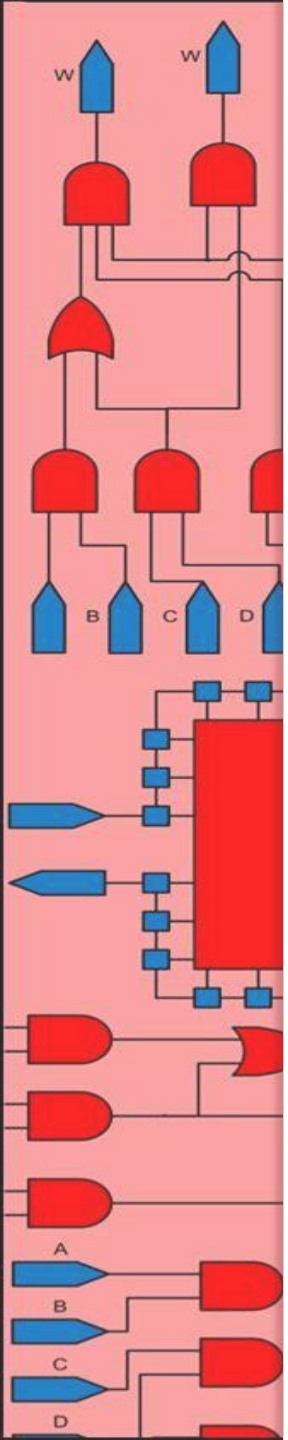
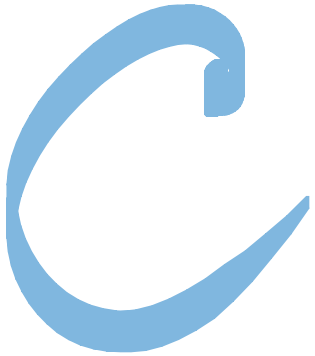


Course on Details of Hardware of Processors (Processors)

VHDL Design and Implementation of an Instruction Set Architecture

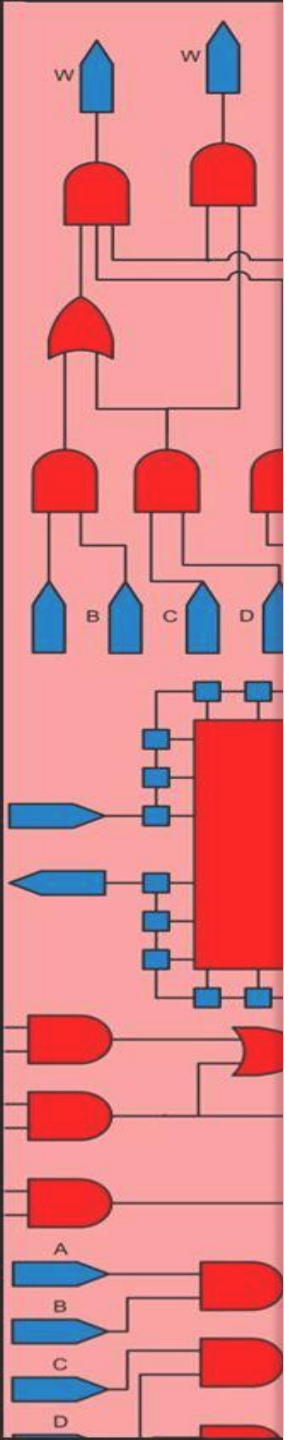


Zain Navabi

Slides prepared by Katayoon Basharkhah

cini Cybersecurity National Lab

CYBER CHALLENGE.IT



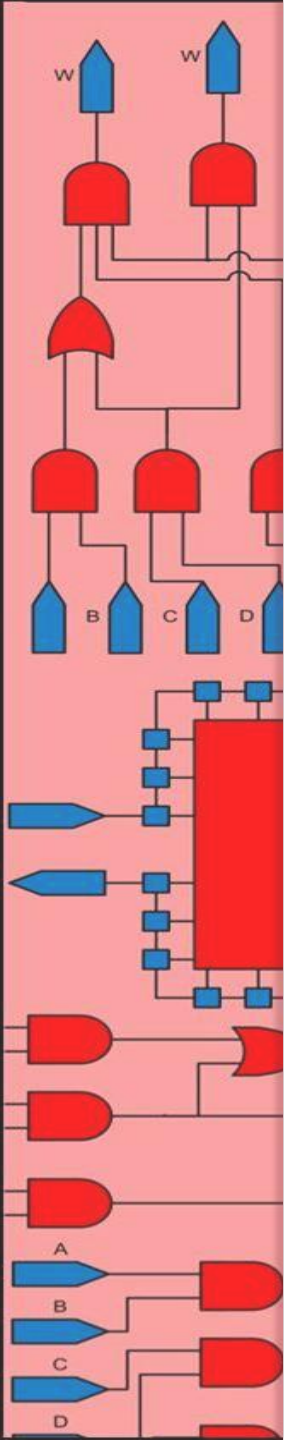
Topic 3

Processor Devices

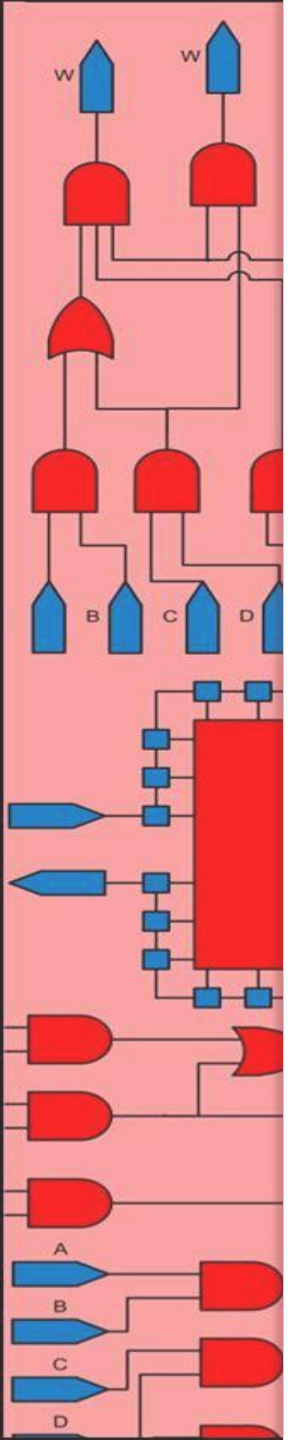
Zain Navabi

Slides prepared by Katayoon Basharkhah

Course Roadmap



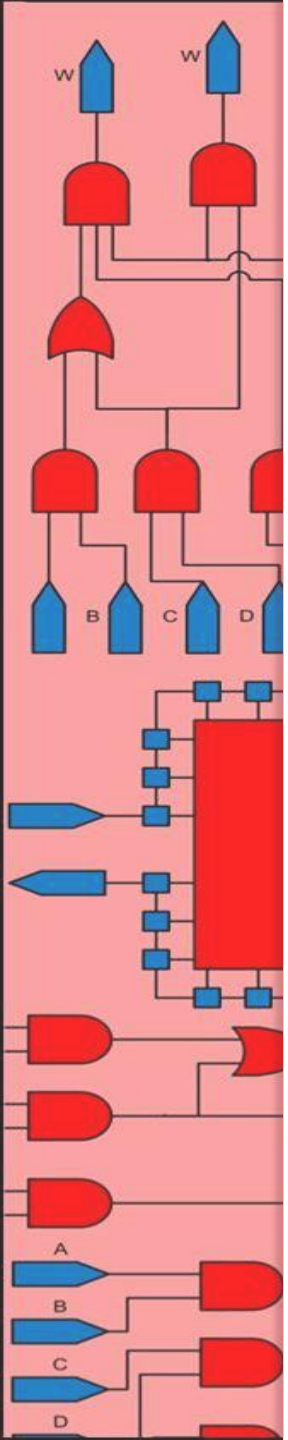
	Topic 1	Topic2	Topic 3	Topic 4
Goals	A Processing System	Memory Model	Processor Devices	Processor Architecture
	Embedded System Introduction	Peripheral Hardware		Show Case ISA
Contents	•CPU •Instructions •Memory-Map •Memory •I/O device	Different memory structures Memory mapping in the bussing and devices' communications.		• SAYAC Instructions • Datapath Design • Controller Design • Simulation and FPGA Synhesis
	Introduction to processor-based system, general concepts and different components of the system	•SDRAM •ROM •SRAM	• Address Logic • Data Registers • Handshaking • Interrupt	A sample processor named SAYAC. Processor internal architecture starting form the processor instruction set and coming down to the all RTL components of the processor datapath and controller.



Processor Devices

Learning Outcomes:

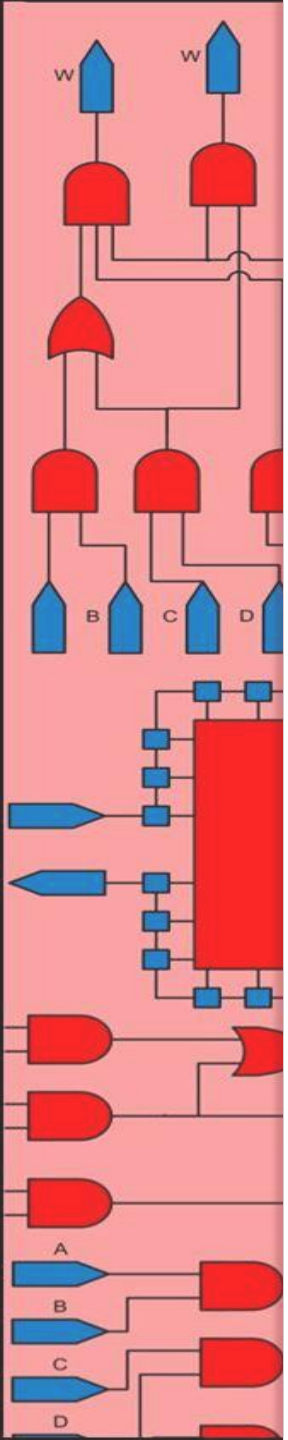
- *Learn* I/O device structure
- *Learn* memory mapped I/O
- *Learn* Bus interface
- *Learn* address decoding



Processor Devices

Outline:

- I/O Devices
- Processor-I/O Communication
- Memory Mapped I/O
- Bussing
- Address Decoding



Processor Devices

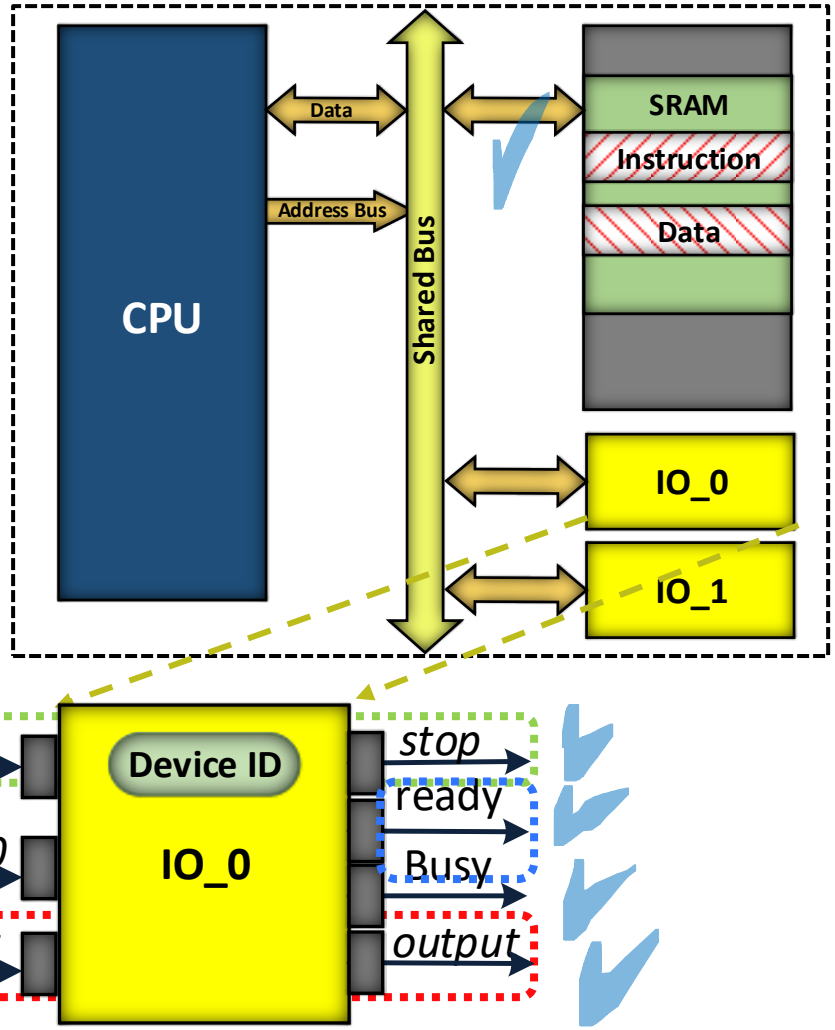
I/O Devices

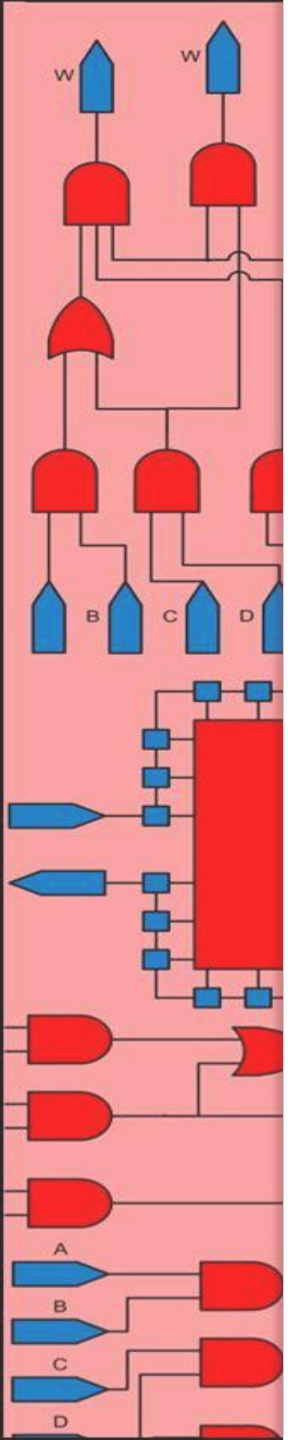
- Storage devices (disk, tapes)
- Transmission devices (network card, modem, serial)
- Human interface devices (screen, keyboard, mouse)
- Process control (start/stop, set device, get status)
- Data IO (A/D, D/A)

Processor Devices

I/O Devices

- Device ID
- Status signals:
  - Device busy, error, completed last task, etc.
- Control signals:
  - start/stop the device, change mode, start reading, start writing, etc.
- Inputs
- Outputs





Processor Devices

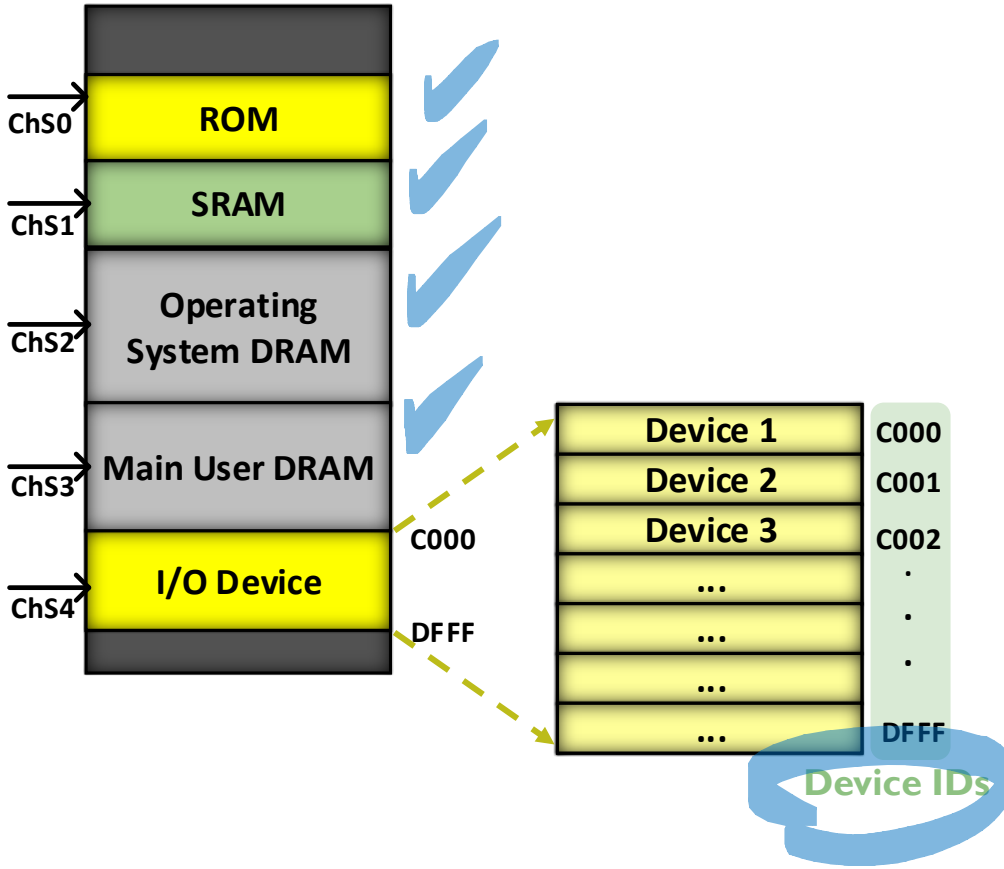
Processor interacting with I/O Devices

- Distinct input and output instructions
- Instructions use device ID as address
- The CPU addresses the device
- The CPU enables the I/O read or I/O write
- The IO device will respond accordingly

Processor Devices

Memory-Mapped I/O

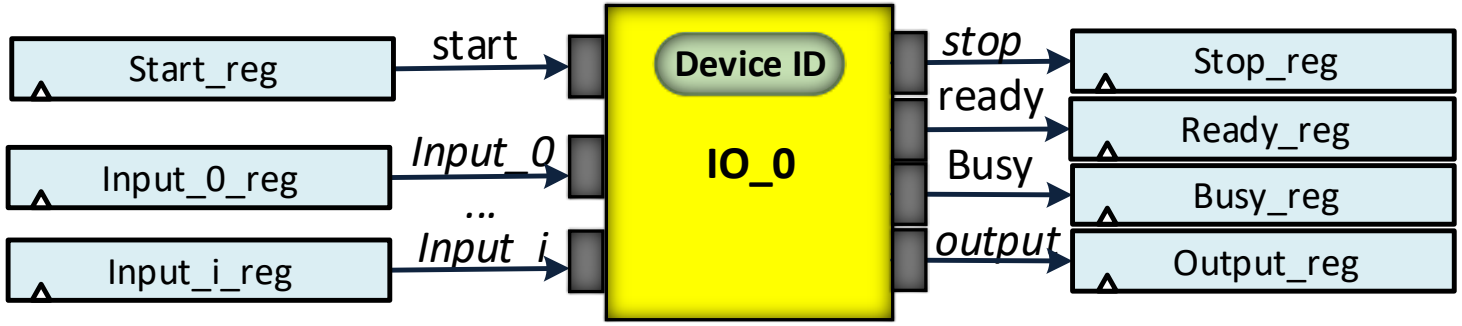
- The memory mapped I/O uses the same address space for both memory and I/O
- In this case device IDs look like memory addresses and occupy part of the memory address space



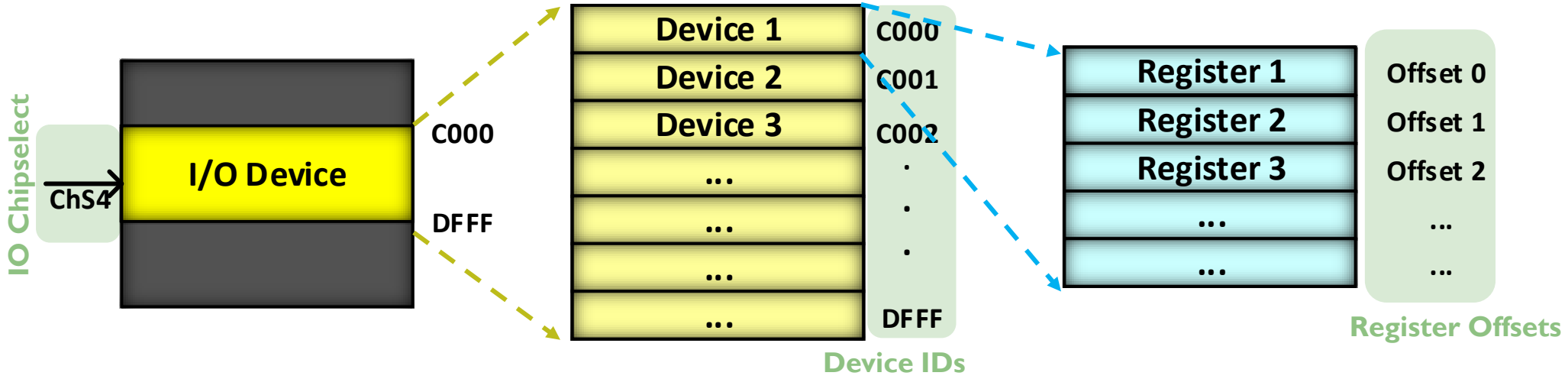
Processor Devices

Memory-Mapped I/O

- Dedicating one interface register for each port of I/O device

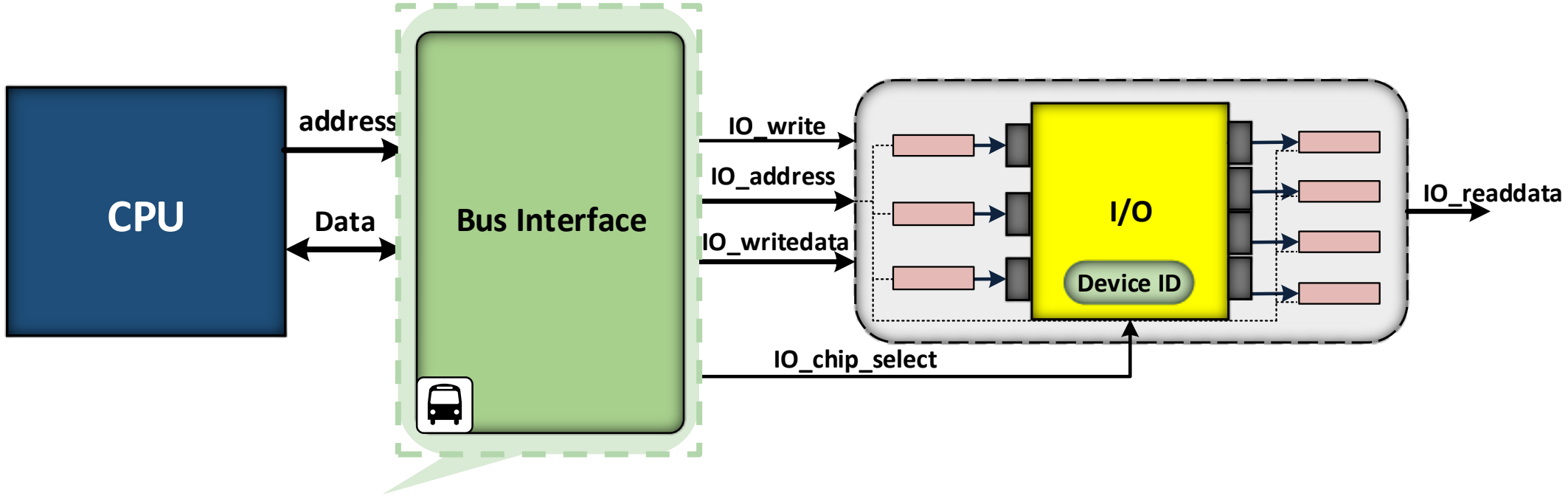


- Register mapping by decoding the corresponding I/O device ID (memory-mapped address)



Processor Devices

Bussing

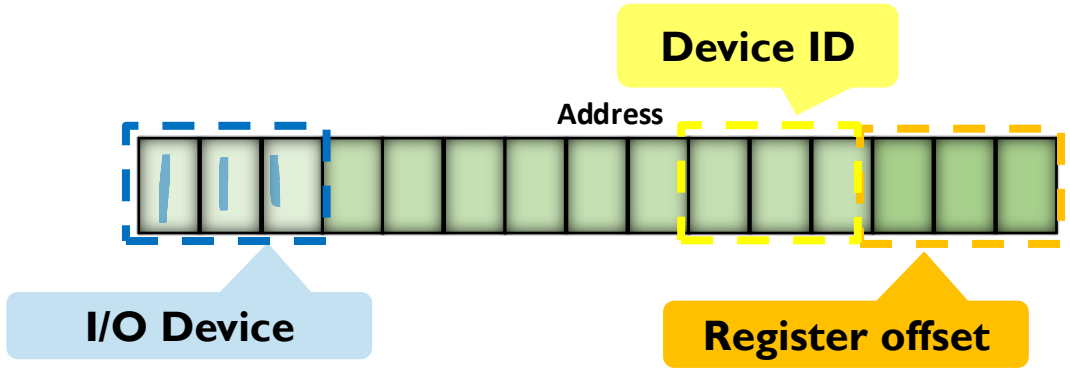


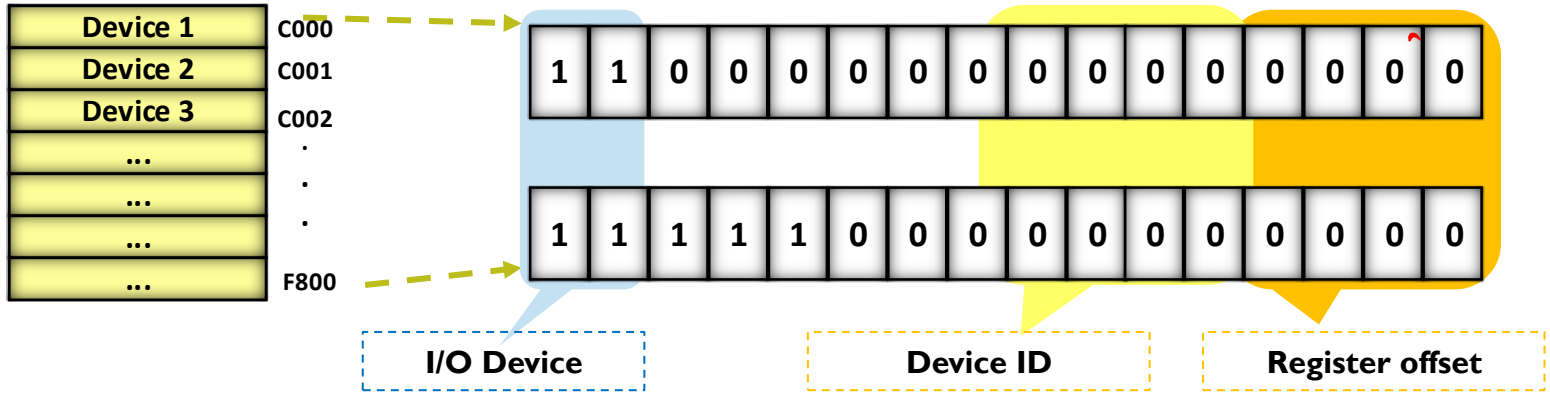
- Converts processor address lines to I/O interface signals
- Address decoding

Processor Devices

Address decoding

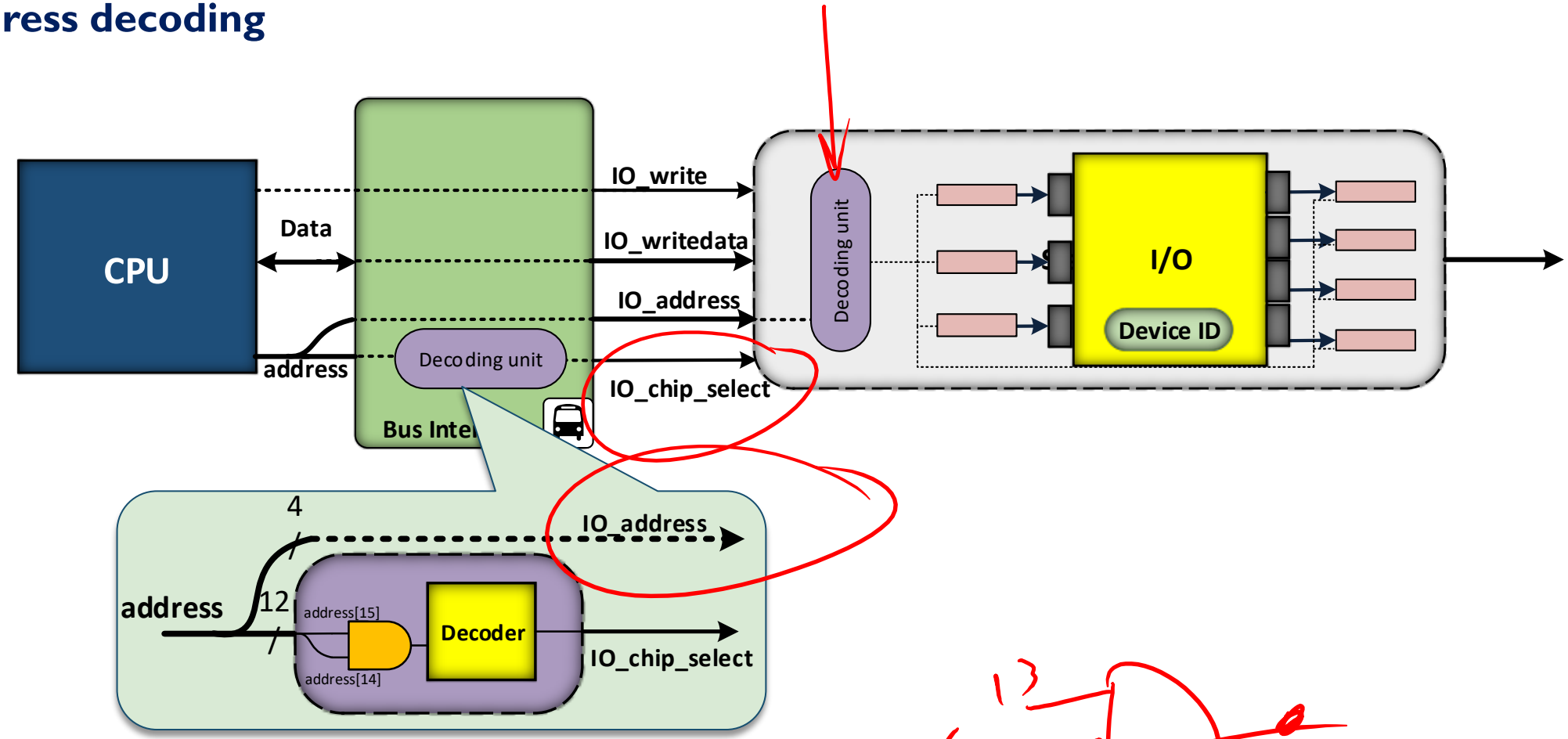
- Use lower order bits for a specific device and specific registers within that device.
- Use higher order bits for enabling specific device

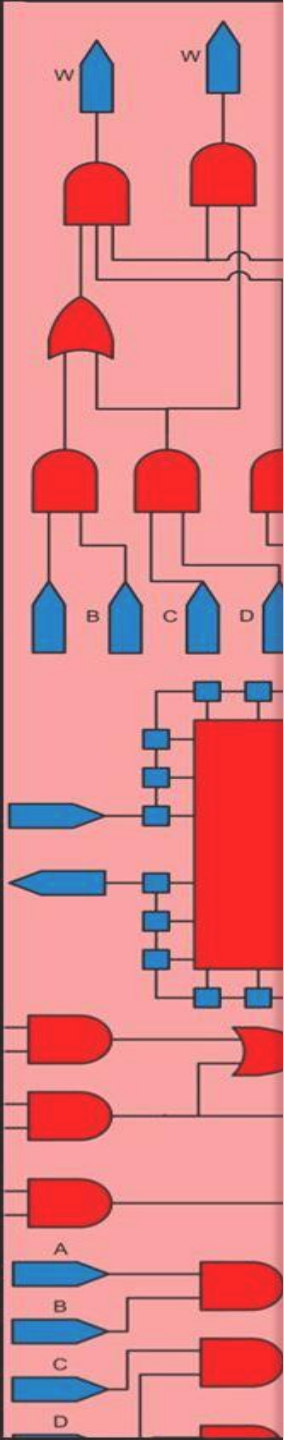




Processor Devices

Address decoding





Conclusion

In this topic we have covered:

- I/O Devices
- Processor-I/O Communication
- Memory Mapped I/O
- Bussing
- Address Decoding