
EECE 276

Embedded Systems

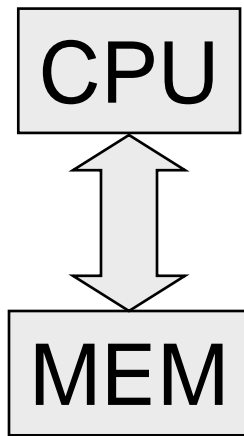
Performance enhancements

Other devices

Non-von-Neumann machines

High-performance Processing

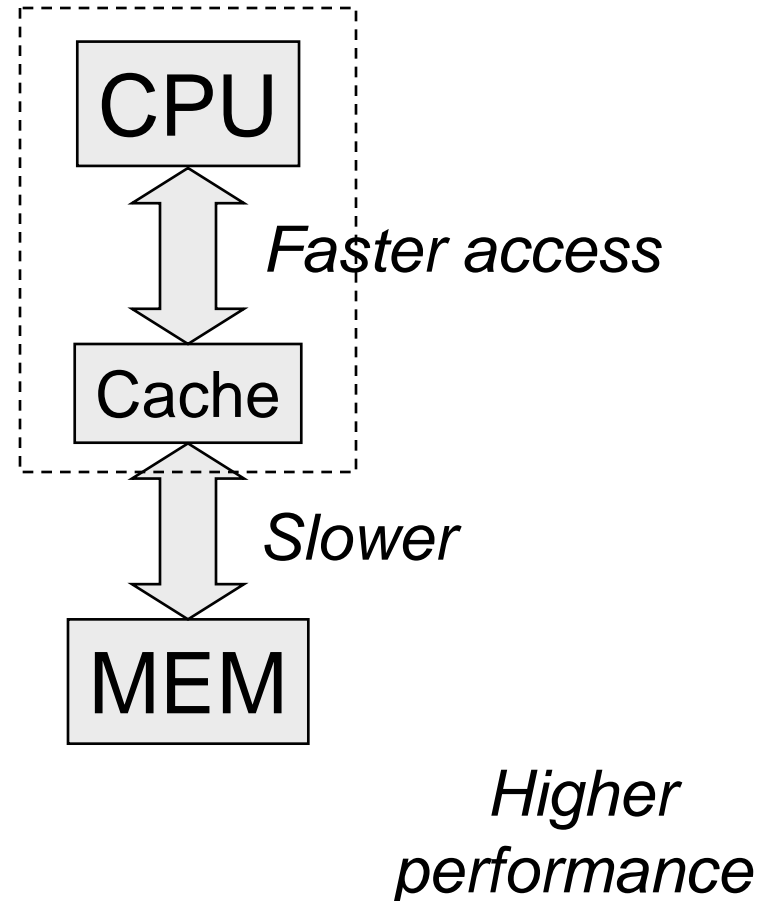
Cache memory



Runs at the speed of the slower.

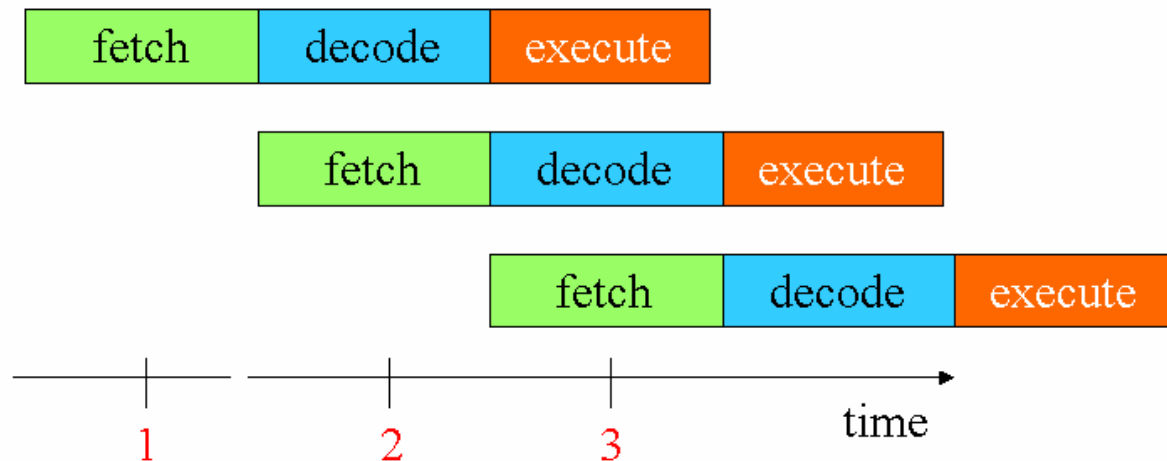
Locality principle:

If many memory references occur in the same “neighborhood”, then keeping that page in high-speed memory will improve performance.



High-performance Processing

Pipelining



Fetch/Decode/Execute stages are done by independent units -> their function overlaps in time.

High-performance Processing

DSP Architectures:

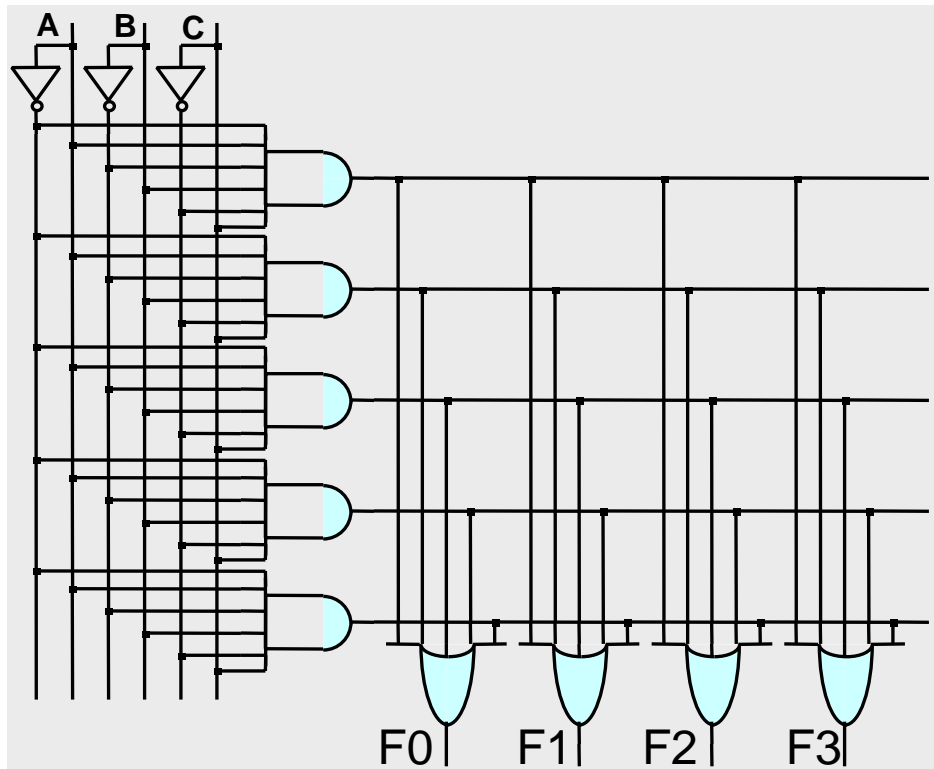
- Traditional microprocessor with support for high-speed DSP-oriented instructions:
 - » “Multiply-Accumulate” instruction
- High-speed communication ports

RISC: Reduced Instruction Set Computer

- Simple Load/Store Instructions, 1-cycle
- Many registers, pipelined design
- Complex compiler

Special Devices

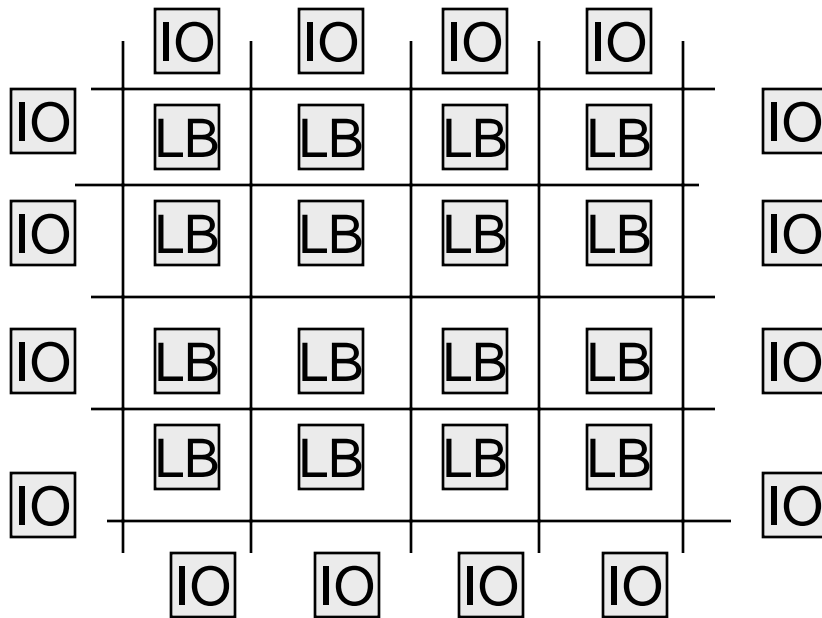
Programmable Array Logic



- AND-array: products
- OR-array: sums
- “Programmable” connections

Special Devices

Field Programmable Gate Array



Flexible “fabric” to implement arbitrary digital circuits

- I/O Blocks
- Logic Blocks
 - LUTs, memory
- Interconnects

Non-von-Neumann Systems

Classic: Single Instruction/Single Data

Parallel systems:

- Multiple Instructions/Single Data
 - » Pipelined architectures
 - » Very Long Instruction Word architectures
- Single Instruction/Multiple Data
 - » Systolic arrays – all elements perform the same operation
- Multiple Instructions/Multiple Data
 - » Full multi-processing
 - » Dataflow architectures
 - » Transputers, DSP networks: high-speed comm. ports
 - » Time-triggered architecture: time-shared bus – fault tolerant

Non-von-Neumann Systems

Examples:

- **MISD:**
 - » VLIW: TransMeta Processor. X86 instructions are transcribed on-the-fly into VLIW instructions
- **SIMD:**
 - » Systolic arrays for real-time image processing
- **MIMD:**
 - » Multi-processor servers: multiple CPUs, shared bus

Non-von-Neumann Systems

Time-Triggered Architecture

- Time-shared, scheduled bus
- Communication Network Interface
- Fault-tolerant clock synchronization protocol

All tasks and communications are strictly scheduled at design time. Nodes must exhibit “fault-silent” behavior.

