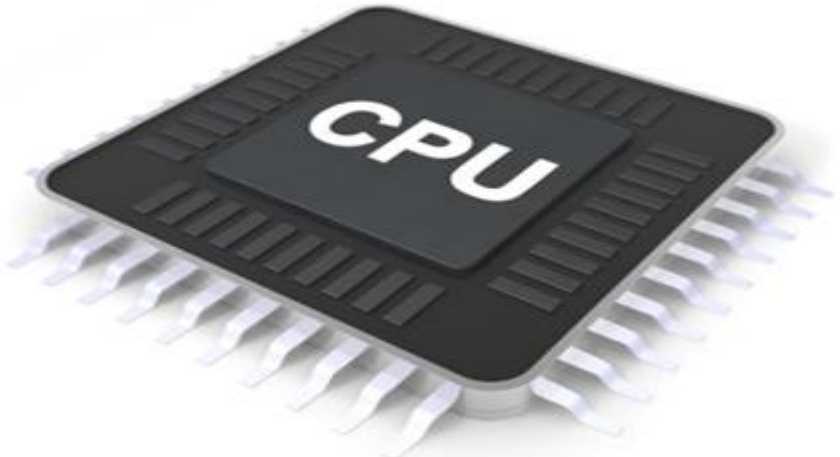
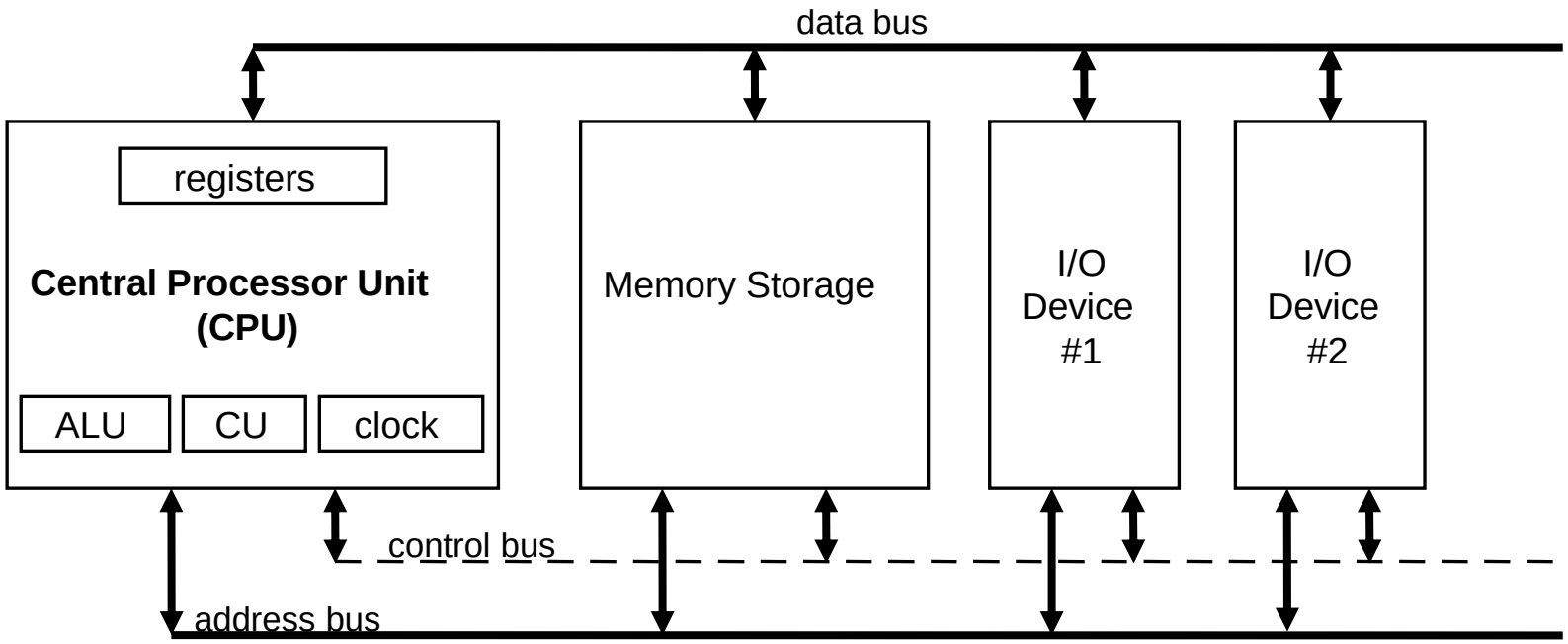


Intro to AVR Architecture

ENEE 3582

Microp

Microcomputer Design



Main Components

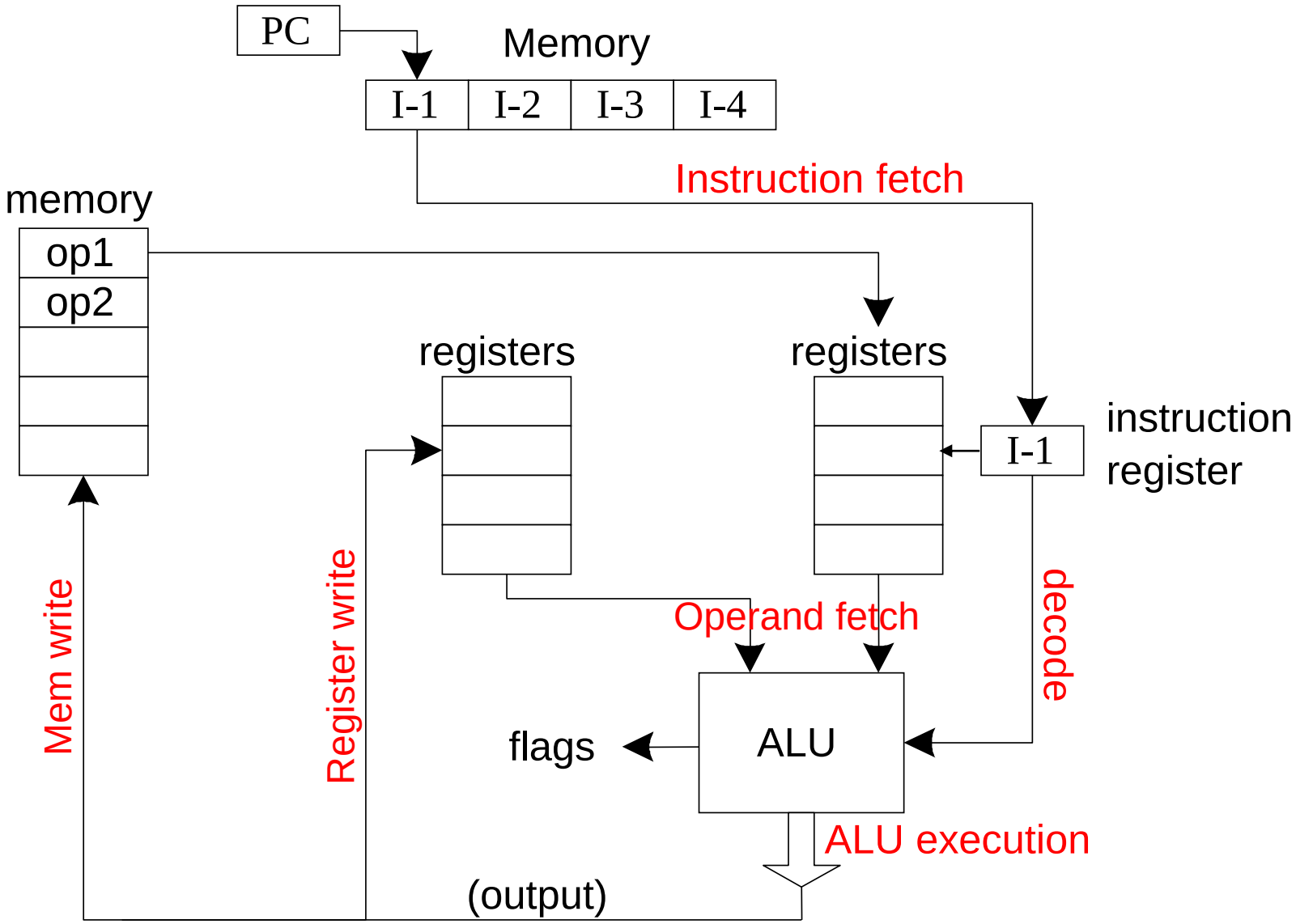
- Clock
  - clock is used to trigger events
  - synchronizes CPU and bus operations
  - machine (clock) cycle measures time of a single operation
- Control unit (CU)
  - Logic that coordinates sequence of execution steps
- ALU performs
  - arithmetic and
  - logical processing
- Registers
  - store data for processing.

Clock

- Memory
  - store data
  - Stores code
- Input/Output (I/O)
  - communication with the “real” world
- System bus
  - Data lines: for transmitting data
  - Address lines: indicate where information is sent to/from
  - Control lines: regulate activity on the bus

Execution Cycle

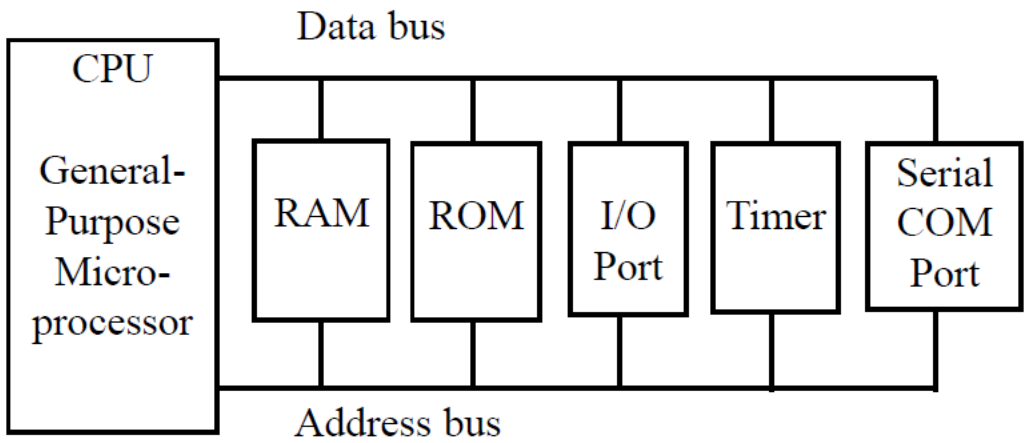
- The high-level programming instruction are converted into machine language
  - Executable code
- Code is first stored on a mass storage device (disk).
- Program code are transferred from disk to RAM
- CPU executes code in stages:
  1. Instruction fetch: Read instruction into CPU from RAM (or cache)
  2. Instruction decode: Logic decode of binary
  3. Operand fetch: Values needed
  4. ALU operation
  5. Memory write: Update memory
  6. Register write: Update registers


PC
Memory
I-1
I-2
I-3
I-4
Instruction fetch
memory
op1
op2
registers
registers
I-1
instruction
register
Mem write
Register write
Operand fetch
decode
flags
ALU
ALU execution
(output)

General Purpose vs Embedded

- Microprocessors in computers
- Processes information
- Runs many apps
- Software and OS
- Newer (started in 1980)
- Lots of RAM
- Many pins
- Power hungry
- Non-RT apps
- Advanced, expensive, large die
- Multicore
- Multiple human interfaces
- Not typically rated

- Microcontrollers for devices
- Controls mech/elect/electronic
- Runs single/limited apps
- Firmware, no OS
- Older (1970)
- Tiny RAM
- Limited pins
- Low power
- RT apps
- Simple, cost-effective, small die
- Peripheral support, System-on-a-chip(SoC)
- No/limited human interface
- Rated for temp/pressure/vibration



CPU	RAM	ROM
I/O ----- ADC	Timer	Serial COM Port

A. Mazidi

Choices of Microcontrollers

- Choose a microcontroller based on:
  - Speed/computation power
  - Peripheral support (sub-systems in SoC)
  - Power consumption
  - RAM/ROM needs
  - Number of IO pins needed
    - Packaging: DIP vs QFP
    - https://www.eesemi.com/ic-package-types.htm
  - Cost

Choices of Microcontrollers

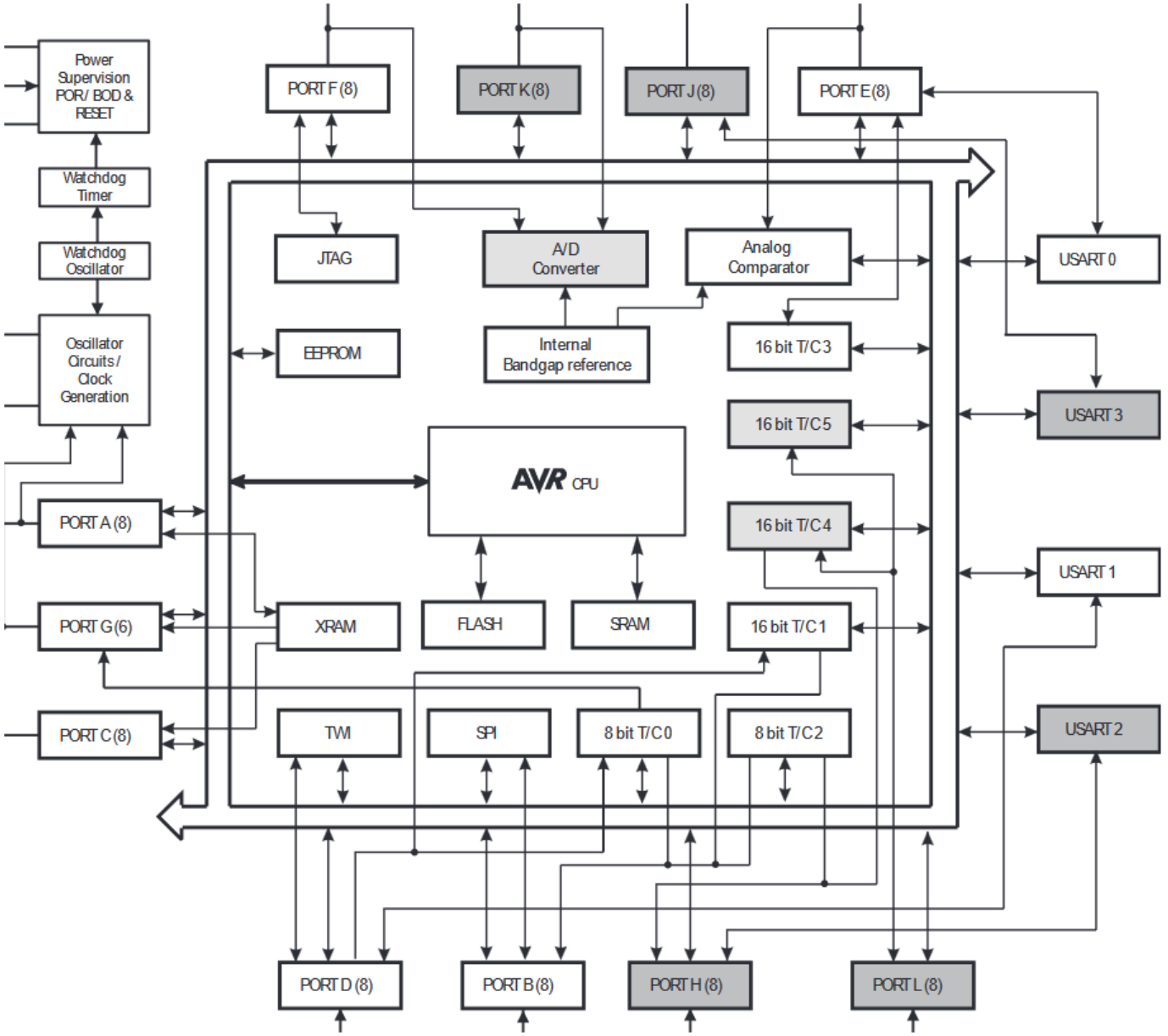
- 4-bit: E.g. COP400, EM73201, W741E260, HD404358
- 8-bit: E.g. 6502, Z80, 8051, AVR, PIC
  - A few bytes to a few hundred KB of RAM
  - Software is in asm and C
  - Still dominate both numbers and dollar volume
  - Two kinds:
    - Old-style CISC, E.g. 6502, Z80, 8051
      - These are >20 years old and doing well
    - Newer style RISC, E.g. AVR, PIC
- 16-bit: E.g. ARM, MIPS, MN10300, PPC
- 32-bit: E.g. 386, AVR32

AVR

- Designed by two students of Norwegian Institute of Technology (NTH)
  - Bought and developed by Atmel in 1996. Atmel bought by Microchip in 2016
- AVR 8-bit family:
  - Classic AVR (AT90Sxxxx)
    - Replaced. Not recommended for new design
  - Mega AVR (ATmegaxxxx)
    - 120 instructions, expandable
    - Extensive peripheral sets
  - Tiny AVR (ATtinyxxxx)
    - Smaller, limited, low power, low cost
  - Special AVR
    - Unique peripherals: USB, ethernet, Zigbee, CAN
- AVR32: 32bit

ATmega 2560

- Datasheets: https://www.microchip.com/en-us/product/ATmega2560#document-table
- AVR CPU Core:
  - 32, 8-bit registers: R0-R31
  - 3, 16-bit indexing registers: X, Y, Z
- 4KB EEPROM, 8KB RAM
- Data registers: address 0x0000-0x001F
- Internal data RAM (IRAM): address 0x0200-0x21FF
- External data RAM (XRAM): address 0x2200-0xFFFF
- EEPROM: address 0x0000
- FLASH address 0x000000-



https://www.microchip.com/en-us/product/ATmega2560#document-table

Device	Flash	EEPROM	RAM	General Purpose I/O pins	16 bits resolution PWM channels	Serial USARTs	ADC Channels
ATmega640	64KB	4KB	8KB	86	12	4	16
ATmega1280	128KB	4KB	8KB	86	12	4	16
ATmega1281	128KB	4KB	8KB	54	6	2	8
ATmega2560	256KB	4KB	8KB	86	12	4	16
ATmega2561	256KB	4KB	8KB	54	6	2	8

Speed (MHz)[2]	Power Supply	Ordering Code	Package[1][3]	Operation Range
8	1.8V - 5.5V	ATmega2560V-8AU ATmega2560V-8AUR[4] ATmega2560V-8CU ATmega2560V-8CUR[4]	100A 100A 100C1 100C1	Industrial (-40°C to 85°C)
16	4.5V - 5.5V	ATmega2560-16AU ATmega2560-16AUR[4] ATmega2560-16CU ATmega2560-16CUR[4]	100A 100A 100C1 100C1	

Special Purpose Registers

- PC: program counter
  - Address of next instruction
  - 17-bit: 2^{17} locations
  - Can only be modified using jump instructions
- IR: instruction register
  - Instruction currently executing
- SP: 16-bit stack pointer
  - Used to point to the top of the stack
  - Can be set and modified
- SREG : Status register, 8-bit flag
  - Reports on output of certain instructions
  - Used to control operations

PC
IR
SP
SREG

R0	
R1	
…	
R25	
R26	X_L
R27	X_H
R28	Y_L
R29	Y_H
R30	Z_L
R31	Z_H

Registers

- Can read and write into these registers
- 32, 8-bit, General purpose registers (GPR):
  - R0..R31
  - Contain operands for ALU, output of ALU
  - Used as storage for data from mem, to mem
- 3, 16-bit, Indexing registers
  - X, Y, Z
  - Combines 2,8-bit GPRs
    - H:high byte, L: low byte
    - Be careful not to use respective registers when indexing
  - Used to create addresses
  - Address=16 => 2^{16} locations

PC
IR
SP
SREG

R0	
R1	
…	
R25	
R26	X_L
R27	X_H
R28	Y_L
R29	Y_H
R30	Z_L
R31	Z_H