EE-222: Microprocessor Systems

AVR Microcontroller:
Pipelining
&
Instruction Cycle Time

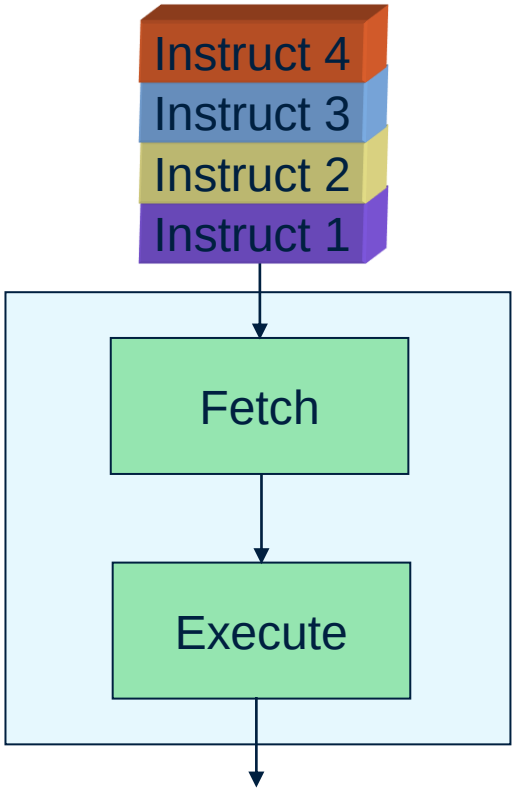
Instructor: Dr. Arbab Latif

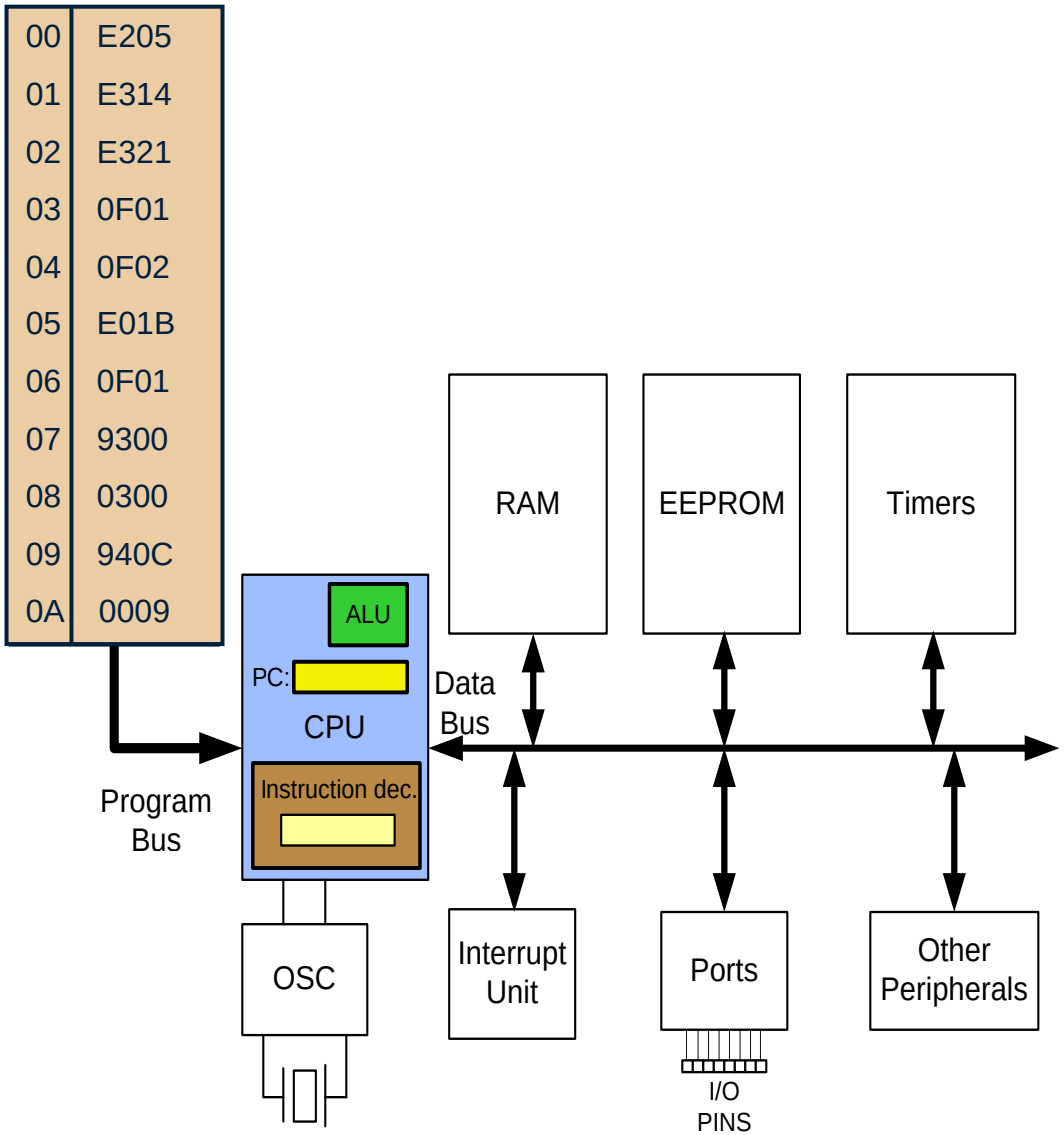
NUST
SCHOOL OF ELECTRICAL ENGINEERING &
COMPUTER SCIENCE (SEECS)

Pipelining

Fetch and execute

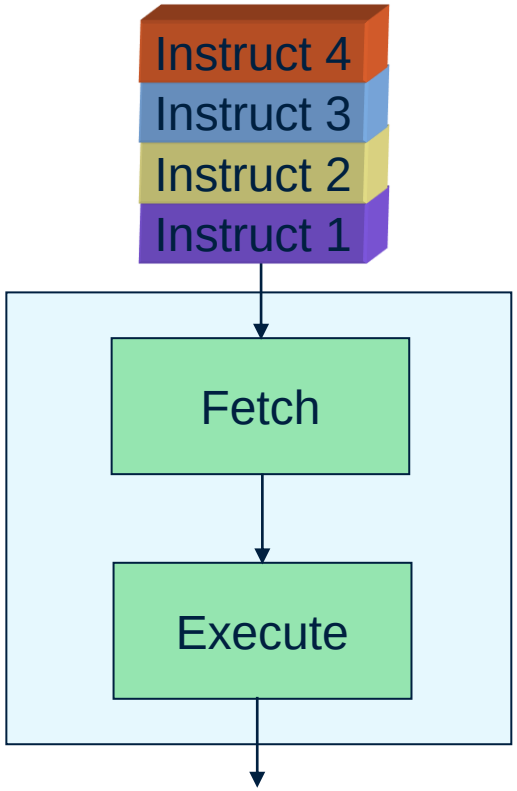
- Old Architectures

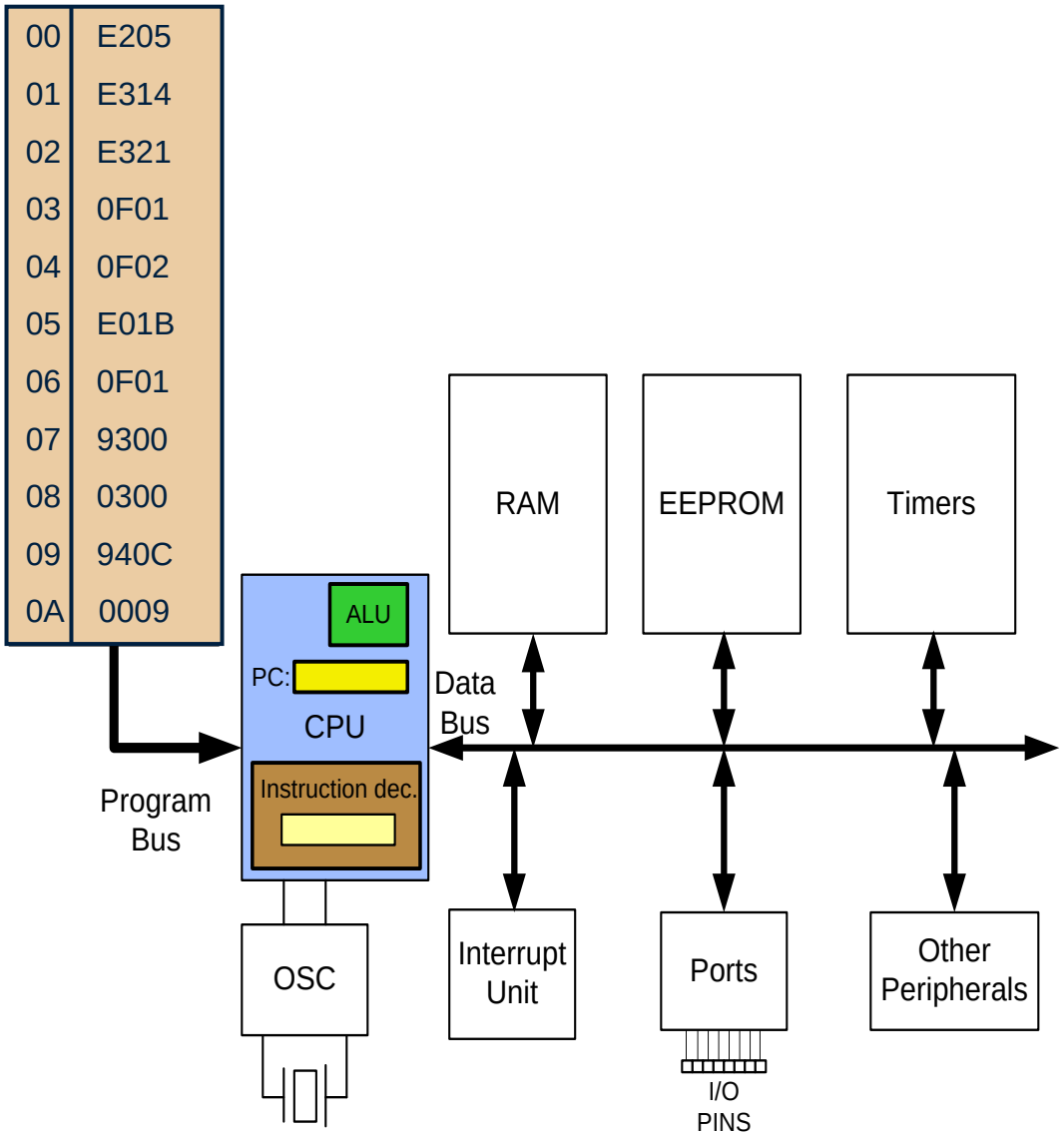




Pipelining

- Pipelining





Instruction Cycle Time for the AVR

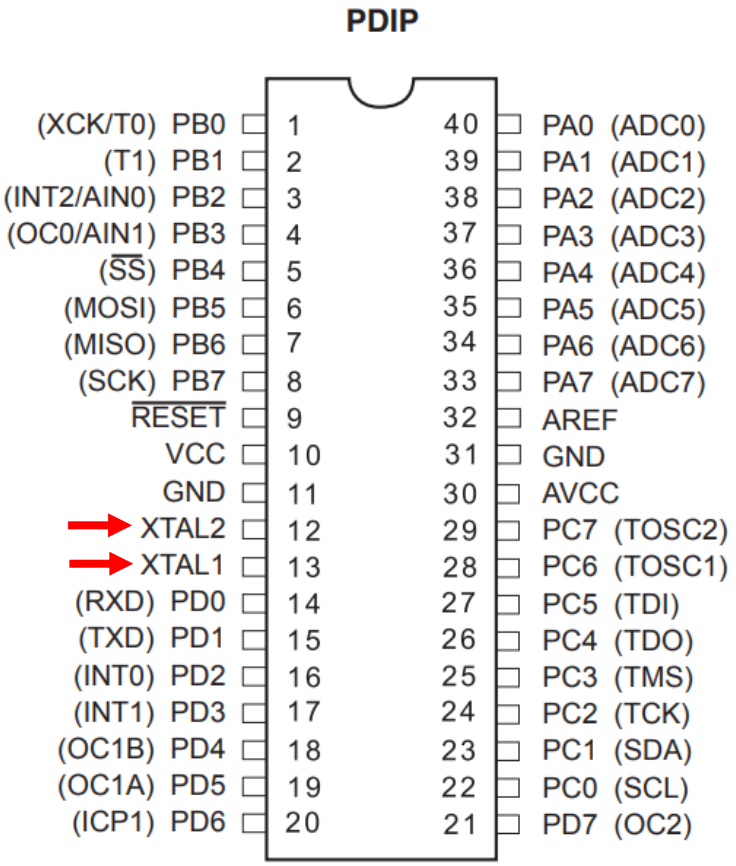
Machine Cycles

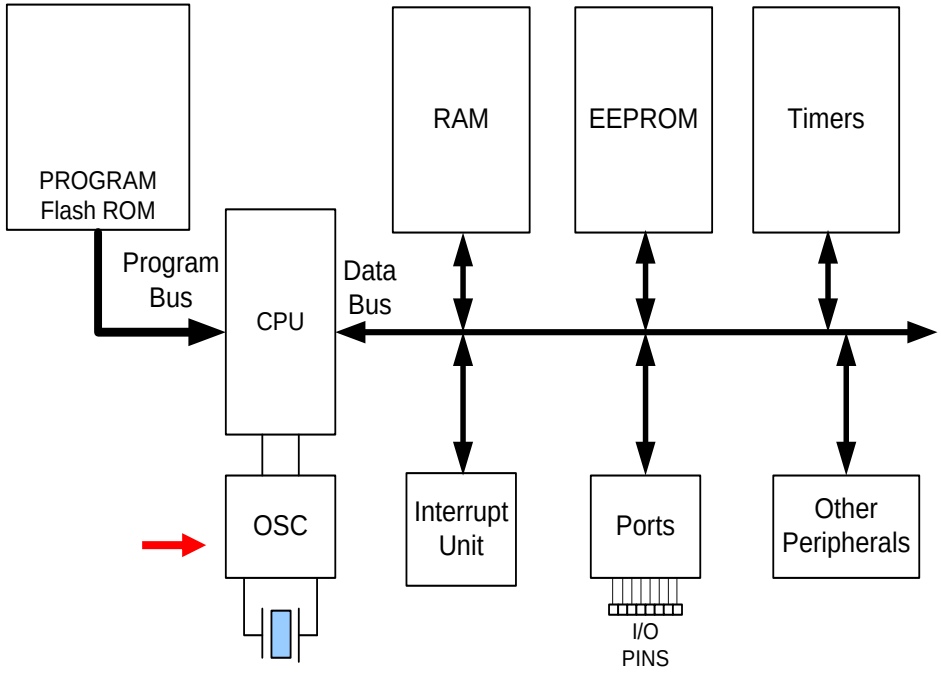
- Machine Cycles:
  - amount of time for the CPU to execute an instruction

Mnemonic	Operands	Description		Op		Flags	#Clocks AVR
ADD	Rd, Rr	Add without Carry	Rd	←	Rd + Rr	Z,C,N,V,S,H	1
ADC	Rd, Rr	Add with Carry	Rd	←	Rd + Rr + C	Z,C,N,V,S,H	1
ADIW	Rd, K	Add Immediate to Word	Rd	←	Rd + 1:Rd + K	Z,C,N,V,S	2
SUB	Rd, Rr	Subtract without Carry	Rd	←	Rd - Rr	Z,C,N,V,S,H	1
SUBI	Rd, K	Subtract Immediate	Rd	←	Rd - K	Z,C,N,V,S,H	1
SBC	Rd, Rr	Subtract with Carry	Rd	←	Rd - Rr - C	Z,C,N,V,S,H	1
SBCI	Rd, K	Subtract Immediate with Carry	Rd	←	Rd - K - C	Z,C,N,V,S,H	1
SBIW	Rd, K	Subtract Immediate from Word	Rd + 1:Rd	←	Rd + 1:Rd - K	Z,C,N,V,S	2
AND	Rd, Rr	Logical AND	Rd	←	Rd • Rr	Z,N,V,S	1

Time taken by the Machine Cycle?

- The length of the Machine Cycle (MC) depends on the frequency of the Oscillator Cycle (OC) connected to the AVR:
  - 1 MC = 1 OC





Example

The following shows the crystal frequency for four different AVR-based systems. Find the period of the instruction cycle in each case.
(a) 8 MHz (b) 16 MHz (c) 10 MHz (d) 1 MHz

Time delay

		machine cycle
LDI	R16, 19	1
LDI	R20, 95	1
LDI	R21, 5	1
ADD	R16, R20	1
ADD	R16, R21	1
		5

Delay = 5 x $T_{machine\ cycle}$ = 5 x 62.5 ns = 312.5 ns

Time delay

```
        LDI     R16, 100
AGAIN:  ADD     R17,R16
        DEC     R16
        BRNE    AGAIN
```

machine cycle	
1	
1	*100
1	*100
1/2	*100

Branch penalty

The penalty is an extra instruction cycle to fetch the instruction from the target location instead of executing the instruction right below the branch.

Delay Calculation

- Find the size of the delay of the code snippet below if the crystal frequency is 10MHz.

```
                                        Instruction Cycles
          .DEF COUNT = R20                      0
DELAY:    LDI      COUNT, 0xFF                  1
AGAIN:    NOP                                   1
          NOP                                   1
          DEC      COUNT                        1
          BRNE     AGAIN                        2/1
          RET                                   4
```

$$[1 + ((1 + 1 + 1 + 2) \times 255) + 4\ ] \times 0.1\ \mu s = 128.0\ \mu s$$

Is this correct?

BRNE takes two MCs if it jumps, else one MC when falling through the loop

[1+((1+1+1+2)x255)-1+4] x 0.1us = 127 us

Reading

- The AVR Microcontroller and Embedded Systems: Using Assembly and C by Mazidi et al., Prentice Hall
  - Chapter-3: 3.3

THANK YOU