



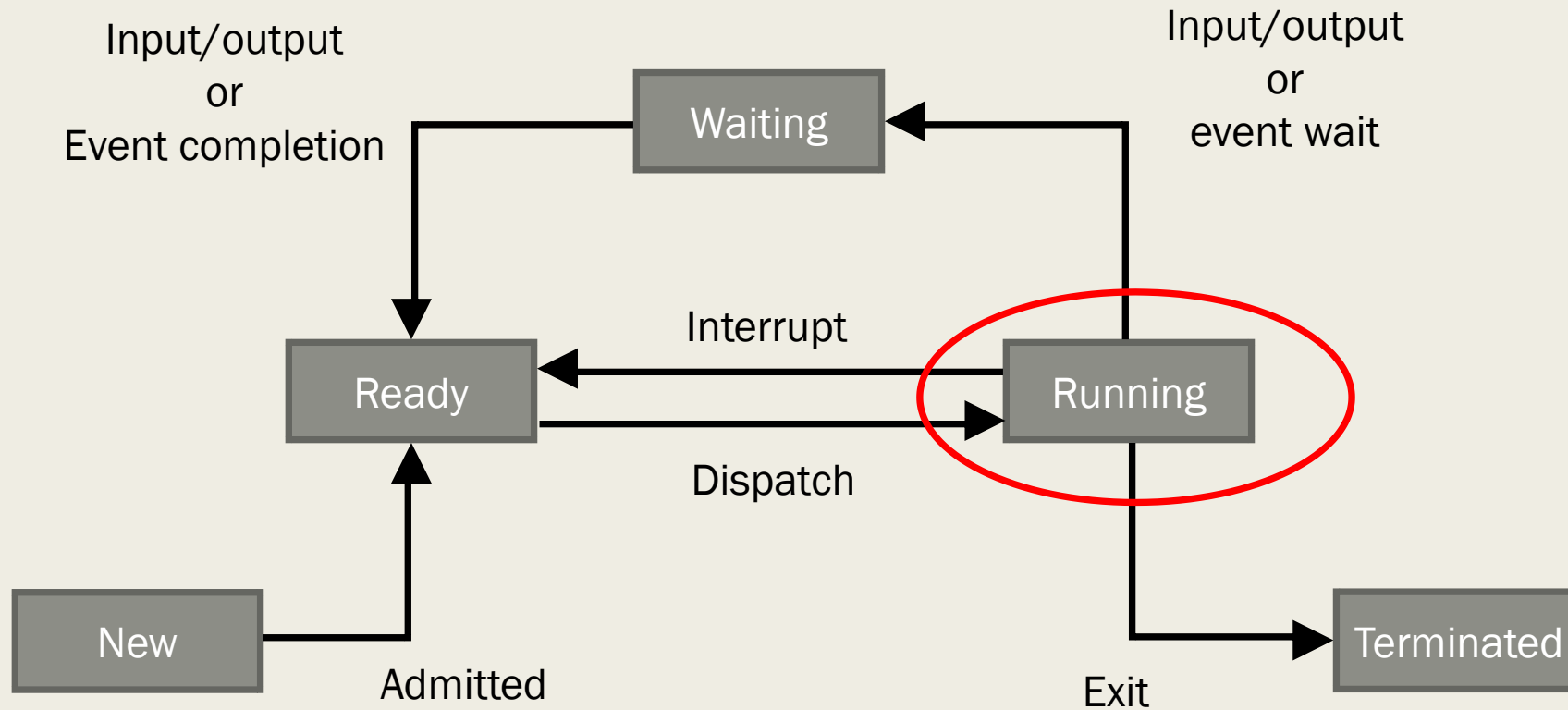
COMPUTER SYSTEMS FUNDAMENTALS (4COSC004W)



COMPUTER ARCHITECTURE

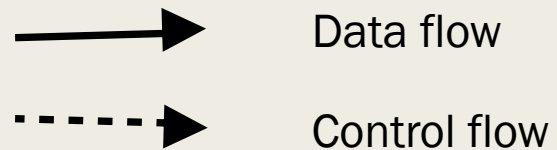
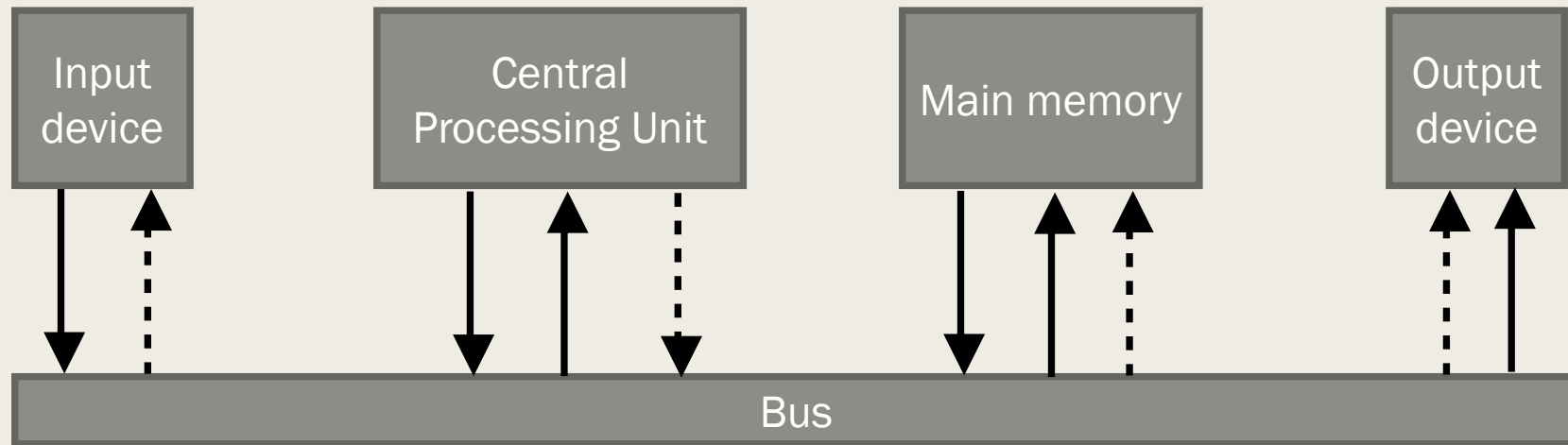
Hardware to the von Neumann Model

Process states



Major hardware components

Pep/9 computer



Major components

- Main memory
 - *We have discussed*
 - *Fully indexed scratch-pad*
 - *Instructions & data*
- Input devices
- Output Devices
- CPU

CPU (Central Processing Unit)

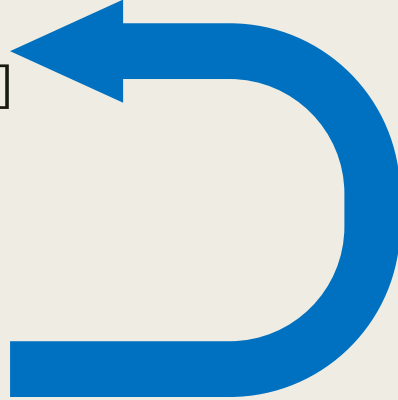
- A number of registers which hold specific pieces of information
 - *16 bit accumulator (A)*
 - Results of calculations
 - *16 bit index register (X)*
 - Elements in arrays
 - *16 bit Program counter (PC)*
 - Instructions
 - *16 bit Stack pointer (SP)*
 - Runtime stack
 - *24 bit Instruction Register (IR)*
 - Holds an instruction that has just been read
 - *4 status bits (N, Z, V, C)*
- Logical gates - electronics

CPU Instruction set

- CPUs have very limited instruction sets
- Pep/9 has an instruction set of 39 instructions
- Compilers are used to compile high-level programming languages into machine language.

Von-Neumann execution cycle

- Fetch instruction from Mem[PC]
- Decode the fetched instruction
- Increment PC
- Execute the fetched instruction



Address:	Machine Language (bin)						Machine Language (hex)
0000	1101	0001	0000	0000	0000	1101	D1000D ; Load byte accumulator 'H'
0003	1111	0001	1111	1100	0001	0110	F1FC16 ; Store byte accumulator output device
0006	1101	0001	0000	0000	0000	1110	D1000E ; Load byte accumulator 'i'
0009	1111	0001	1111	1100	0001	0110	F1FC16 ; Store byte accumulator output device
000C	0000	0000					00 ; stop
000D	0100	1000	0110	1001			4869 ; ASCII "Hi" characters

(Warford, 2017) Figure 4.33, 4.34

Address:	Machine Language (hex)
0000	D1000D ; Load byte accumulator 'H'
0003	F1FC16 ; Store byte accumulator output device
0006	D1000E ; Load byte accumulator 'i'
0009	F1FC16 ; Store byte accumulator output device
000C	00 ; stop
000D	4869 ; ASCII "Hi" characters

CPU	
A	
PC	
IR	

	Memory
0000	
0003	
0006	
0009	
000C	
000D	
	...
FC16	



(a) Initial state before loading

Address:	Machine Language (hex)
0000	D1000D ; Load byte accumulator 'H'
0003	F1FC16 ; Store byte accumulator output device
0006	D1000E ; Load byte accumulator 'i'
0009	F1FC16 ; Store byte accumulator output device
000C	00 ; stop
000D	4869 ; ASCII "Hi" characters

	Memory
0000	D1000D
0003	F1FC16
0006	D1000E
0009	F1FC16
000C	00
000D	4869
	...
FC16	

CPU	
A	
PC	
IR	



(b) Program loaded into memory

Address:	Machine Language (hex)
0000	D1000D ; Load byte accumulator 'H'
0003	F1FC16 ; Store byte accumulator output device
0006	D1000E ; Load byte accumulator 'i'
0009	F1FC16 ; Store byte accumulator output device
000C	00 ; stop
000D	4869 ; ASCII "Hi" characters



CPU	
A	
PC	0000
IR	

	Memory
0000	D1000D
0003	F1FC16
0006	D1000E
0009	F1FC16
000C	00
000D	4869
	...
FC16	



(c) PC ← 0000 (hex)

Address:	Machine Language (hex)
0000	D1000D ; Load byte accumulator 'H'
0003	F1FC16 ; Store byte accumulator output device
0006	D1000E ; Load byte accumulator 'i'
0009	F1FC16 ; Store byte accumulator output device
000C	00 ; stop
000D	4869 ; ASCII "Hi" characters



	Memory
0000	D1000D
0003	F1FC16
0006	D1000E
0009	F1FC16
000C	00
000D	4869
	...
FC16	

CPU	
A	
PC	0000
IR	D1000D

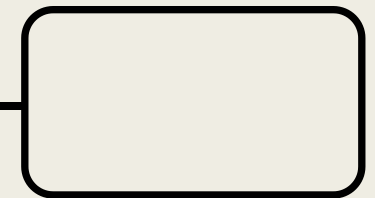


(d) Fetch instruction at Mem(PC)

Address:	Machine Language (hex)
0000	D1000D ; Load byte accumulator 'H'
0003	F1FC16 ; Store byte accumulator output device
0006	D1000E ; Load byte accumulator 'i'
0009	F1FC16 ; Store byte accumulator output device
000C	00 ; stop
000D	4869 ; ASCII "Hi" characters

	Memory
0000	D1000D
0003	F1FC16
0006	D1000E
0009	F1FC16
000C	00
000D	4869
	...
FC16	

CPU	
A	
PC	0003
IR	D1000D



(e) Increment PC

Address:	Machine Language (hex)
0000	D1000D ; Load byte accumulator 'H'
0003	F1FC16 ; Store byte accumulator output device
0006	D1000E ; Load byte accumulator 'i'
0009	F1FC16 ; Store byte accumulator output device
000C	00 ; stop
000D	4869 ; ASCII "Hi" characters

	Memory
0000	D1000D
0003	F1FC16
0006	D1000E
0009	F1FC16
000C	00
000D	4869
	...
FC16	

CPU	
A	0048
PC	0003
IR	D1000D



(f) Execute. Load byte for H to accumulator

Address:	Machine Language (hex)
0000	D1000D ; Load byte accumulator 'H'
0003	F1FC16 ; Store byte accumulator output device
0006	D1000E ; Load byte accumulator 'i'
0009	F1FC16 ; Store byte accumulator output device
000C	00 ; stop
000D	4869 ; ASCII "Hi" characters



	Memory
0000	D1000D
0003	F1FC16
0006	D1000E
0009	F1FC16
000C	00
000D	4869
	...
FC16	

CPU	
A	0048
PC	0003
IR	F1FC16



(g) Fetch instruction at Mem(PC)

Address:	Machine Language (hex)
0000	D1000D ; Load byte accumulator 'H'
0003	F1FC16 ; Store byte accumulator output device
0006	D1000E ; Load byte accumulator 'i'
0009	F1FC16 ; Store byte accumulator output device
000C	00 ; stop
000D	4869 ; ASCII "Hi" characters

	Memory
0000	D1000D
0003	F1FC16
0006	D1000E
0009	F1FC16
000C	00
000D	4869
	...
FC16	

CPU	
A	0048
PC	0006
IR	F1FC16

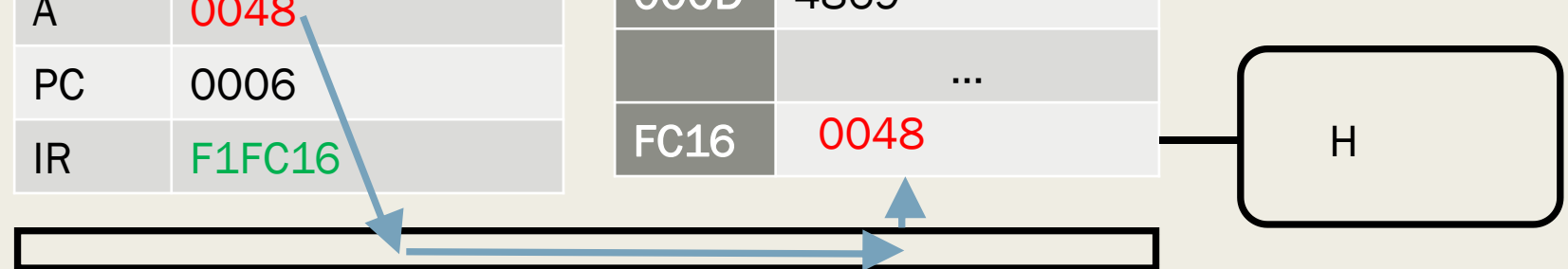


(h) Increment PC

Address:	Machine Language (hex)
0000	D1000D ; Load byte accumulator 'H'
0003	F1FC16 ; Store byte accumulator output device
0006	D1000E ; Load byte accumulator 'i'
0009	F1FC16 ; Store byte accumulator output device
000C	00 ; stop
000D	4869 ; ASCII "Hi" characters

	Memory
0000	D1000D
0003	F1FC16
0006	D1000E
0009	F1FC16
000C	00
000D	4869
	...
FC16	0048

CPU	
A	0048
PC	0006
IR	F1FC16

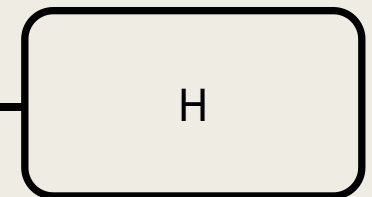


(i) Execute. Store byte from accumulator to output device

Address:	Machine Language (hex)
0000	D1000D ; Load byte accumulator 'H'
0003	F1FC16 ; Store byte accumulator output device
0006	D1000E ; Load byte accumulator 'i' ← PC
0009	F1FC16 ; Store byte accumulator output device
000C	00 ; stop
000D	4869 ; ASCII "Hi" characters

CPU	
A	0048
PC	0006
IR	D1000E

	Memory
0000	D1000D
0003	F1FC16
0006	D1000E
0009	F1FC16
000C	00
000D	4869
	...
FC16	

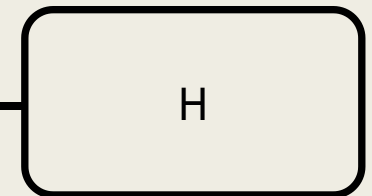


(j) Fetch instruction at Mem(PC)

Address:	Machine Language (hex)
0000	D1000D ; Load byte accumulator 'H'
0003	F1FC16 ; Store byte accumulator output device
0006	D1000E ; Load byte accumulator 'i'
0009	F1FC16 ; Store byte accumulator output device
000C	00 ; stop
000D	4869 ; ASCII "Hi" characters

	Memory
0000	D1000D
0003	F1FC16
0006	D1000E
0009	F1FC16
000C	00
000D	4869
	...
FC16	

CPU	
A	0048
PC	0009
IR	D1000E

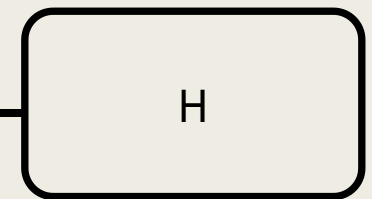


(k) Increment PC

Address:	Machine Language (hex)
0000	D1000D ; Load byte accumulator 'H'
0003	F1FC16 ; Store byte accumulator output device
0006	D1000E ; Load byte accumulator 'i'
0009	F1FC16 ; Store byte accumulator output device
000C	00 ; stop
000D	4869 ; ASCII "Hi" characters

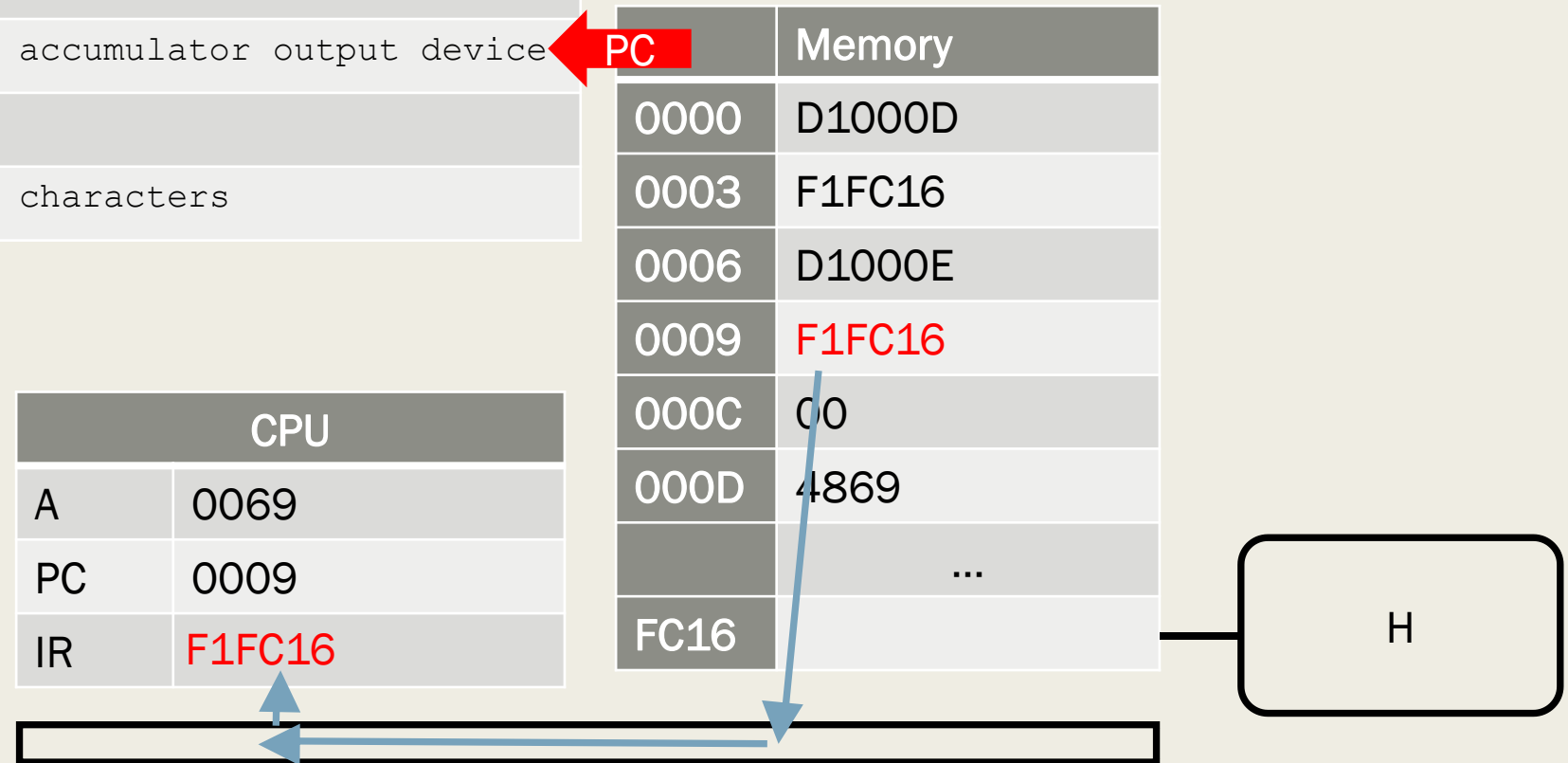
	Memory
0000	D1000D
0003	F1FC16
0006	D1000E
0009	F1FC16
000C	00
000D	4869
	...
FC16	

CPU	
A	0069
PC	0009
IR	D1000E



(I) Execute. Load byte for i to accumulator

Address:	Machine Language (hex)
0000	D1000D ; Load byte accumulator 'H'
0003	F1FC16 ; Store byte accumulator output device
0006	D1000E ; Load byte accumulator 'i'
0009	F1FC16 ; Store byte accumulator output device
000C	00 ; stop
000D	4869 ; ASCII "Hi" characters

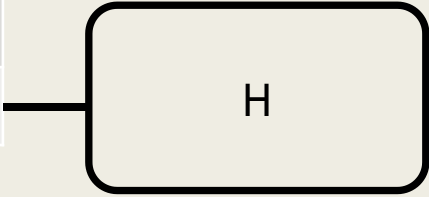


(m) Fetch instruction at Mem(PC)

Address:	Machine Language (hex)
0000	D1000D ; Load byte accumulator 'H'
0003	F1FC16 ; Store byte accumulator output device
0006	D1000E ; Load byte accumulator 'i'
0009	F1FC16 ; Store byte accumulator output device
000C	00 ; stop
000D	4869 ; ASCII "Hi" characters

	Memory
0000	D1000D
0003	F1FC16
0006	D1000E
0009	F1FC16
000C	00
000D	4869
	...
FC16	

CPU	
A	0069
PC	000C
IR	F1FC16



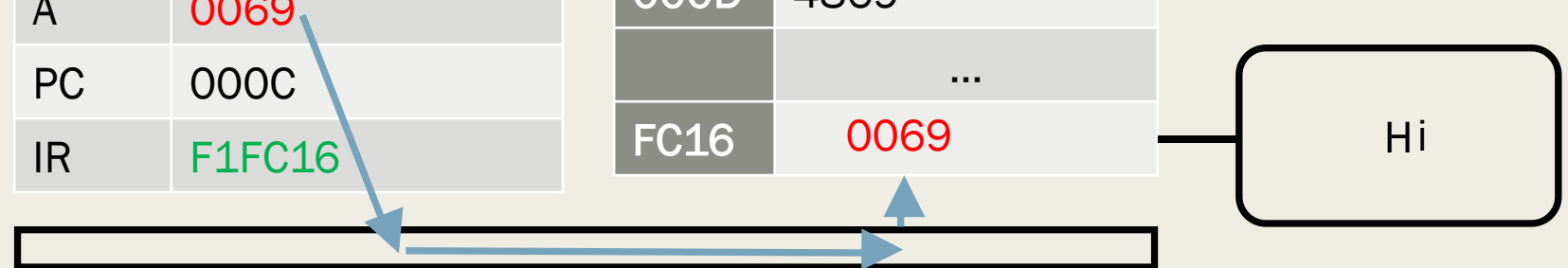
(n) Increment PC

Address:	Machine Language (hex)
0000	D1000D ; Load byte accumulator 'H'
0003	F1FC16 ; Store byte accumulator output device
0006	D1000E ; Load byte accumulator 'i'
0009	F1FC16 ; Store byte accumulator output device
000C	00 ; stop
000D	4869 ; ASCII "Hi" characters



	Memory
0000	D1000D
0003	F1FC16
0006	D1000E
0009	F1FC16
000C	00
000D	4869
	...
FC16	0069

CPU	
A	0069
PC	000C
IR	F1FC16

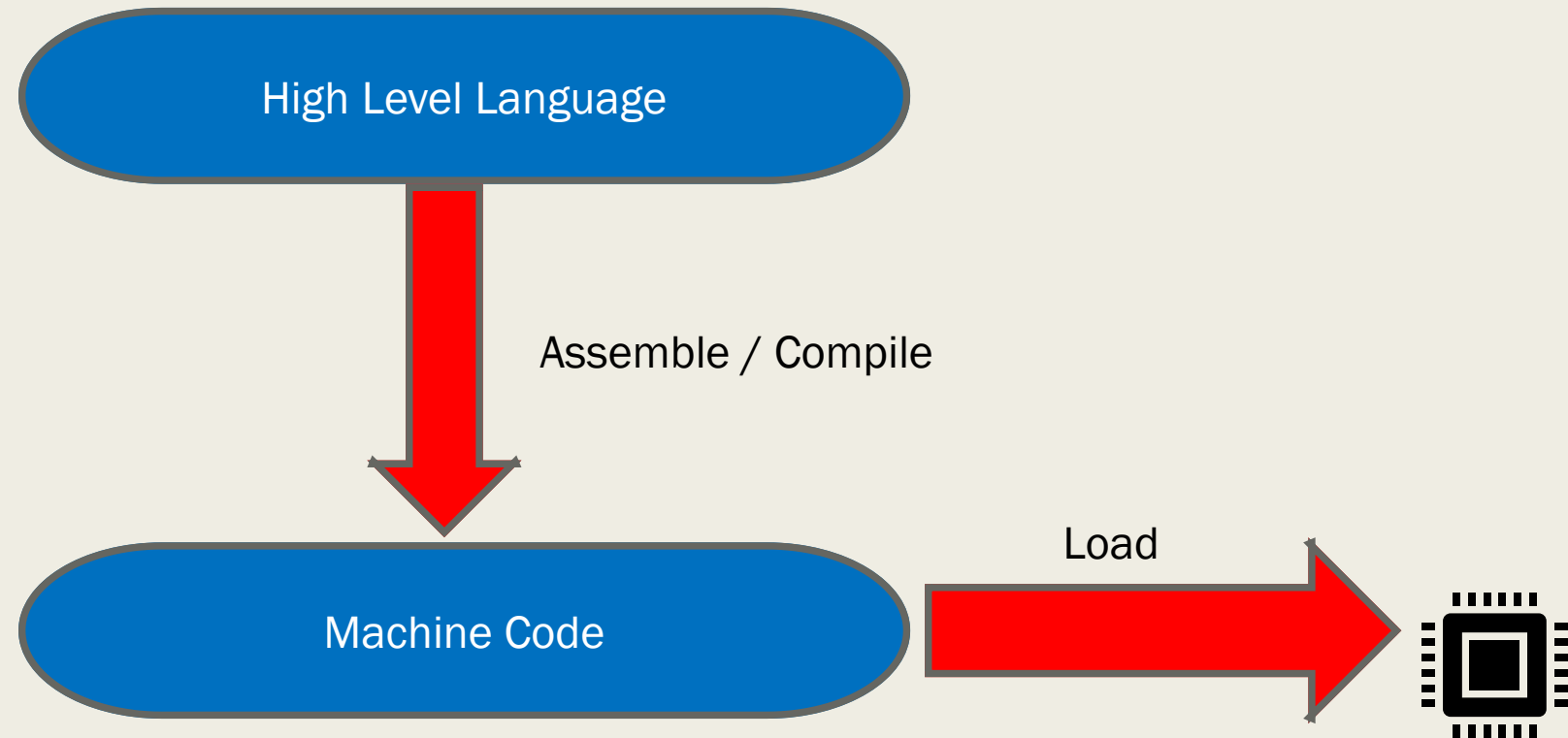


(o) Execute. Store byte from accumulator to output device

ASSEMBLY LANGUAGE

One-step up from Machine Language

High and Low Level languages:



Instruction set

- RISC (**R**educed **I**nstruction **S**et **C**omputing)
Asmb5 – 39 commands
- CISC (**C**omplex **I**nstruction **S**et **C**omputing)

Operand addressing methods

Direct

- $\text{Oprnd} = \text{Mem}[\text{OprndSpec}]$
- Asmb5 letter: d
- The operand specifier is the *address* in memory of the operand

Immediate

- $\text{Oprnd} = \text{OprndSpec}$
- Asmb5 letter: i
- The operand specifier is the operand

PEP9 DEMONSTRATION

Further reading:

- Computer Systems, S. Warford
 - *4. Computer Architecture (pp. 184 – 230)*

By this stage you:

- Have gain an appreciation of ;
 - *Major hardware components*
 - *CPU*
 - Cycles
 - *Von Neumann Model*
 - *Machine code*

Thank you

© The University of Westminster (2021)

These notes were modified from the lecture slides generated by Noam Weingarten.
The right of Noam Weingarten to be identified as author of this work has been asserted by them in accordance with the Copyright, Designs and Patents Act 1988