

alem x Turing

OS, Процессор, Память, I/O

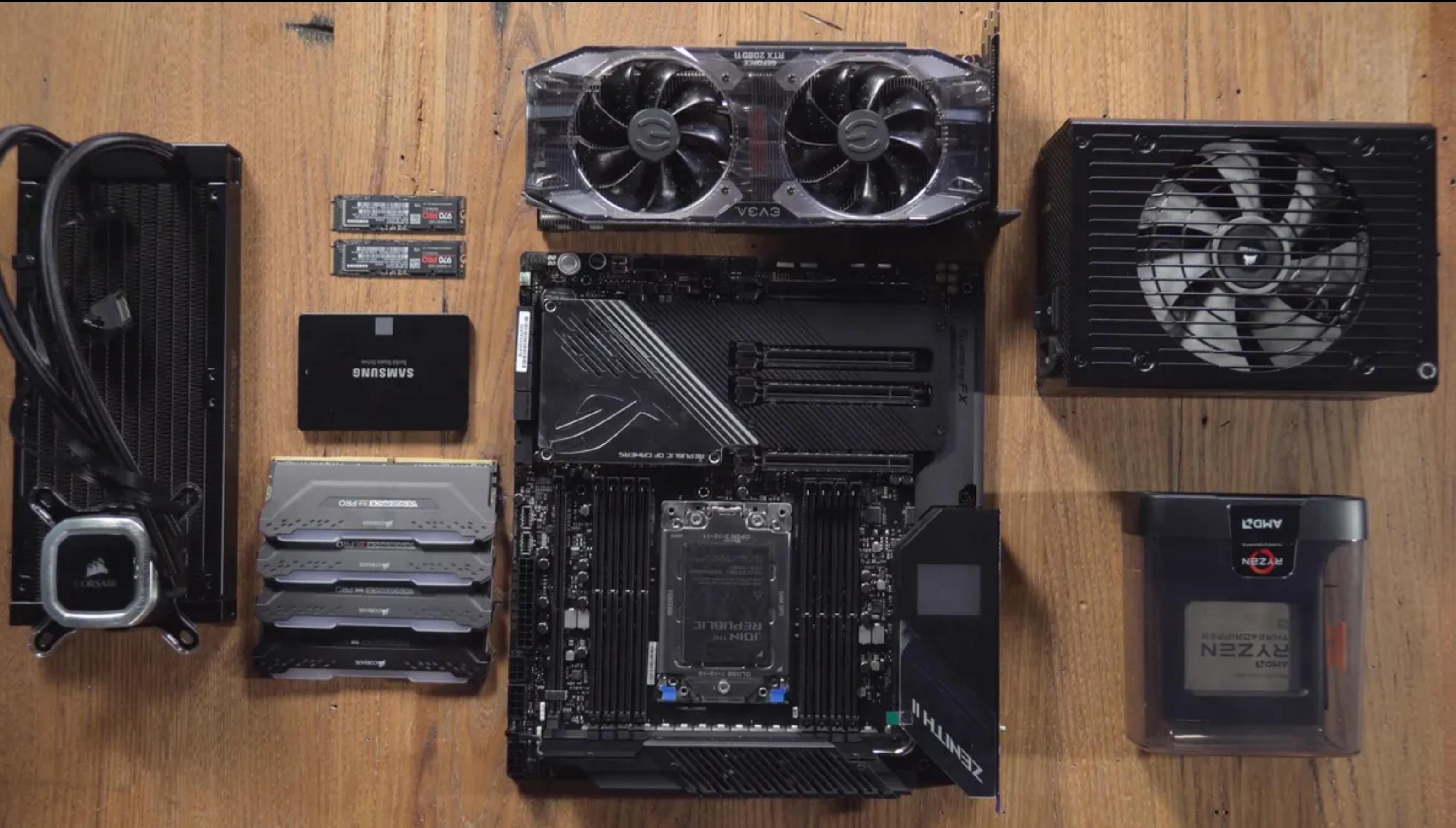
03.12.2022 Abylaikhan Zulbukharov

Абстракция





Компоненты компьютера



Память

MEMORY	
Address	Data
0	0000000000000000
1	0000000000000000
2	0000000000000000
3	0000000000000000
4	0000000000000000
5	0000000000000000
6	0000000000000000
7	0000000000000000
8	0000000000000000

Процессор

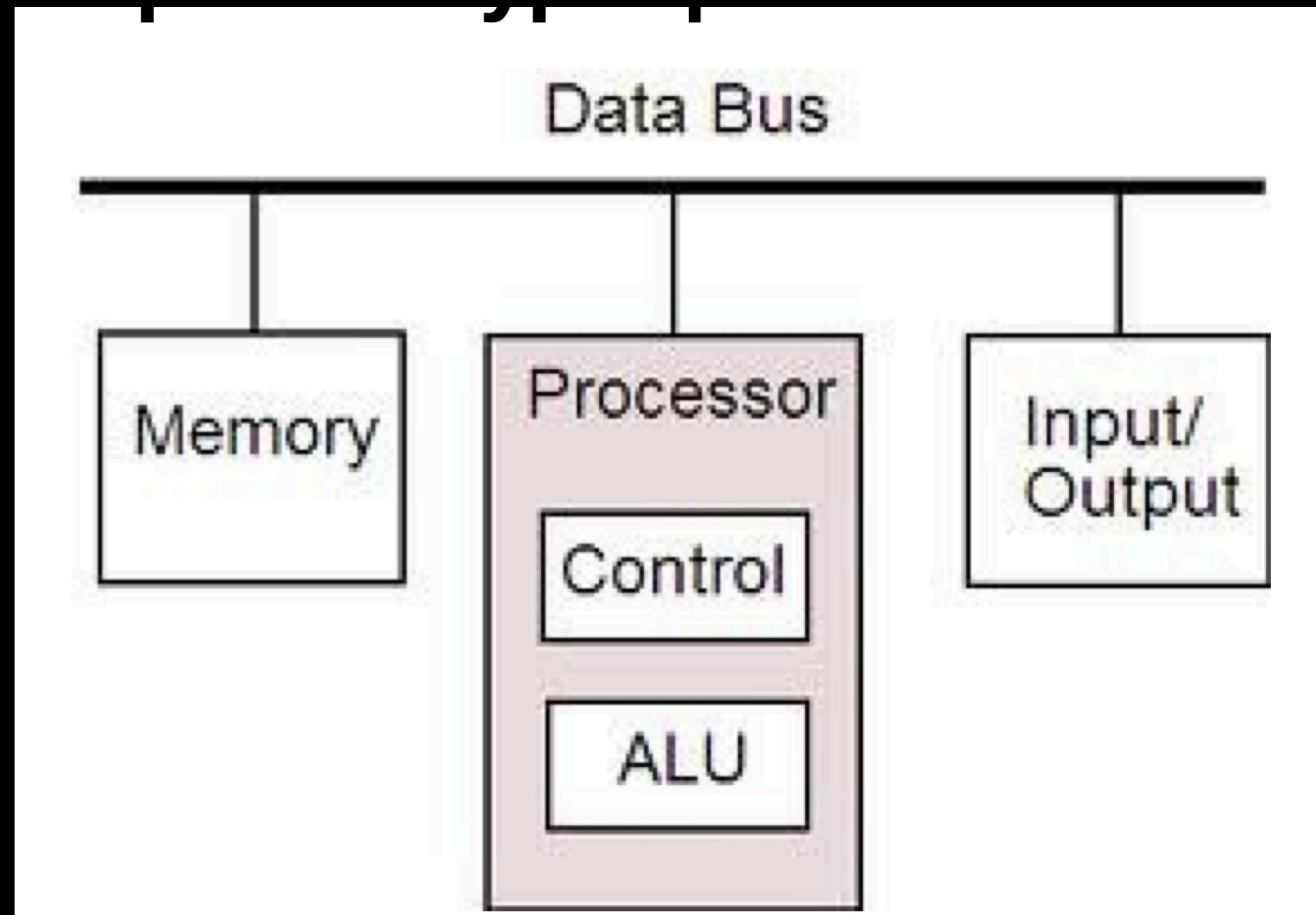
- Должен уметь быстро исполнять команды.
- Должен иметь доступ к памяти.
- Должен быть многозадачным.
- При ошибках, должен оповещать.
- Должен иметь состояние.

Набор машинных инструкций

- перемещение данных (mov, push, pop)
- арифметика/логика (add, sub, inc, dec, and, or, xor)
- поток управления (jmp, je, jne, jg, call)

`mov ax, bx ; moves contents of register bx into ax`

`mov ax, 010Ch ; moves value of 0x010C into register ax`



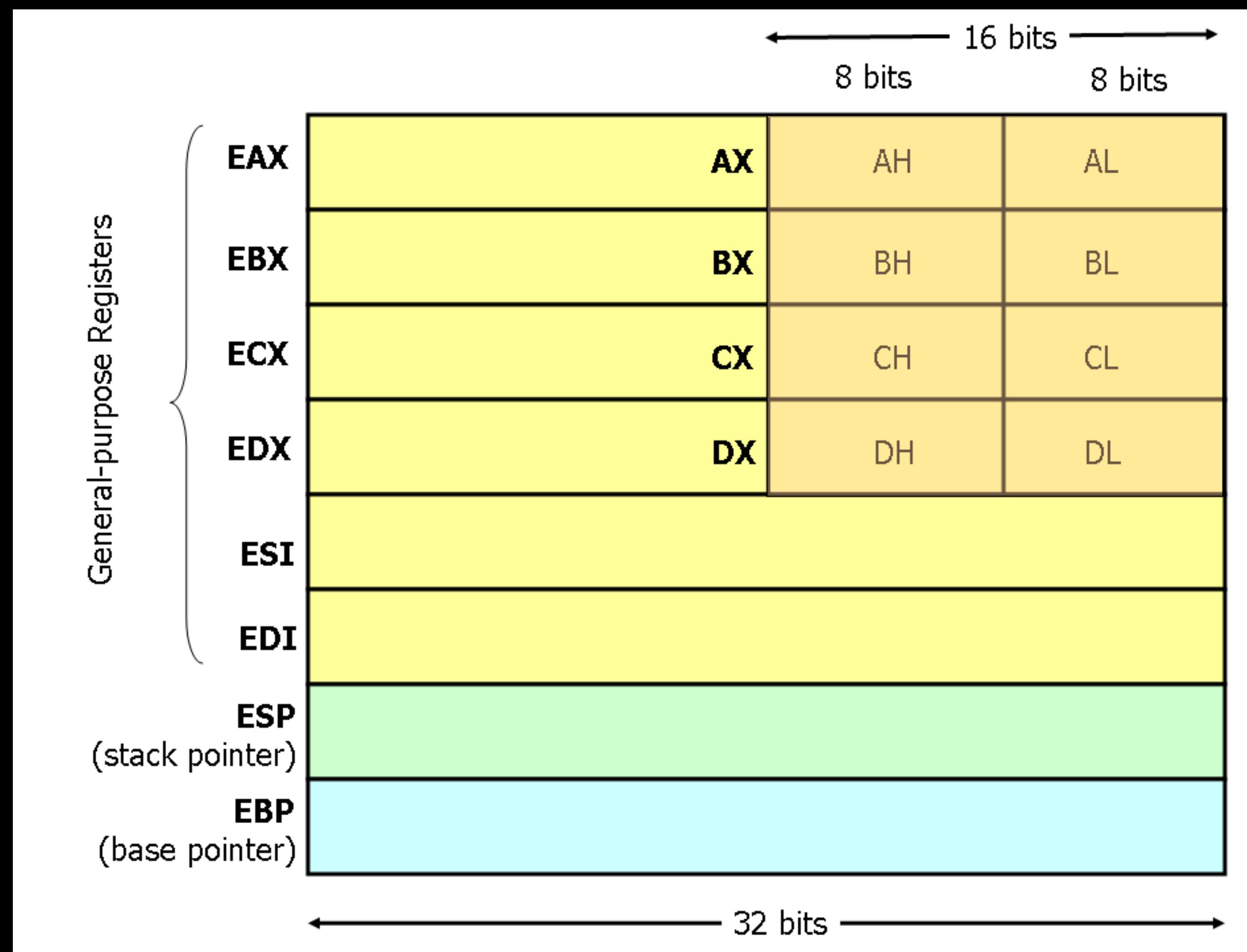
Процессоры

- Различные архитектуры (ARM, x86).
- Выполняет инструкции из памяти.
- Умеет выполнять простые задачи.
- Регистры.
- Таймер.

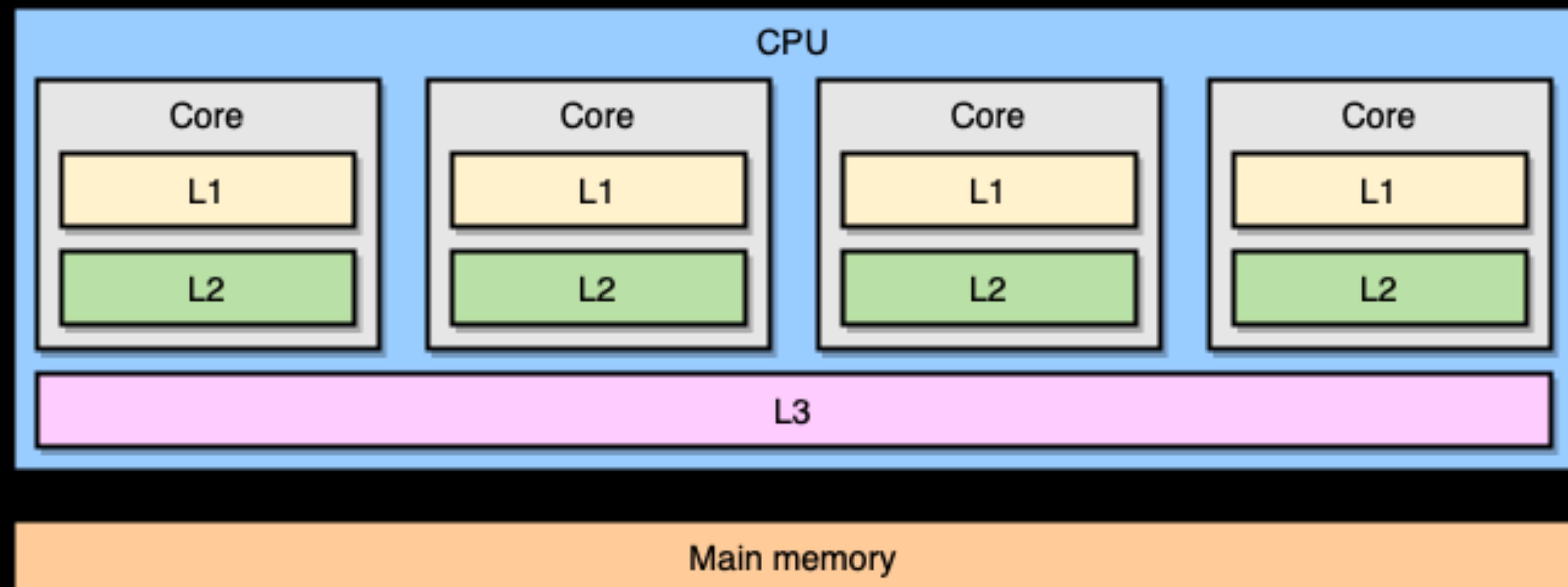
ARM vs x86

- x86
 - Долгое выполнение операций сложения, умножения, доступа к памяти.
 - Довольно древняя со своими подводными камнями.
 - Имеет большое количество возможных инструкций.
- ARM
 - Быстрая скорость, но с использованием большего количества инструкций.
 - Не поддерживается большим количеством программ.

Регистры



Кэш



OS

- Управление аппаратной частью (cpu, ram, disk).
- Абстракция аппаратной части. (API)
- Изоляция процессов от аппаратной части.

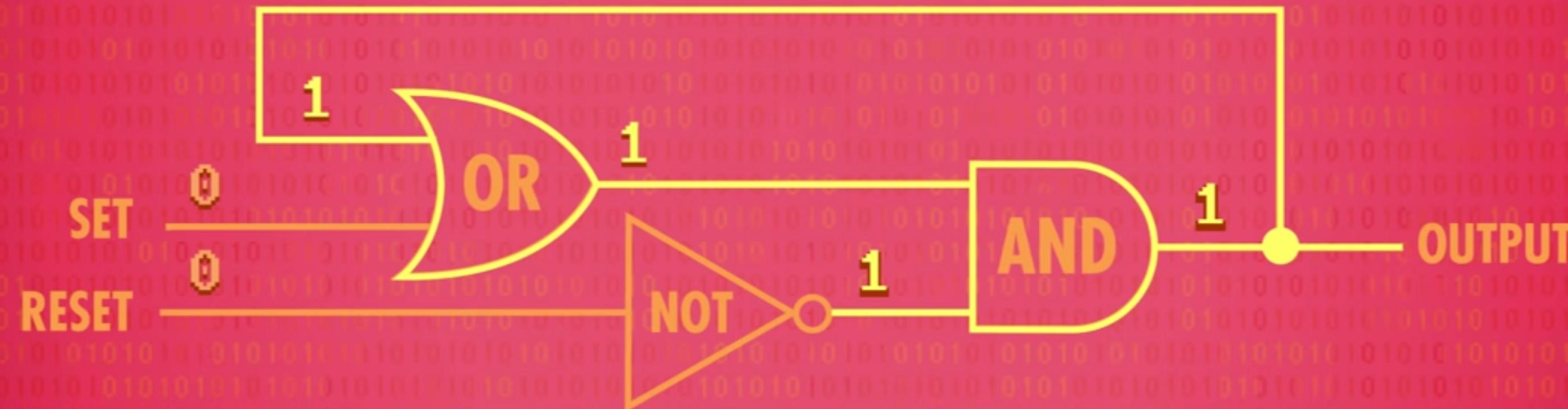
Память

- Должен сохранять данные.
 - Во время работы.
 - После выключения.
- Должен иметь адреса.

Память (simple RAM)



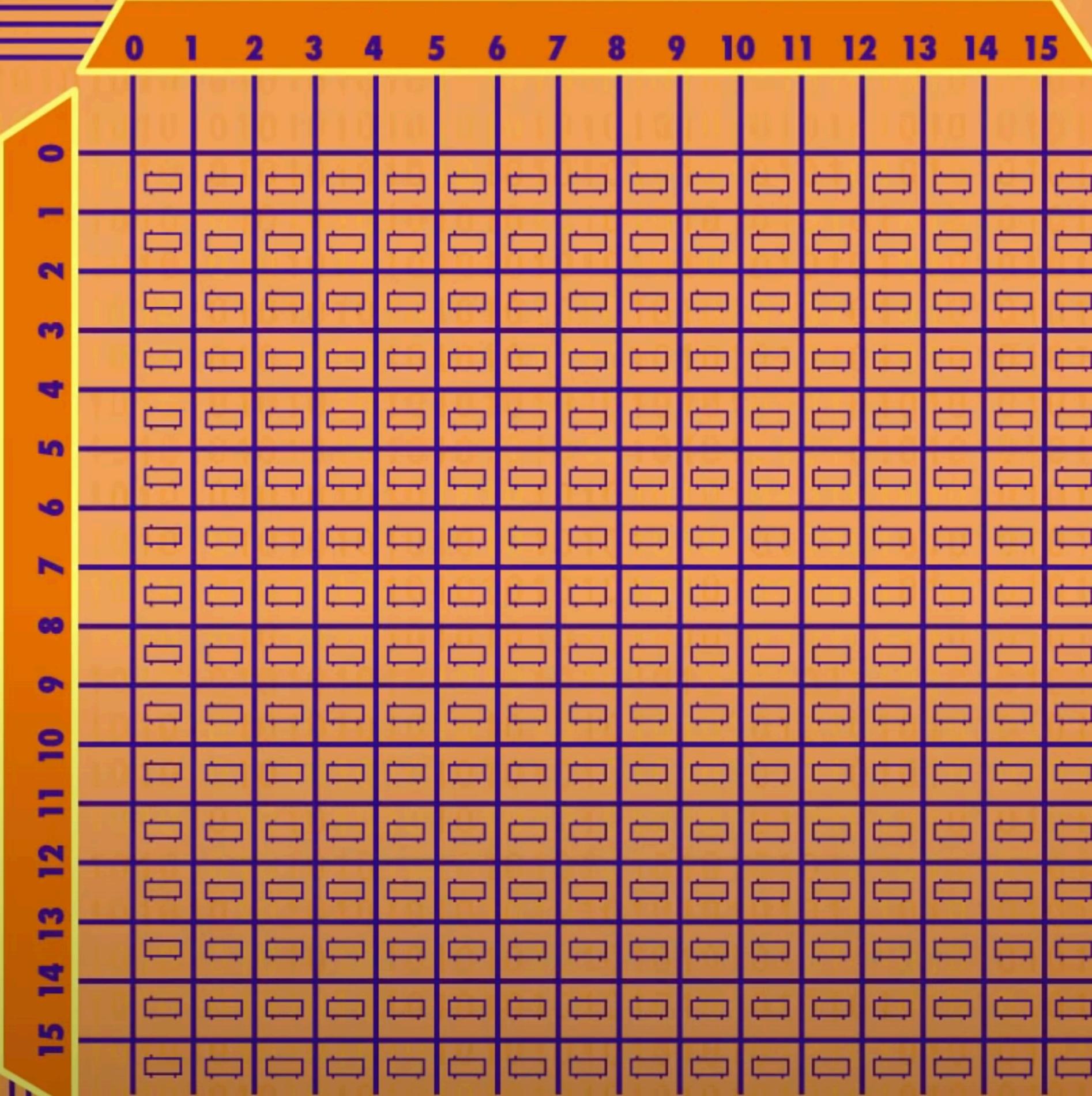
AND-OR LATCH

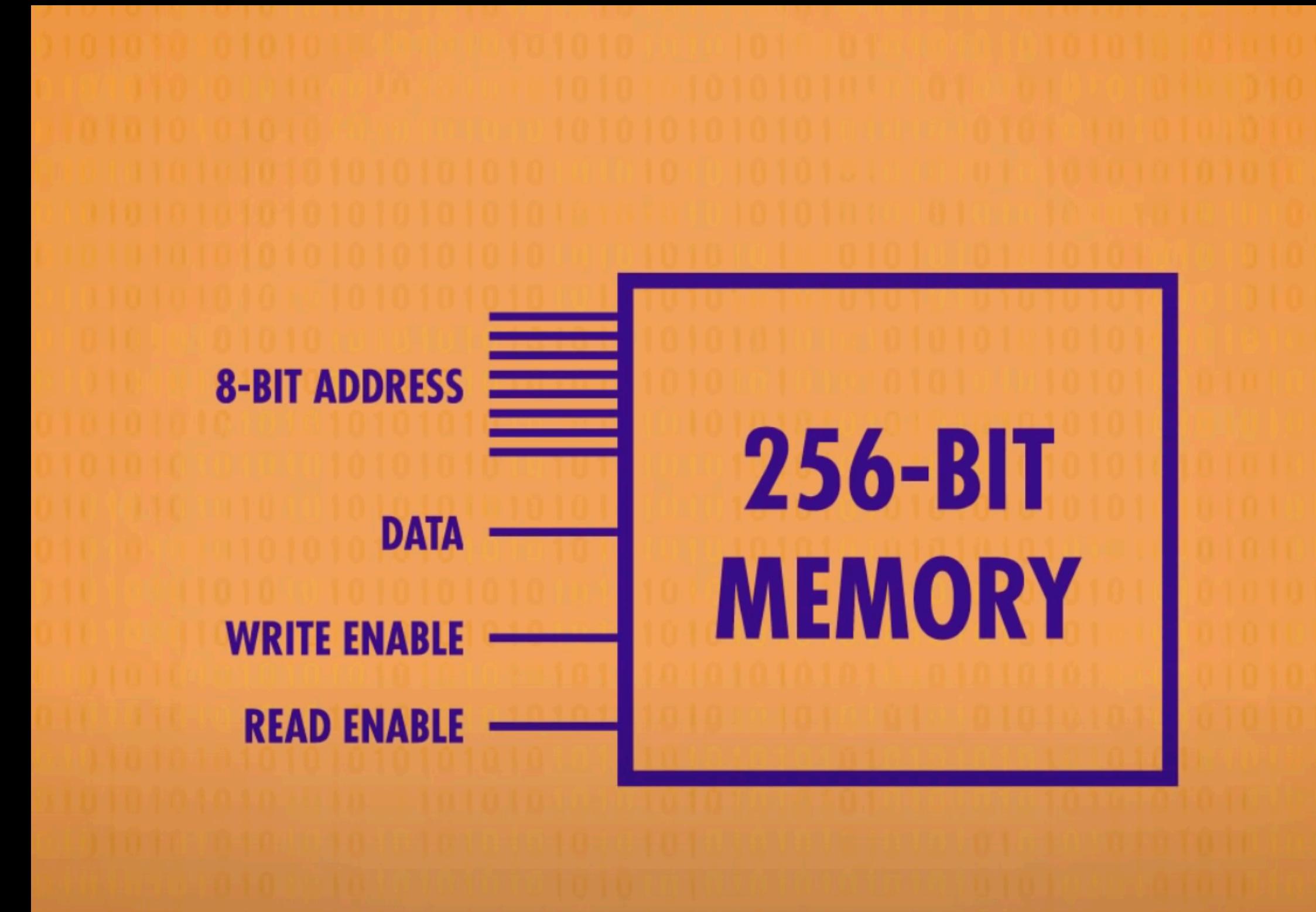


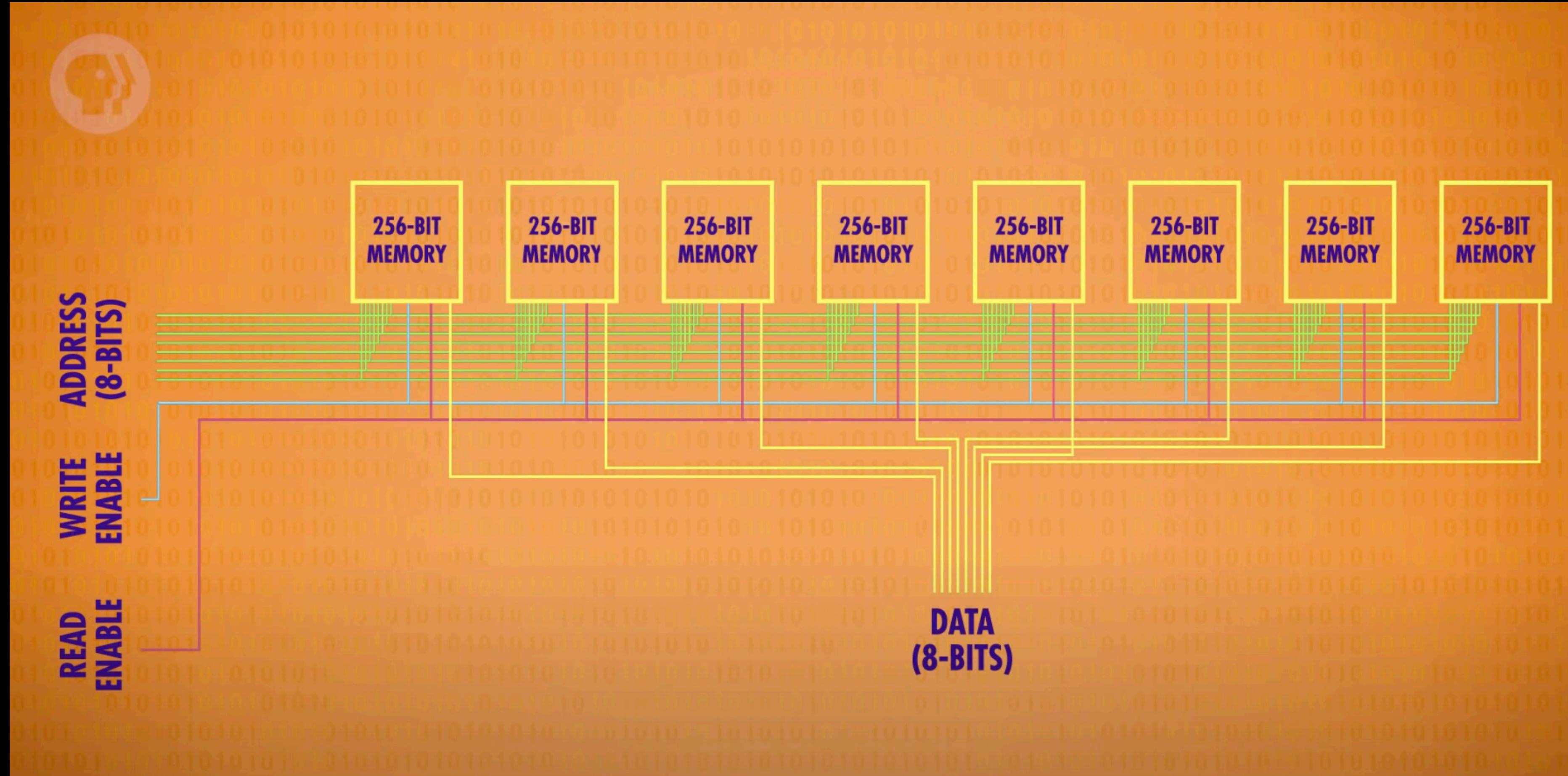
MULTIPLEXER

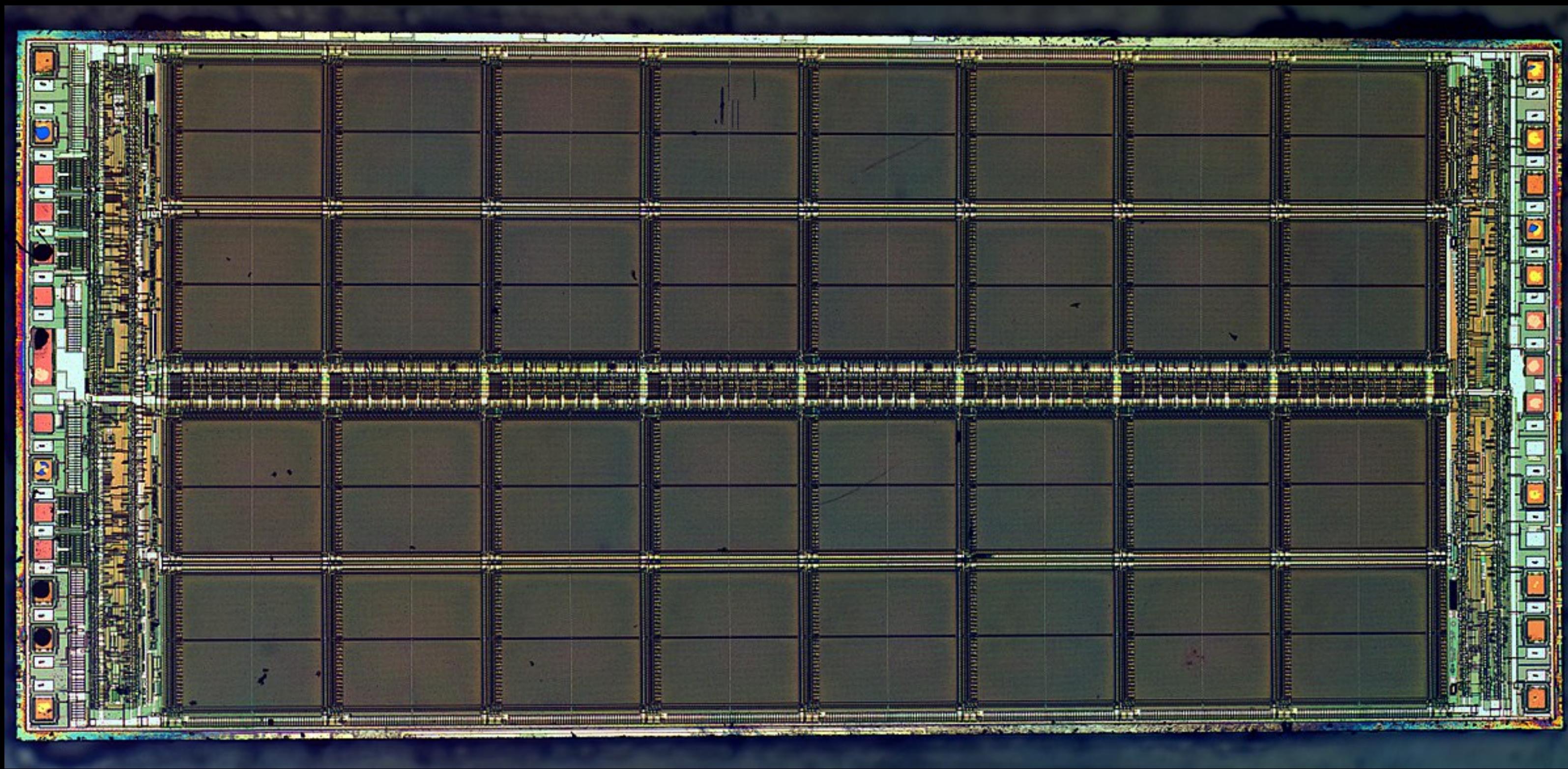
COLUMN
ADDRESS
(4 BITS)

ROW
ADDRESS
(4 BITS)



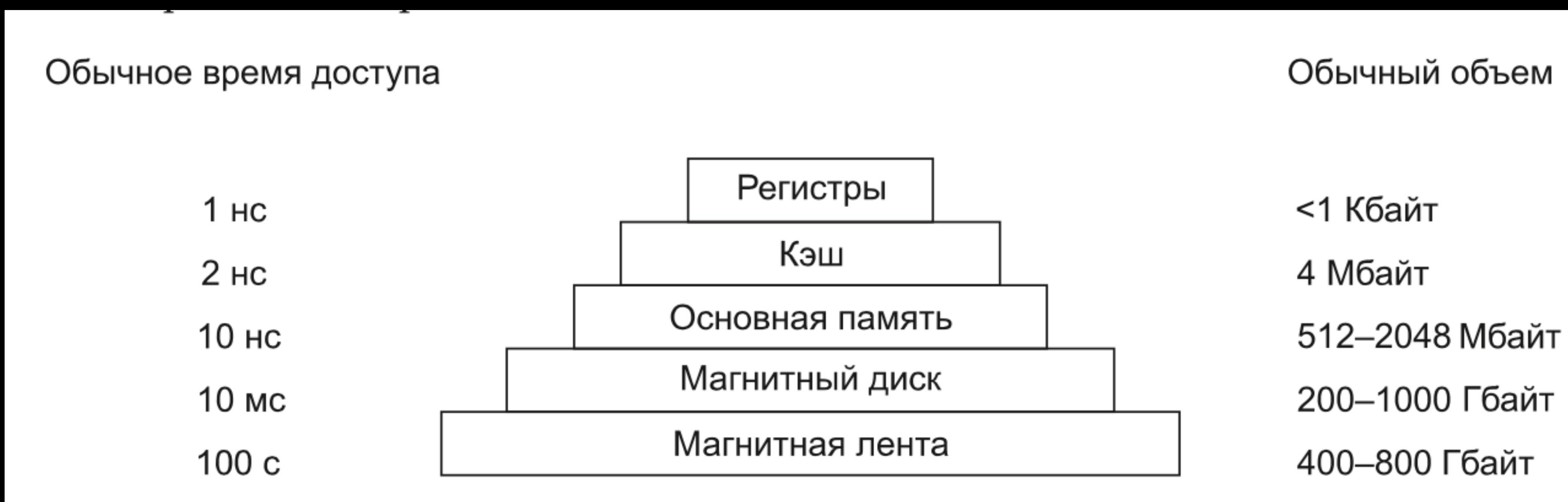






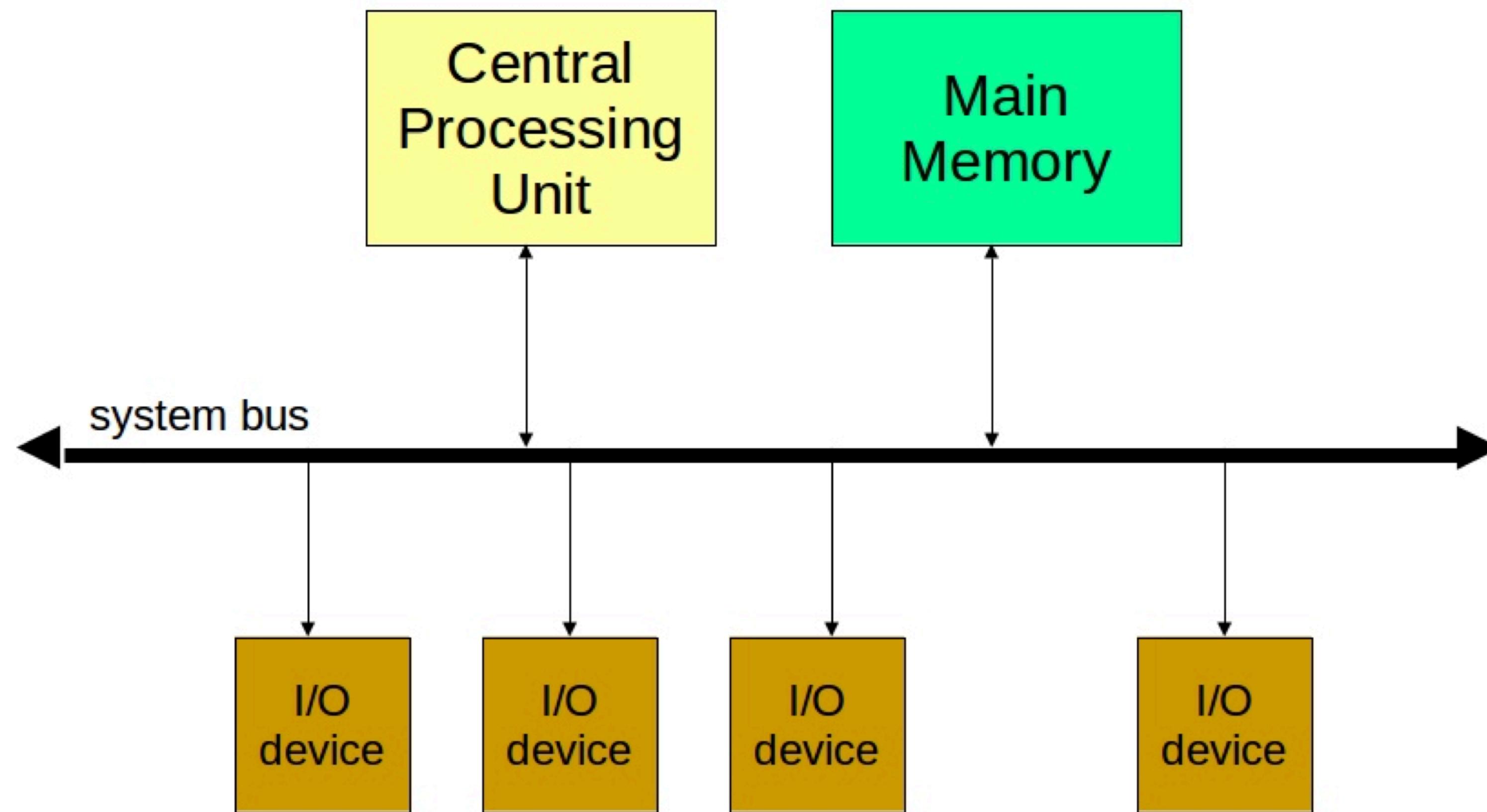
Память

- Регистр
- Кэш
- RAM/ROM
- HDD, SSD
- Другие...



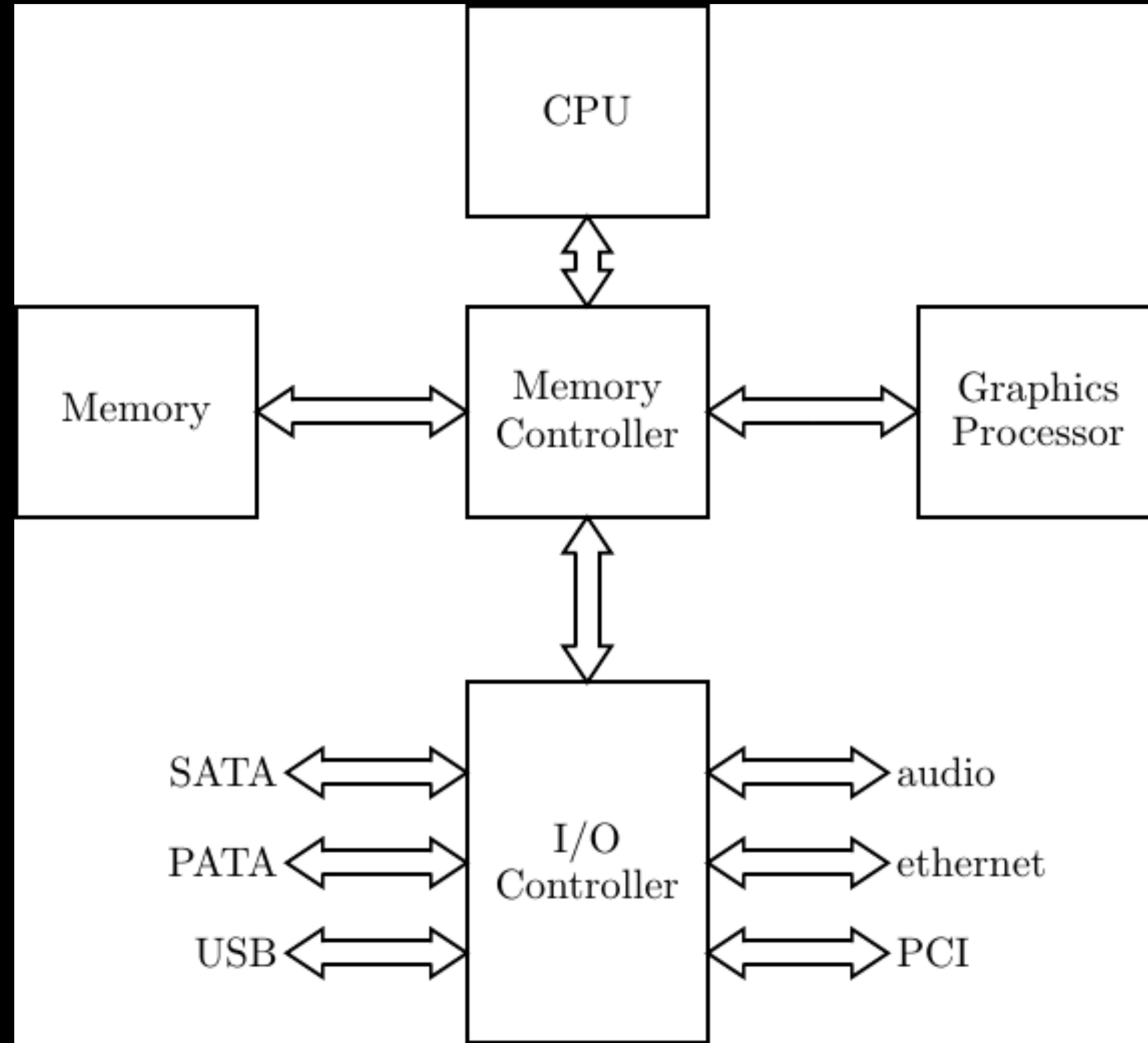
I/O

- Процессор должен уметь работать с сигналом.
- У устройства должен быть свой приоритет.
- Устройство должно уметь работать с процессором.



Прерывания (Interruptions)

- Аппаратные/Hardware.
 - PIC (Programmable Interrupt controller)
- Программные/Software.
 - Исключения: abort, trap, fault
 - Программируемые.
- X86 процессор имеет 256 разных типов прерываний. 32 из которых зарезервированы.
 - 0: divide-overflow fault
 - 6: undefined opcode
 - 11: Segment-not-present fault

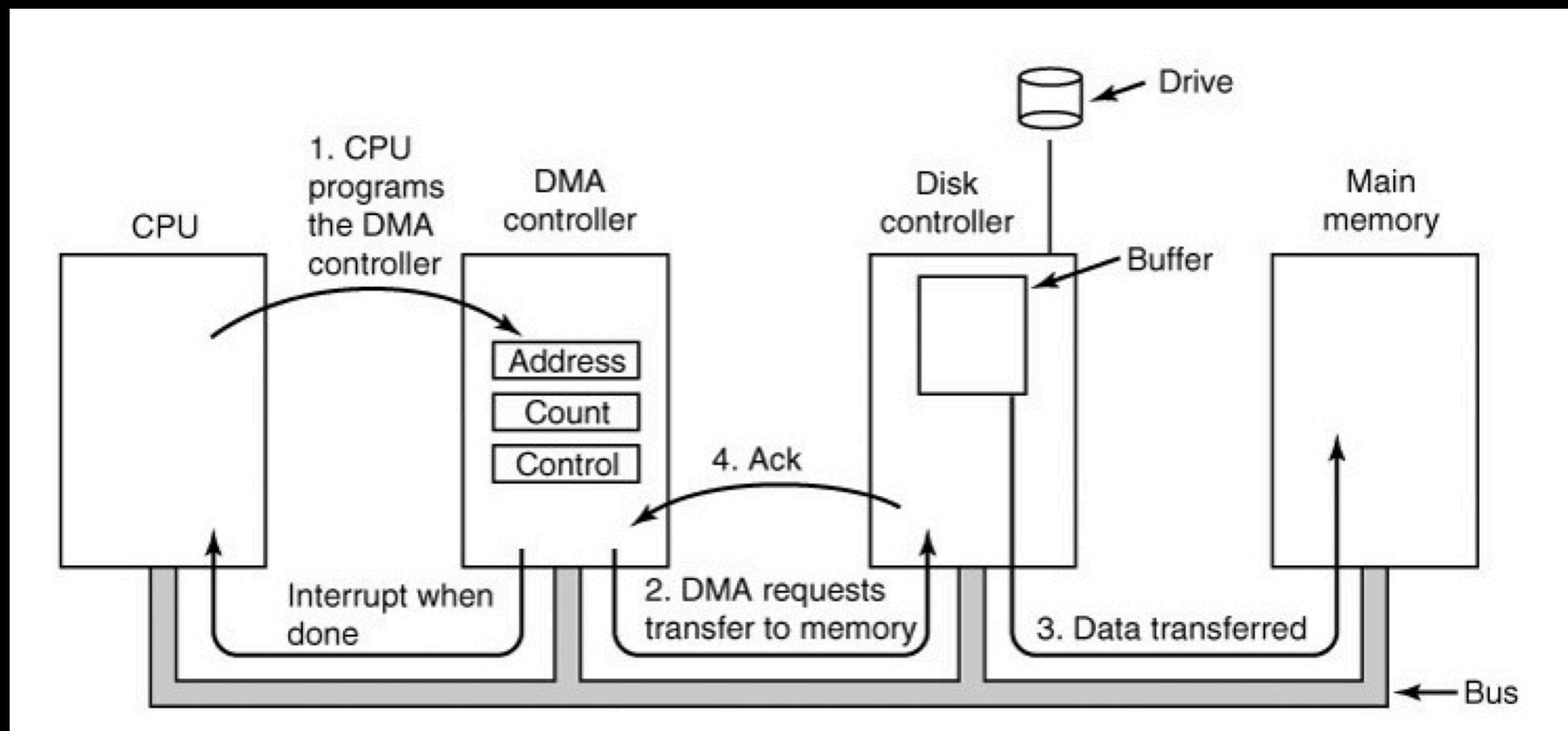


Контроллер

- Каждое устройство имеет свой контроллер.
- Имеет схему которая коммуницирует с устройством и процессором.
- Декодирует сигнал от устройств в формат, понимаемый процессором.
- Буферилизирует данные от устройств.

Взаимодействие процессора

- PIO (Programmed I/O) - ввод-вывод через процессор.
- DMA (Direct Memory Access) - прямой доступ к памяти.



Драйвер

- Запускается ОС или пользователем.
- Преобразует данные, которые будут понятны контроллеры.
- Бывают символичные, блочные, сетевые.

```
#include<linux/module.h>
#include<linux/init.h>
#include<linux/kernel.h>

static int hello_init(void){
    printk(KERN_ALERT "Hello World\n");
    return 0;
}

static void hello_exit(void){
    printk(KERN_INFO "Goodbye cruel world\n");
}

module_init(hello_init);
module_exit(hello_exit);

MODULE_AUTHOR("Abylaikhan Zulbukharov");
MODULE_LICENSE("MIT");
MODULE_DESCRIPTION("Simple hello world driver.");
```