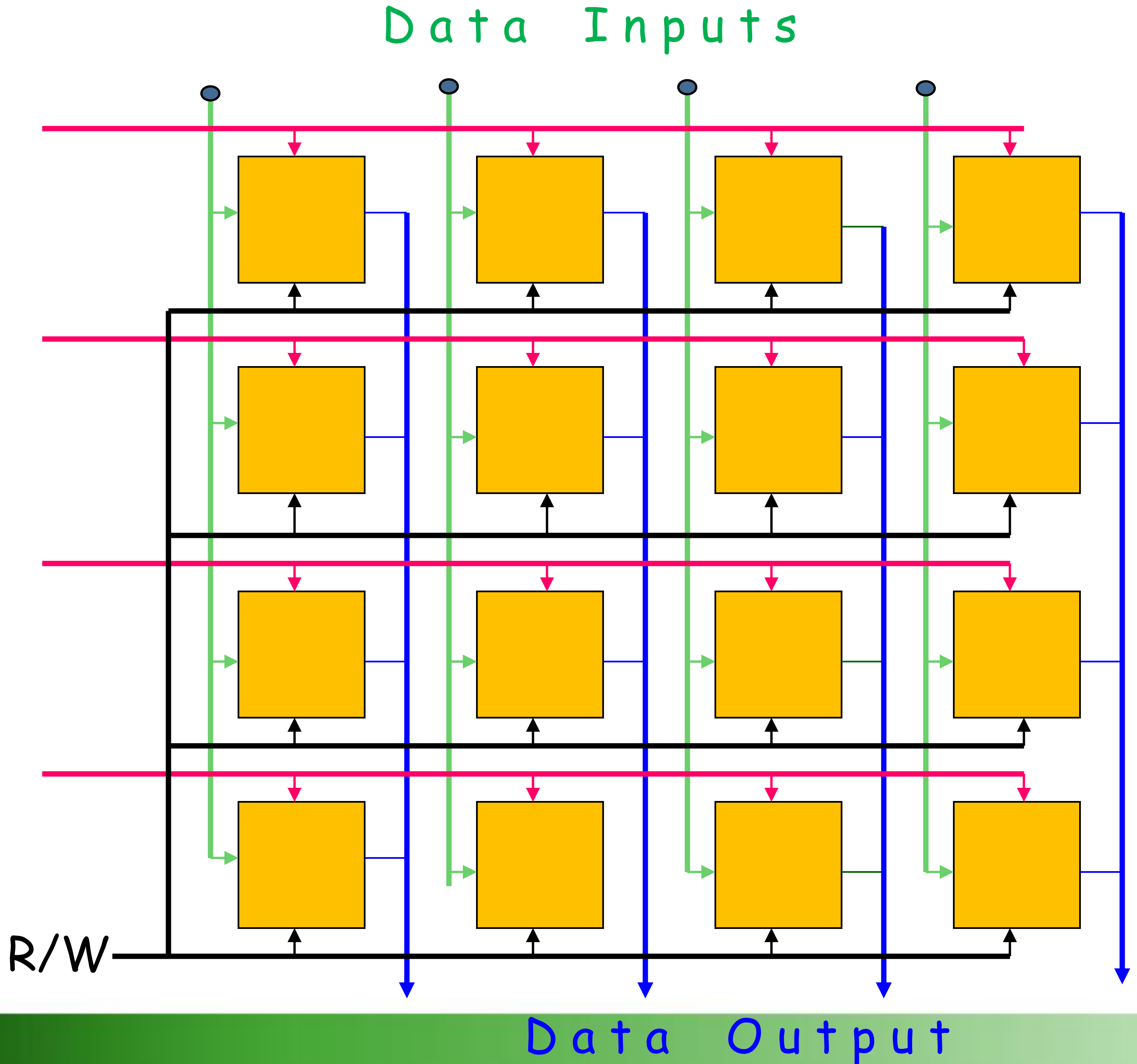
Data outputs



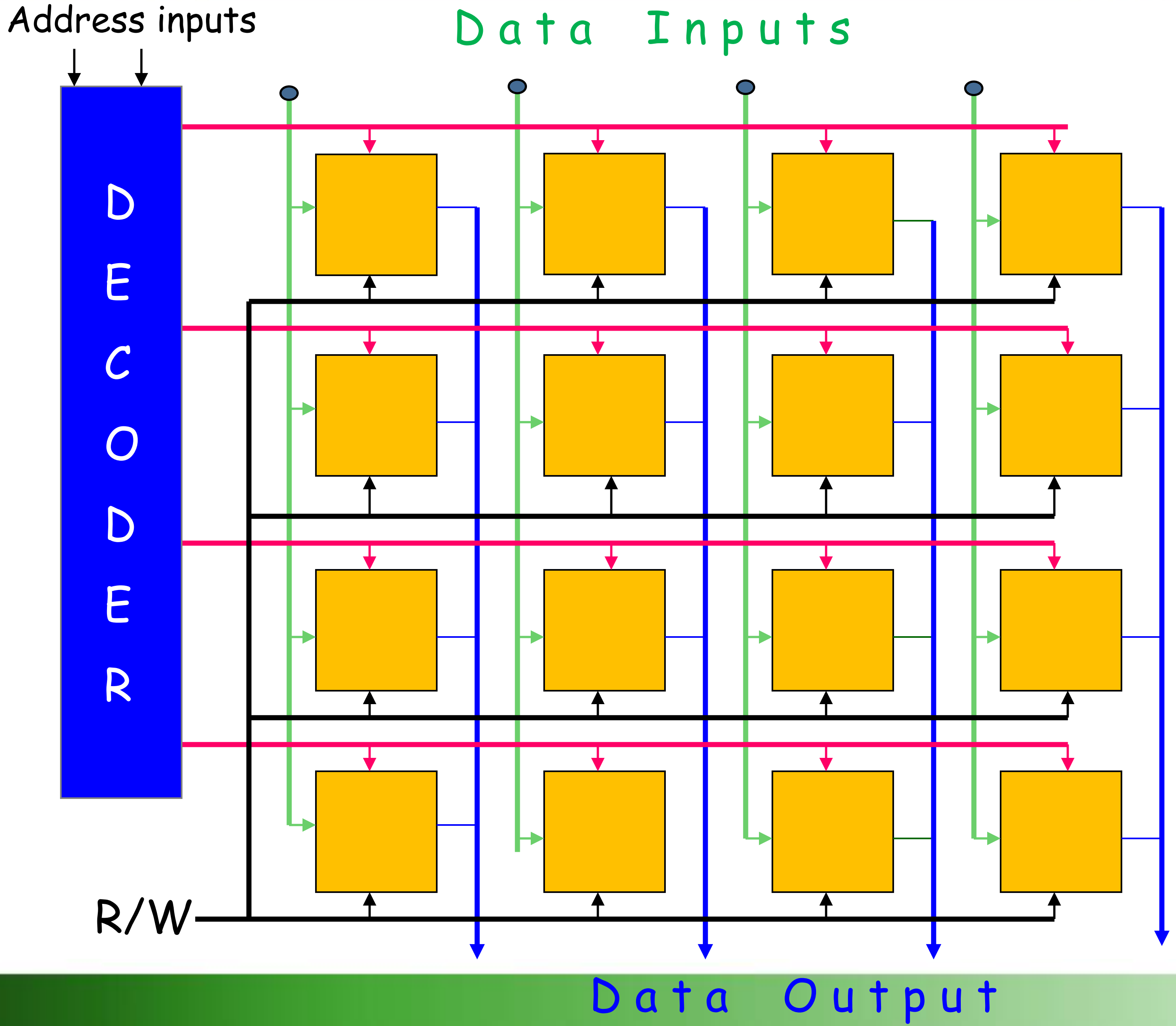
Read/Write (R/W) line

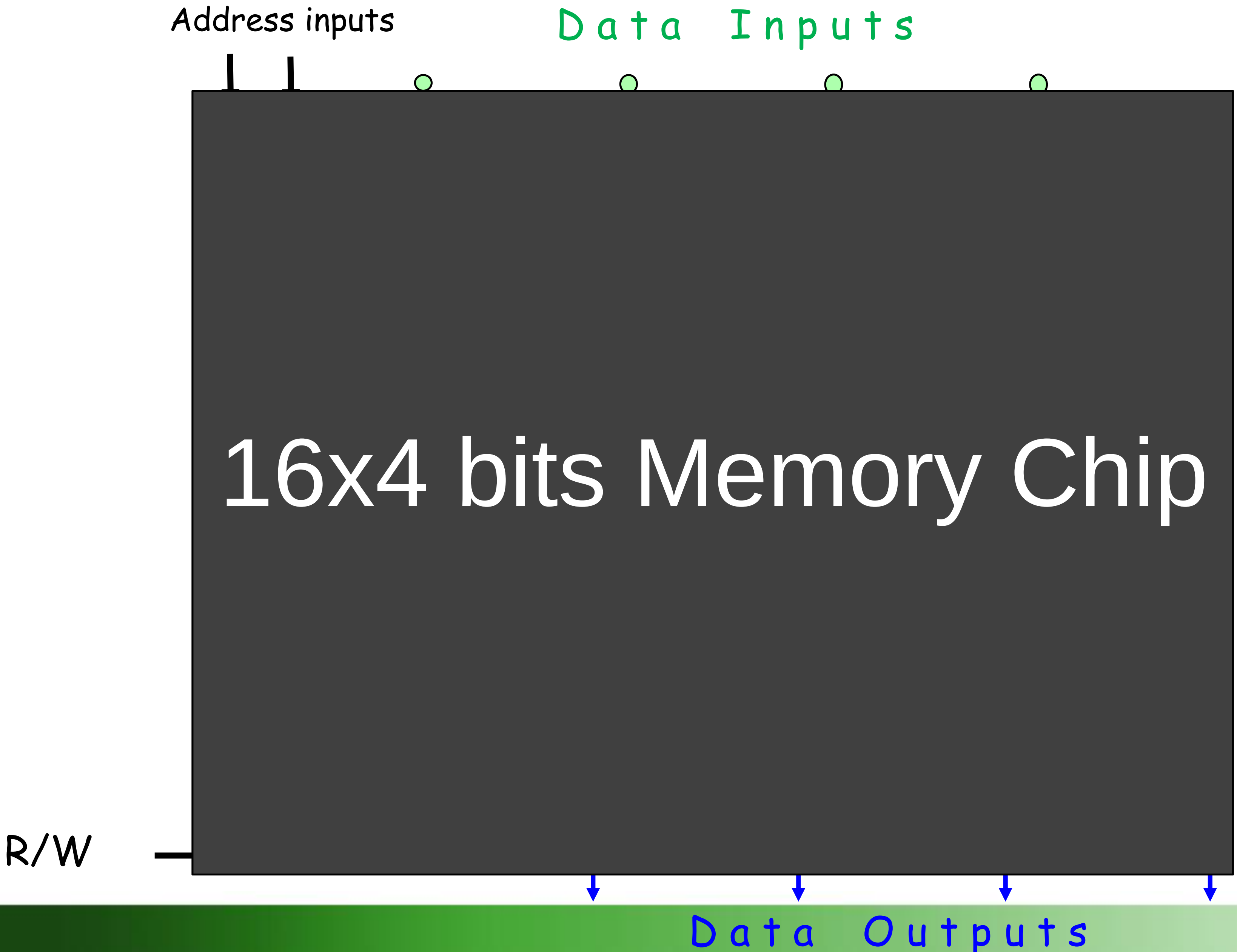


Address lines: 4 (16 = 2^4)



With a 2-4 Decoder 2 lines are fine …





Address inputs

Data Inputs

16x4 bits Memory Chip

R/W

Data Outputs

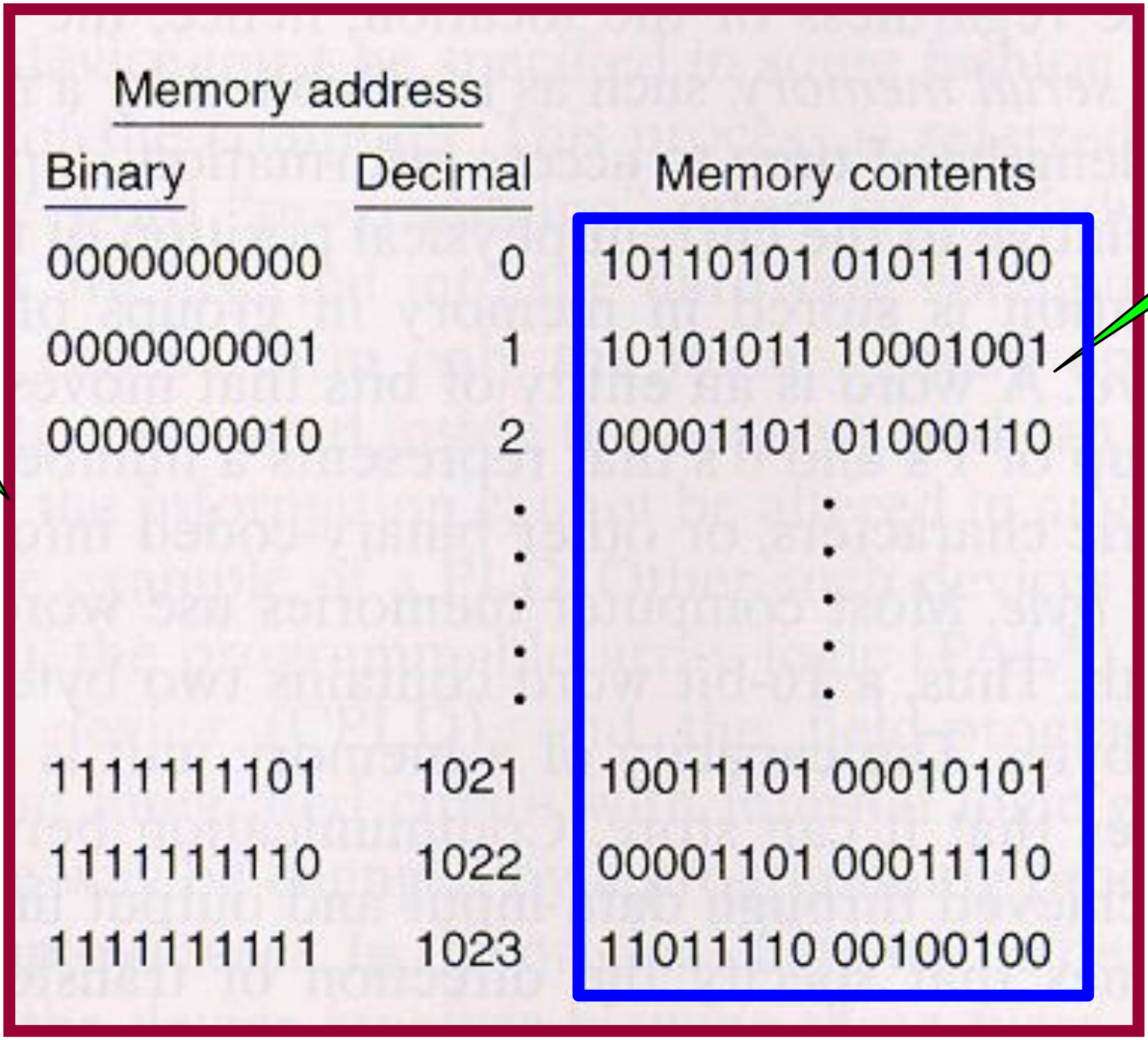
A 1Kb DRAM memory chip

1Kb = 1024 = 2^{10} bits

https://en.wikipedia.org/wiki/Intel_1103

1024 x 16 memory (1Kb x 16)

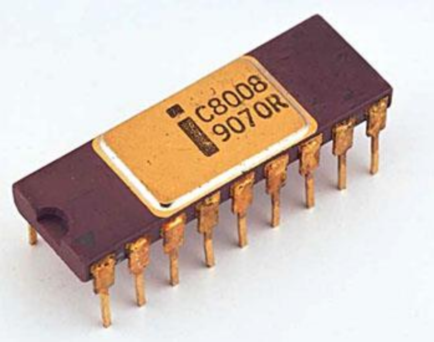
10 address lines.
Since 2^{10} =1024

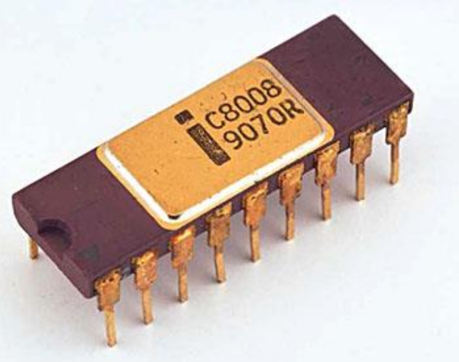


Memory address		
Binary	Decimal	Memory contents
0000000000	0	10110101 01011100
0000000001	1	10101011 10001001
0000000010	2	00001101 01000110
	⋮	⋮
1111111101	1021	10011101 00010101
1111111110	1022	00001101 00011110
1111111111	1023	11011110 00100100

data

Memory design using memory-chips





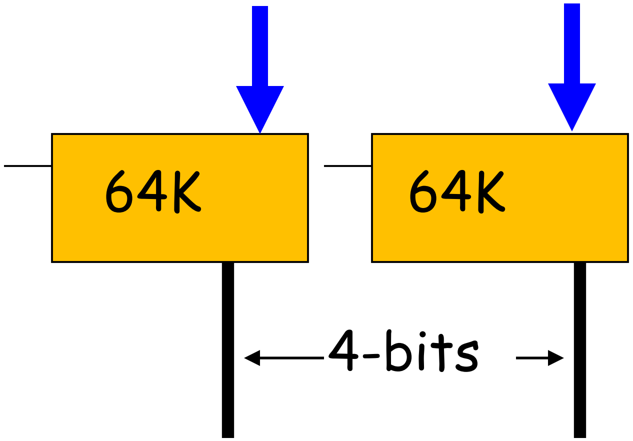
Use two 64K x 4 ROM memory modules to create a 64K x 8 ROM memory system.

How many bits has the:

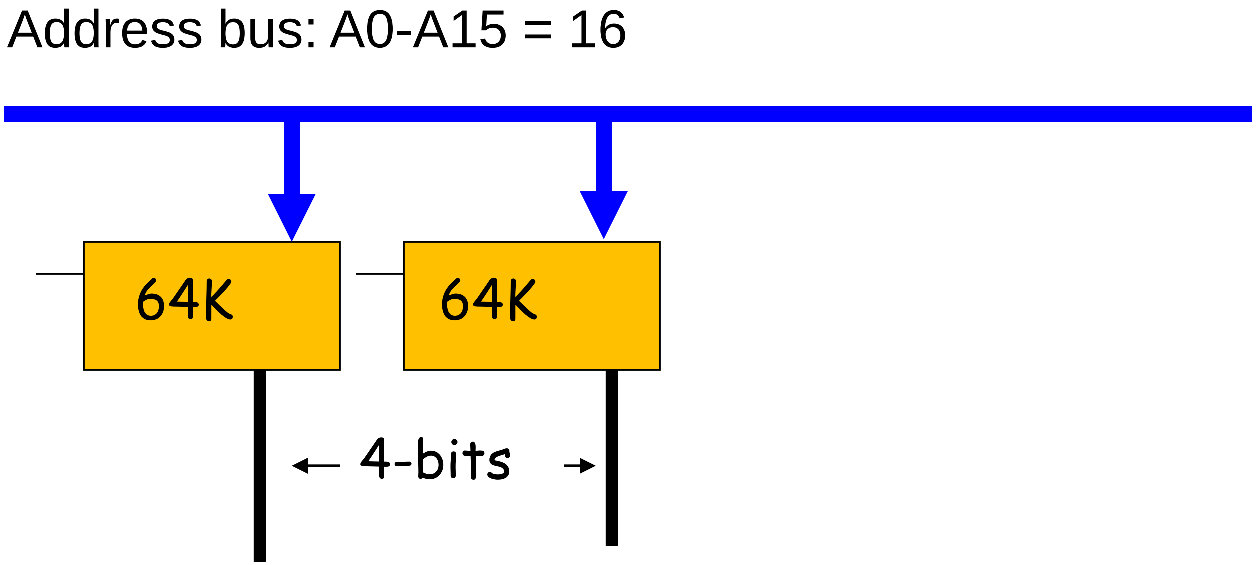
→ Data bus (?)

→ Address bus (?)

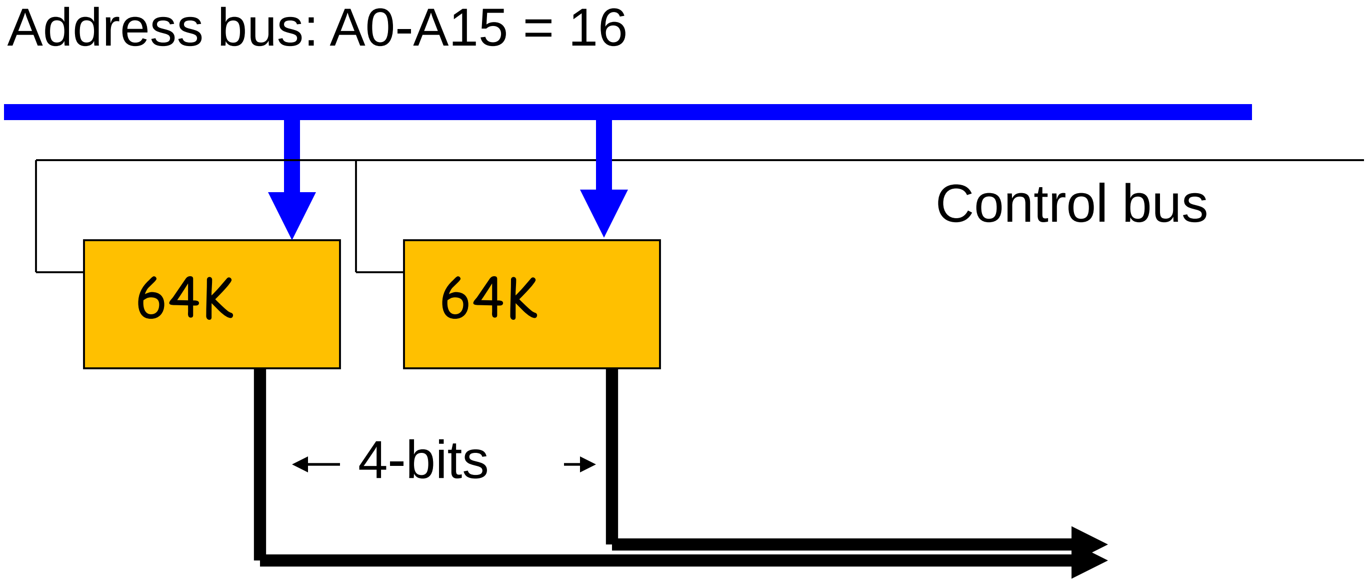
64K x 8 ROM, using 2-64 x 4



64K x 8 ROM, using 2-64 x 4



64K x 8 ROM, using 2-64 x 4



Data Bus: D0-D7 = 8

Address Bus: 16

Data Bus: 8

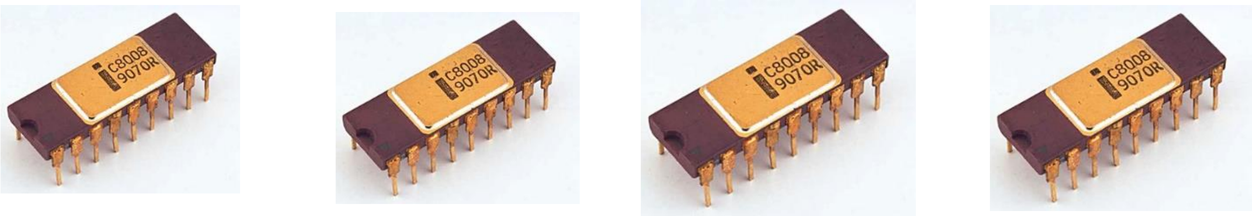
Use (?) 64K x 4 ROM memory modules to create a 64K x 16 ROM memory system.

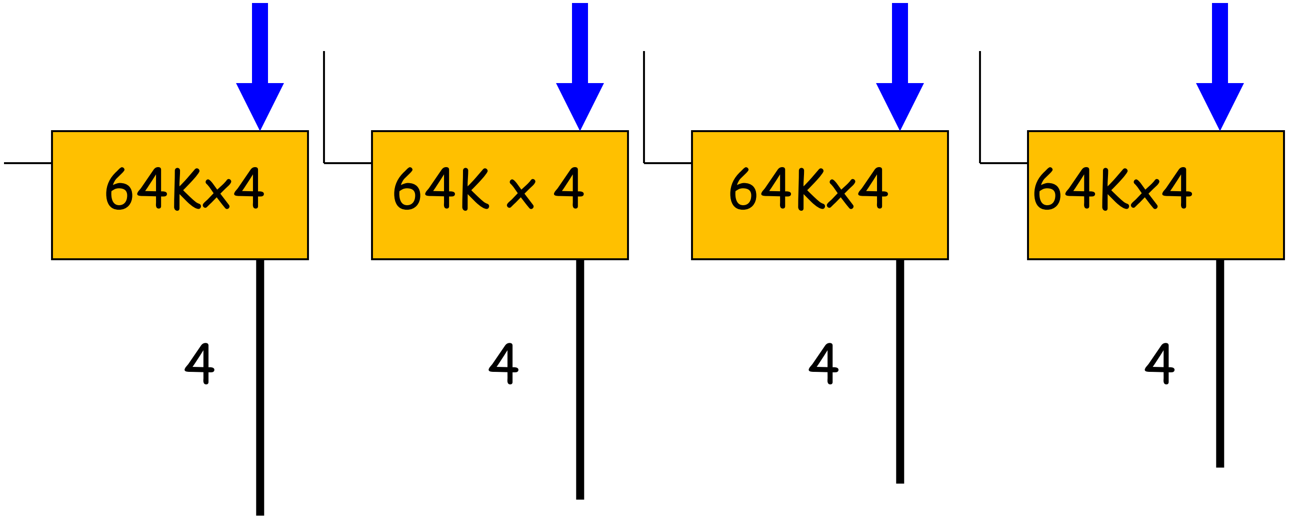
How many bits has the:

→ Data bus (?)

→ Address bus (?)

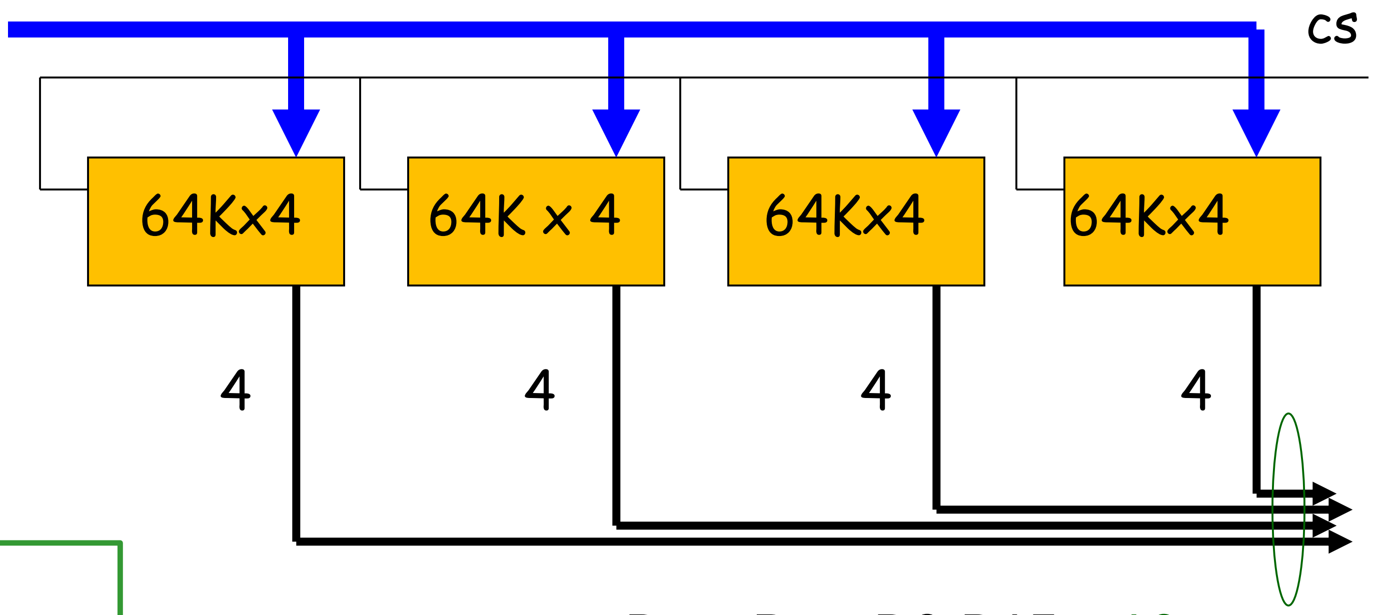
64 K x 16 ROM





64 K x 16 ROM

Address bus: A0-A15 = 16



Data Bus: D0-D15 = 16

Address Bus: 16

Data Bus: 16

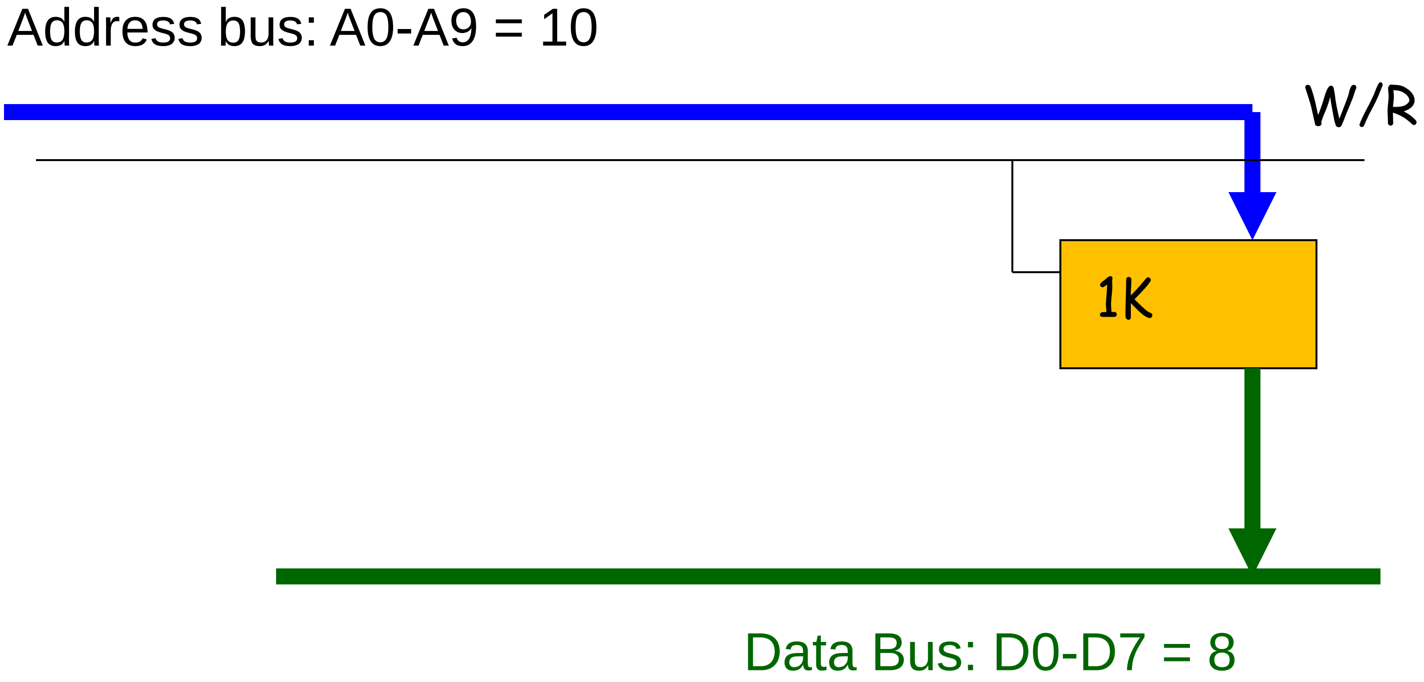
Use (?) 1k x 8 RAM memory modules to create a 4k x 8 RAM memory system.

How many bits has the:

→ Data bus (?)

→ Address bus (?)

4K x 8 (using 1Kx8) RAM



Address bus: A0-A9 = 10

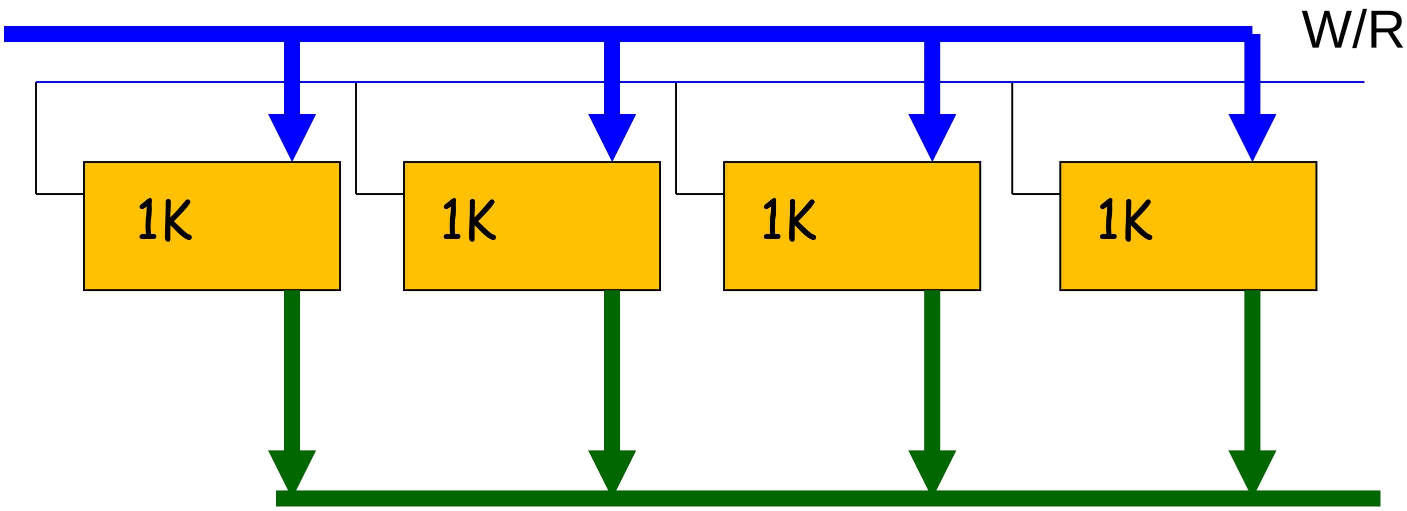
W/R

1K

Data Bus: D0-D7 = 8

What else?

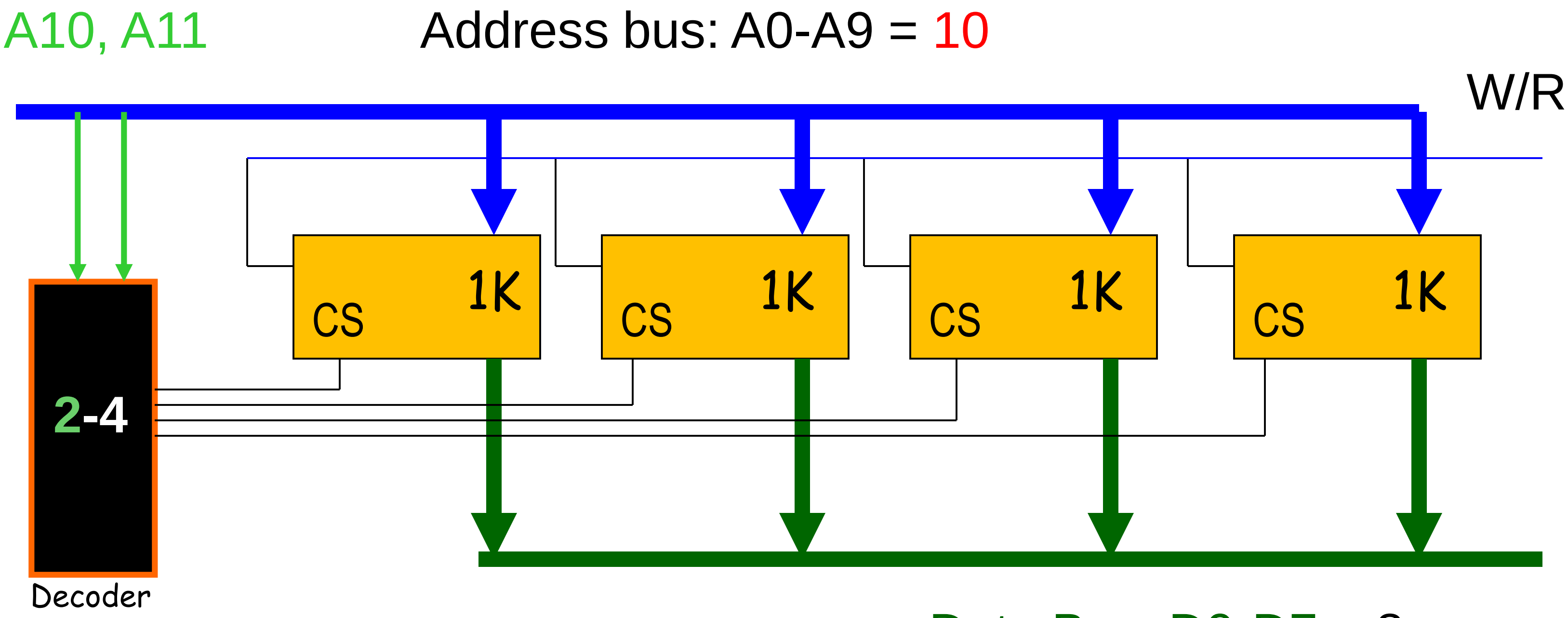
Address bus: A0-A9 = 10



Data Bus: D0-D7 = 8

What else?

4K x 8 (using 1Kx8's) RAM



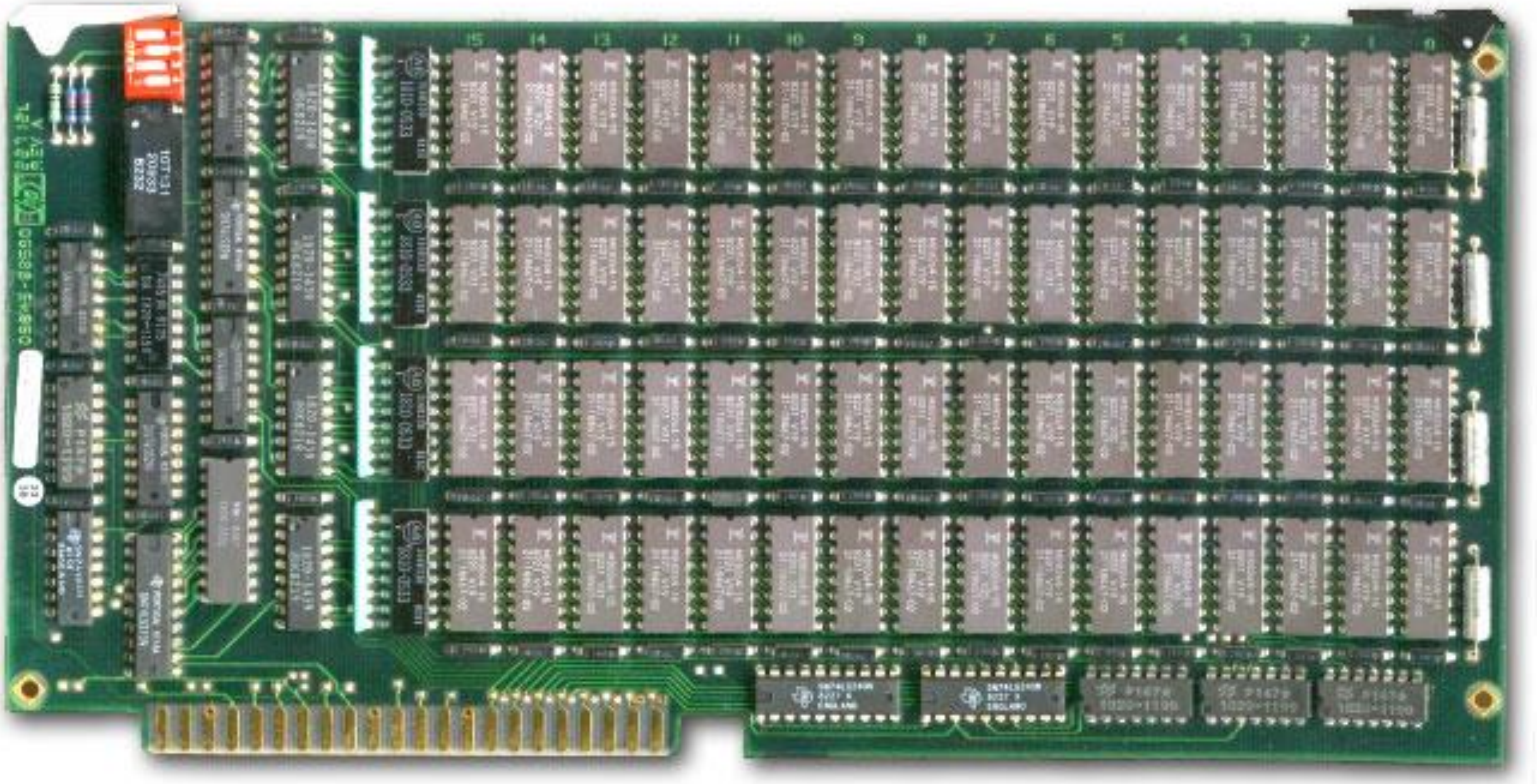
Data Bus: D0-D7 = 8

Address Bus: 10 + A10 + A11 = 12

Data Bus: 8

CS = Chip Selector

RAM Memory board



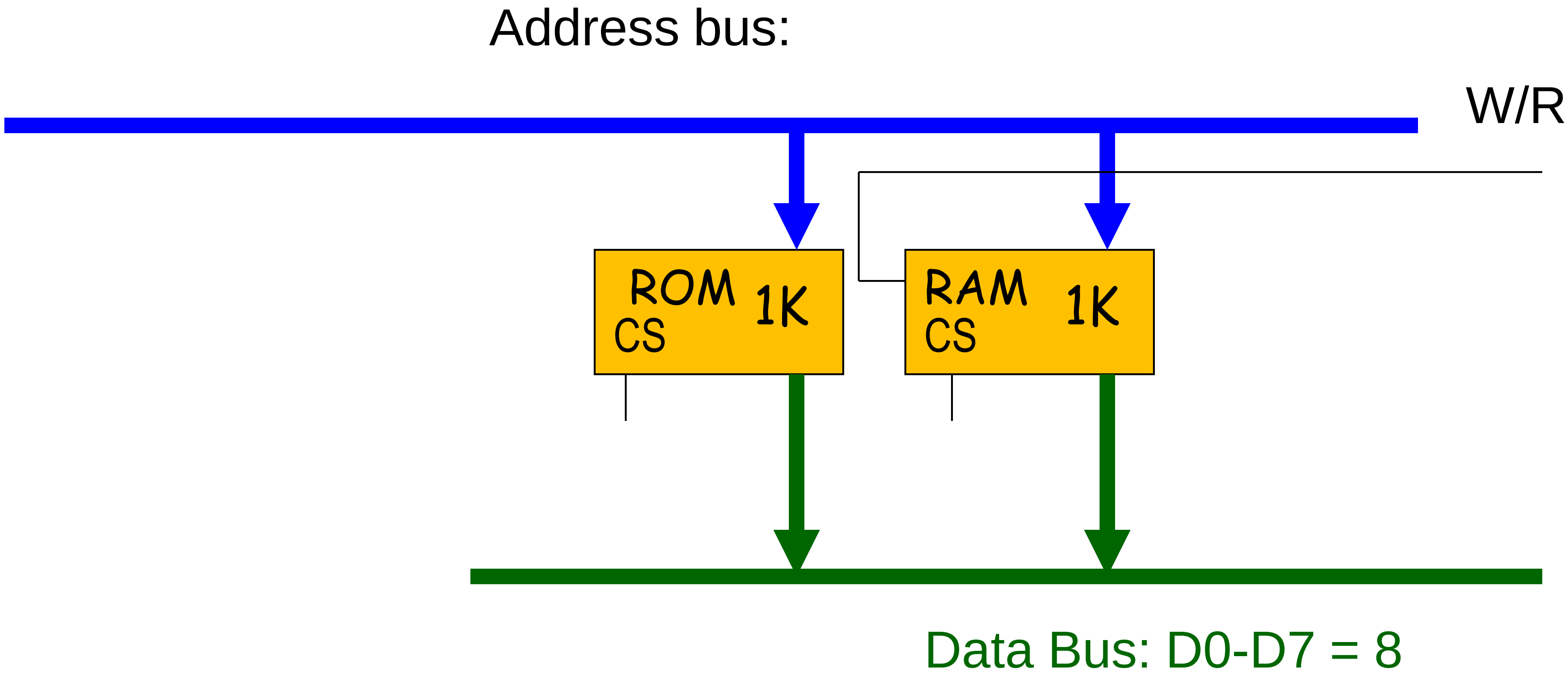
Design a 2K x 8 Memory system using:
RAM (1K x 8) + ROM (1K x 8)

How many bits has the:

→ Data bus (?)

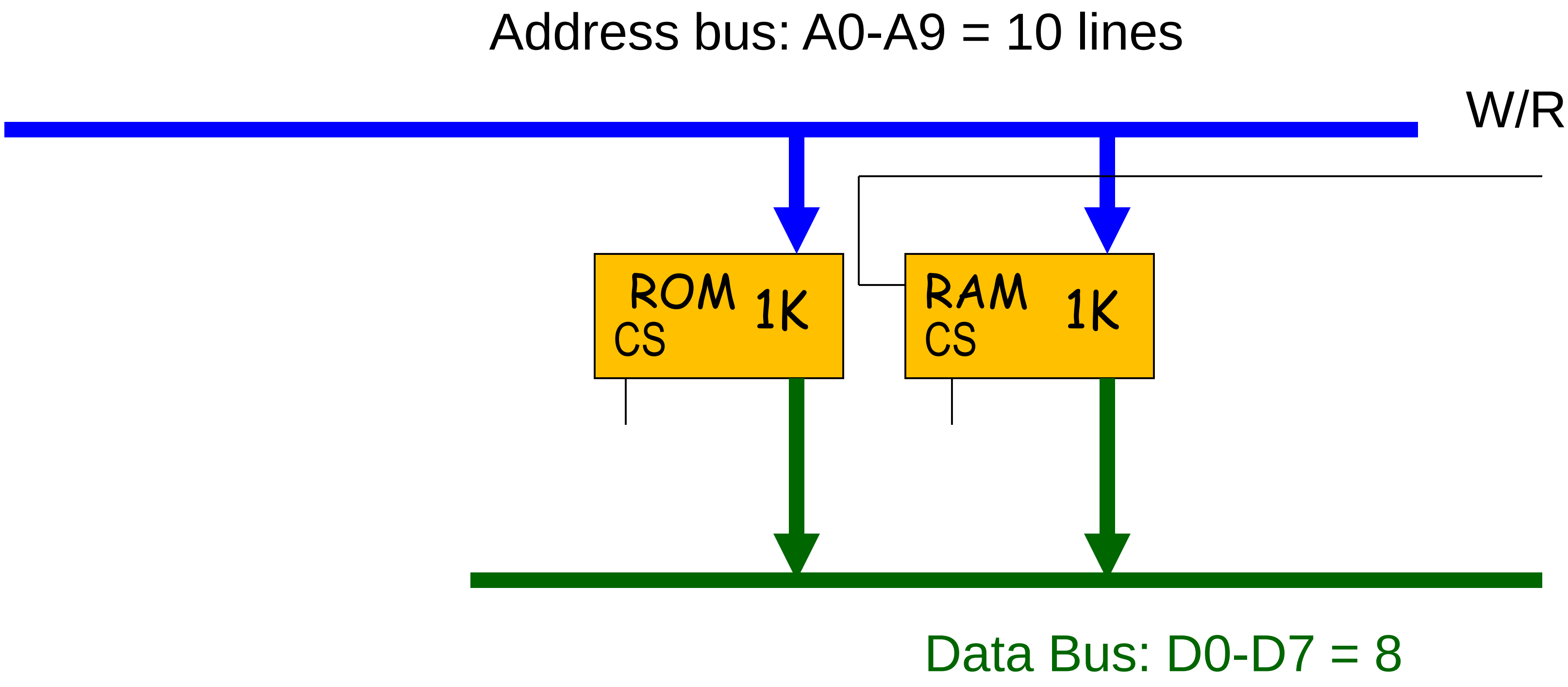
→ Address bus (?)

2K x 8: RAM + ROM



CS = Chip Selector

2K x 8: RAM + ROM



Address bus: A0-A9 = 10 lines

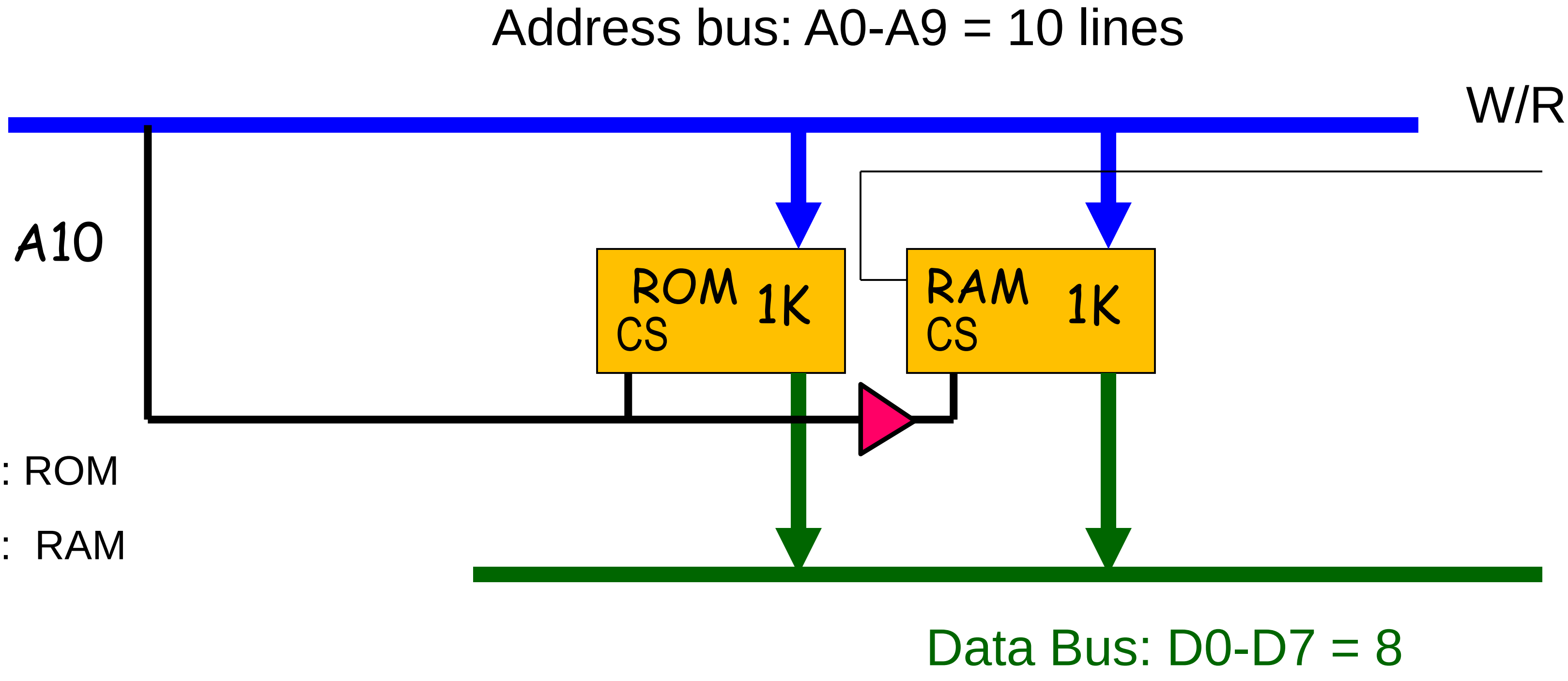
W/R

ROM 1K
CS

RAM 1K
CS

Data Bus: D0-D7 = 8

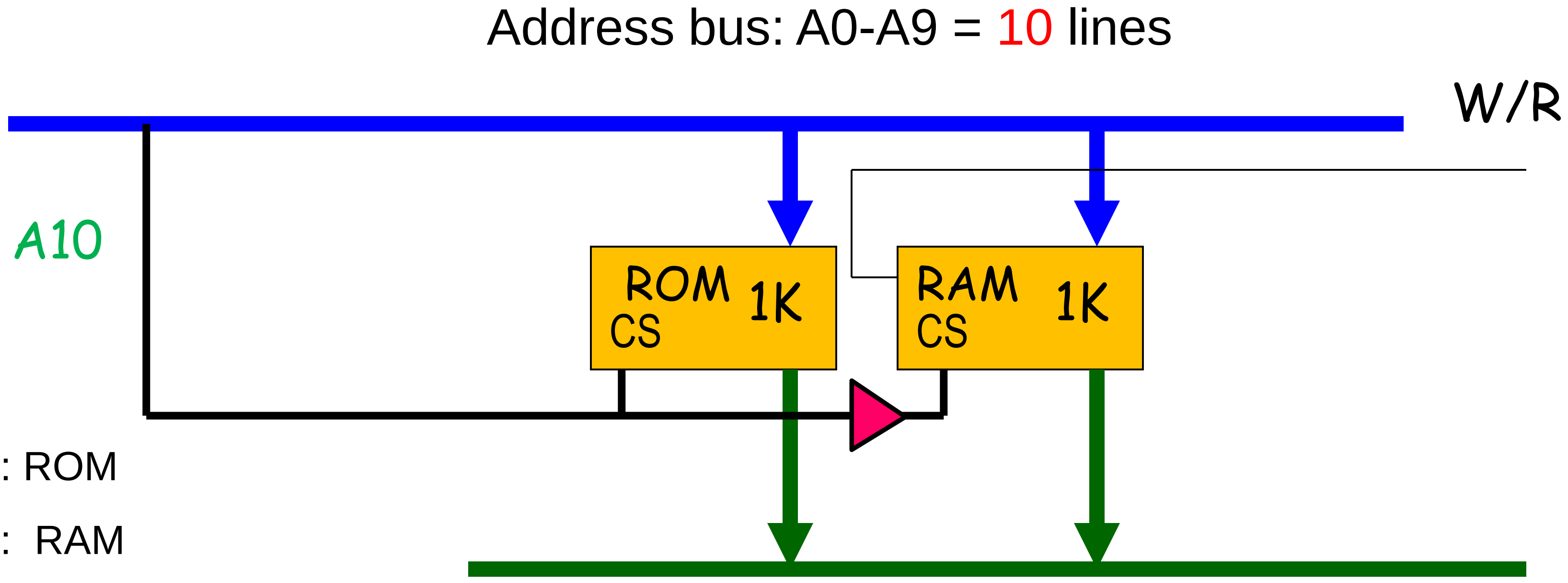
2K x 8: RAM + ROM



A10 = 0: ROM

A10 = 1: RAM

2K x 8: RAM + ROM



A10 = 0: ROM

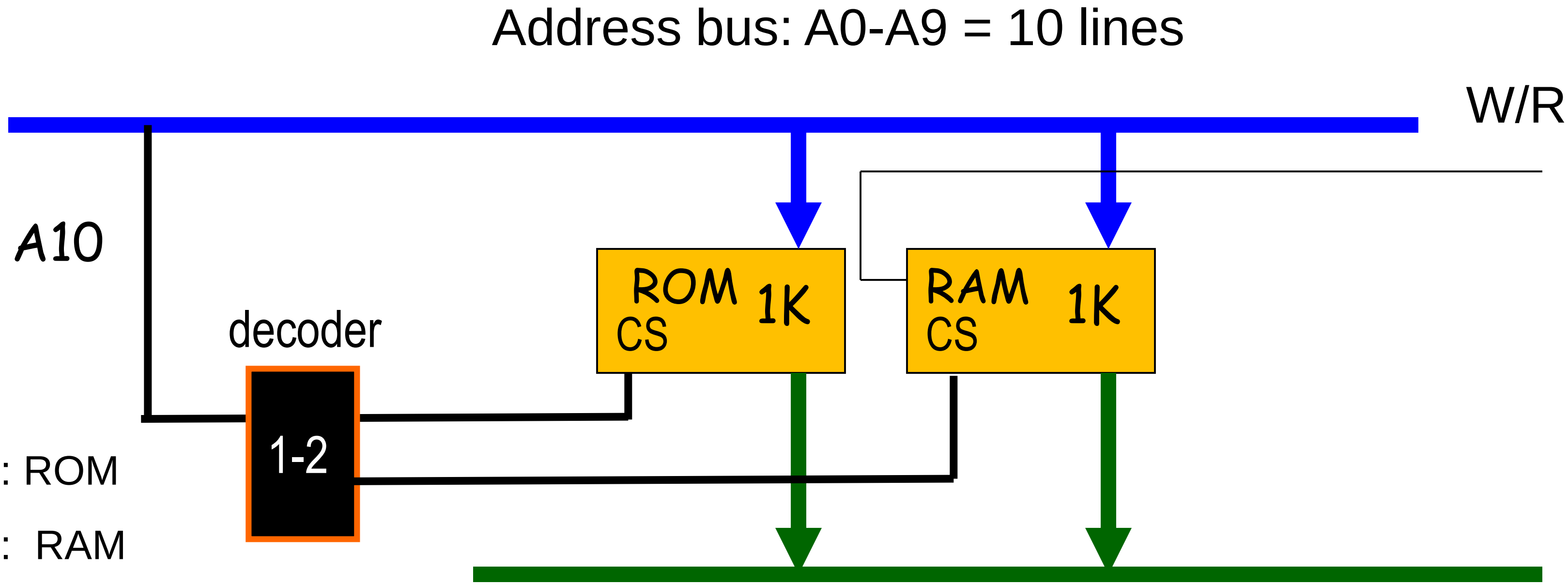
A10 = 1: RAM

Data Bus: D0-D7 = 8

Address Bus: 10+A10 = 11

Data Bus: 8

Equivalently using a decoder



A10 = 0: ROM

A10 = 1: RAM

Address Bus: 11

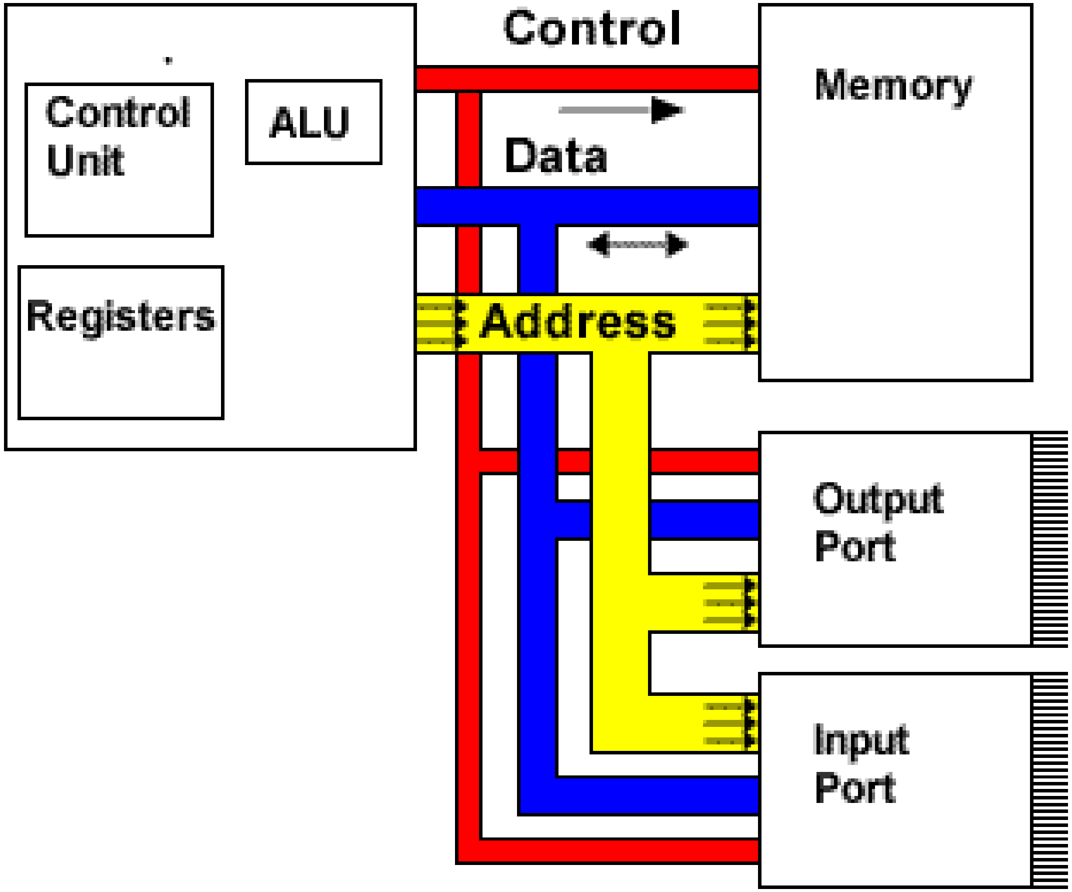
Data Bus: 8

Data Bus: D0-D7 = 8

ROM & RAM

ROM	RAM
• Store instructions (permanent) to communicate with hardware components (Input/Output devices) • Store instructions (permanent) to implement functions (logic, arithmetic, …) • …	• Allows data (temporary) to be read or written … • Store (temporary) CPU and memory transfers • Stores (temporary) currently used program instructions … • …

We studied …


Control
Control Unit
ALU
Registers
Data
Address
Memory
Output Port
Input Port

End