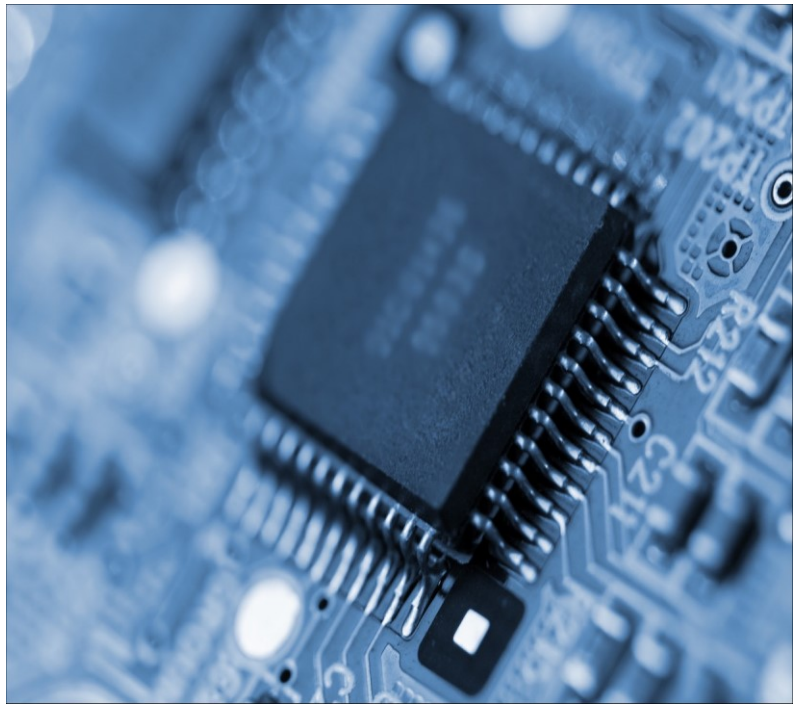
UNIVERSITY OF LEEDS

Computer Processors

Introduction & module logistics



Module Staff

UNIVERSITY OF LEEDS

Module leader:

- Samson Fabiyi (s.d.fabiyi@leeds.ac.uk)

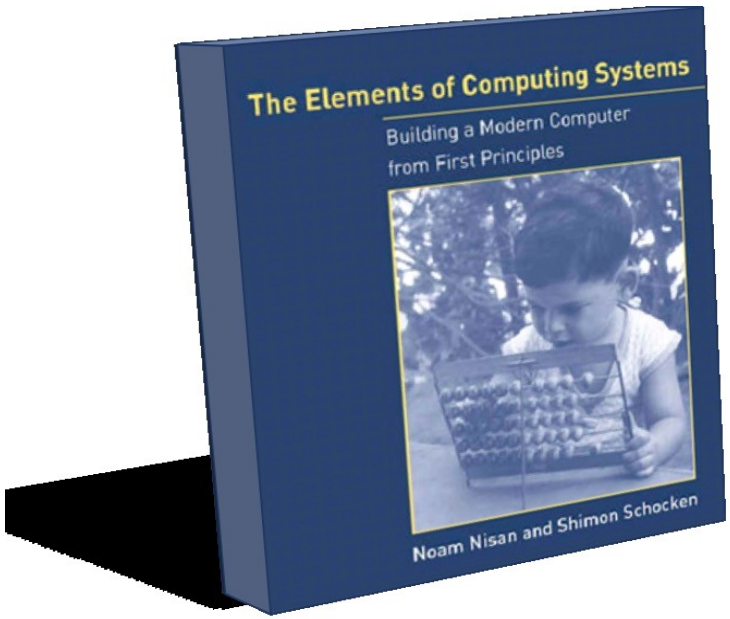
Module Co-Leaders:

- Liu Heng (hengliu@swjtu.edu.cn)

Module Delivery

UNIVERSITY OF LEEDS

- Lectures and Tutorials
  - Lecture – 135 mins/day
  - Tutorial - 135 mins/day
- Lab classes
- Reading
- nand2tetris.org
- XJCO1212 module channel in Microsoft Teams



Module Assessment

UNIVERSITY OF LEEDS

- **Coursework 1 – 40% Gradescope (Deadline: 21 March 2024)**
  - You will be expected to create HDL files given a description:
    - As a boolean formula, as a truth table and as a HDL file

- **Coursework 2 – 60% Gradescope (Deadline: 2 May 2024)**
  - You will be expected to implement a software systems for a defined encryption system.

In-class Formative quizzes and tasks

UNIVERSITY OF LEEDS

- **Top Hat:** We will be using this to engage with questions during lectures**.**
- Are marks/grades attached to the in-class formative quizzes and tasks? **No**
- Then, why are they necessary? **We will discuss this in the next two slides**

In-class Formative quizzes and tasks: why are they necessary?

UNIVERSITY OF LEEDS

"The most striking finding was that students who participated in the quiz scored, on average, 13% higher on the summative examinations than students who did not participate. Complete data for each individual quiz are shown in Table 1." - Jonathan D. Kibble

Table 1. *Student participation and performance on formative assessments*

Assessment	Students Who Took Quizzes		Students Who Did Not Take Quizzes		*P* Value	*R* Value (Correlation of Quiz Score Versus Exam Score)
	Mean test score, %	*n*	Mean test score, %	*n*		
Quiz 1	66 ± 14	102	N/A	7	0.01*	0.46†
Exam 1	79 ± 16	102	64 ± 15	7		
Quiz 2	58 ± 15	104	N/A	4	NS	0.27†
Exam 2	74 ± 12	104	64 ± 17	4		
Quiz 3	67 ± 18	94	N/A	12	NS	0.44†
Exam 3	70 ± 15	94	66 ± 17	12		
Quiz 4	52 ± 21	94	N/A	10	0.03*	0.56†
Exam 4	79 ± 13	94	64 ± 18	10		
Quiz 5	56 ± 22	95	N/A	8	0.04*	0.26†
Exam 5	79 ± 13	95	67 ± 14	8		
Quiz 6	56 ± 21	89	N/A	15	0.04*	0.47†
Exam 6	76 ± 14	89	62 ± 22	15		

Test score data are expressed as means ± SD; *n*, number of students. N/A, not applicable; NS, no significant difference; *R*, least-squares correlation coefficient relating individual student quiz scores to their examination scores. *Significant difference between the summative exam score for students that took the corresponding formative quiz compared with students that did not take the quiz (by Student's *t*-test; *P* value shown); †significant correlation between the quiz score and the corresponding exam score ($P < 0.01$).

Reference
J. D. Kibble, "Voluntary participation in online formative quizzes is a sensitive predictor of student success," Advances in Physiology Education, vol. 35, no. 1, pp. 95–96, Mar. 2011, doi: https://doi.org/10.1152/advan.00053.2010.

In-class Formative quizzes and tasks: why are they necessary?

UNIVERSITY OF LEEDS

"It was concluded that the students' academic achievement, SSAS, and SRSS scores were similar prior to the experimental procedures. There was a significant difference between the academic achievement post-test scores of the control and experimental group students…..According to this, it can be inferred that the academic achievements of the students in the experimental group statistically significantly increased with the experimental procedures as compared to the control group students.." - Ceyhun Ozan and Remzi Y. Kıncal

Table 1

Descriptive Statistics of the Pre-Test and Post-Test Scores

Dependent Variable	Group	n	□	Adjusted Mean
Academic achievement (Pre-test)	Control	21	23.43	23.43
	Experimental	24	22.42	22.42
SSAS (Pre-test)	Control	21	4.16	4.16
	Experimental	24	4.24	4.24
SRSS (Pre-test)	Control	21	4.27	4.27
	Experimental	24	4.15	4.15
Academic achievement (Post-test)	Control	21	28.19	28.25
	Experimental	24	34.13	34.07
SSAS (Post-test)	Control	21	4.32	4.32
	Experimental	24	4.75	4.75
SRSS (Post-test)	Control	21	4.35	4.35
	Experimental	24	4.62	4.63

Reference
C. Ozan and R. Y. Kıncal, "The Effects of Formative Assessment on Academic achievement, Attitudes toward the lesson, and self-regulation Skills," Educational Sciences: Theory & Practice, vol. 18, no. 1, pp. 85–118, Feb. 2018, doi: https://doi.org/10.12738/estp.2018.1.0216.

Module Objectives

UNIVERSITY OF LEEDS

- To understand how hardware and software systems are built and to understand the interaction between them.

- To be able to design and implement hardware systems

- To be able to decompose a problem into smaller problems

- To understand the role of Computer Aided Design software in the design and implementation of computer systems

About this Module

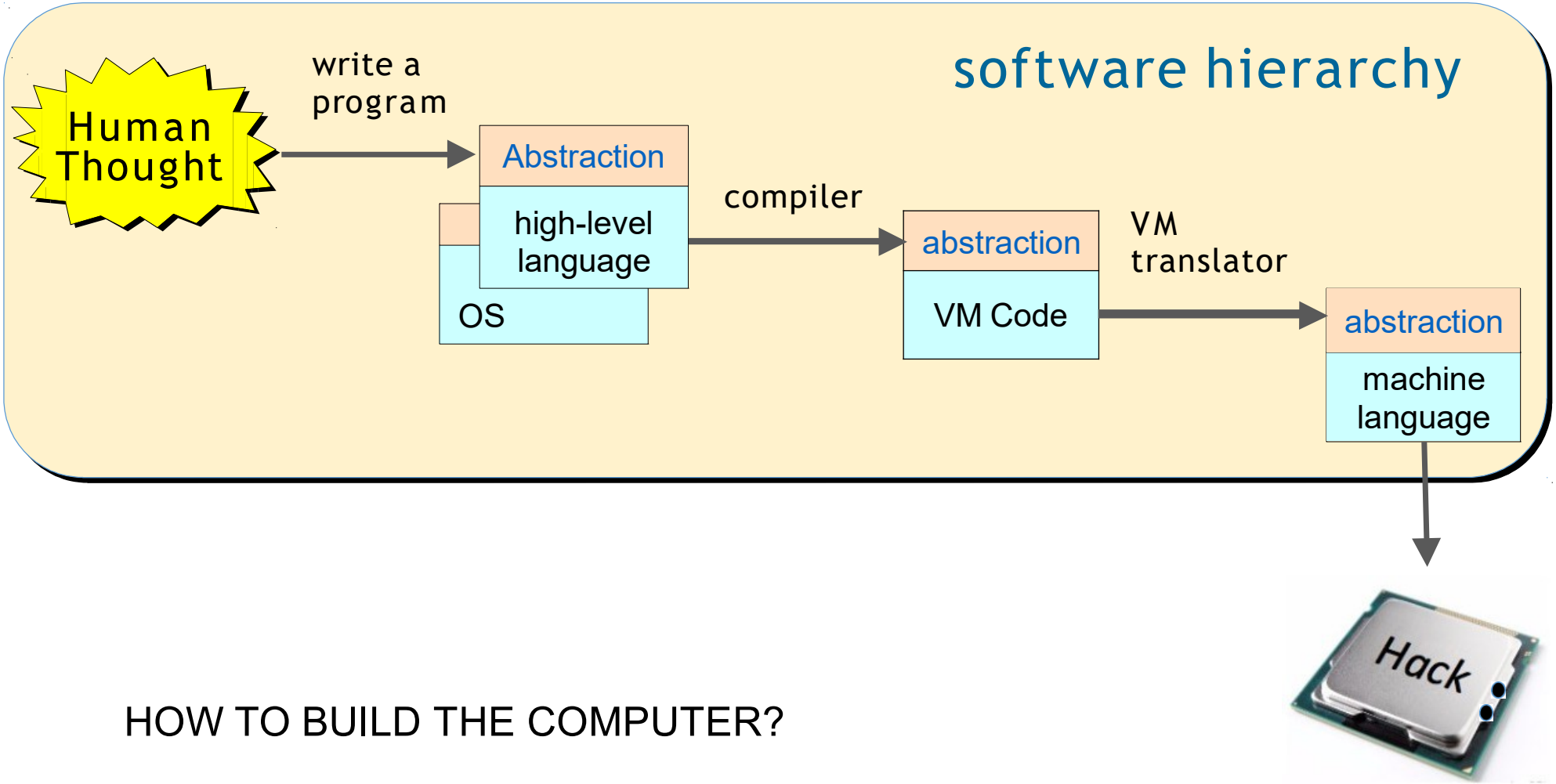
UNIVERSITY OF LEEDS

```
// First example in Programming 101
class Main {
    function void main() {
        do Output.printString("Hello World!");
        do Output.println(); // New line.
        return;
    }
}
```

HOW DOES THIS WORK?

About this Module

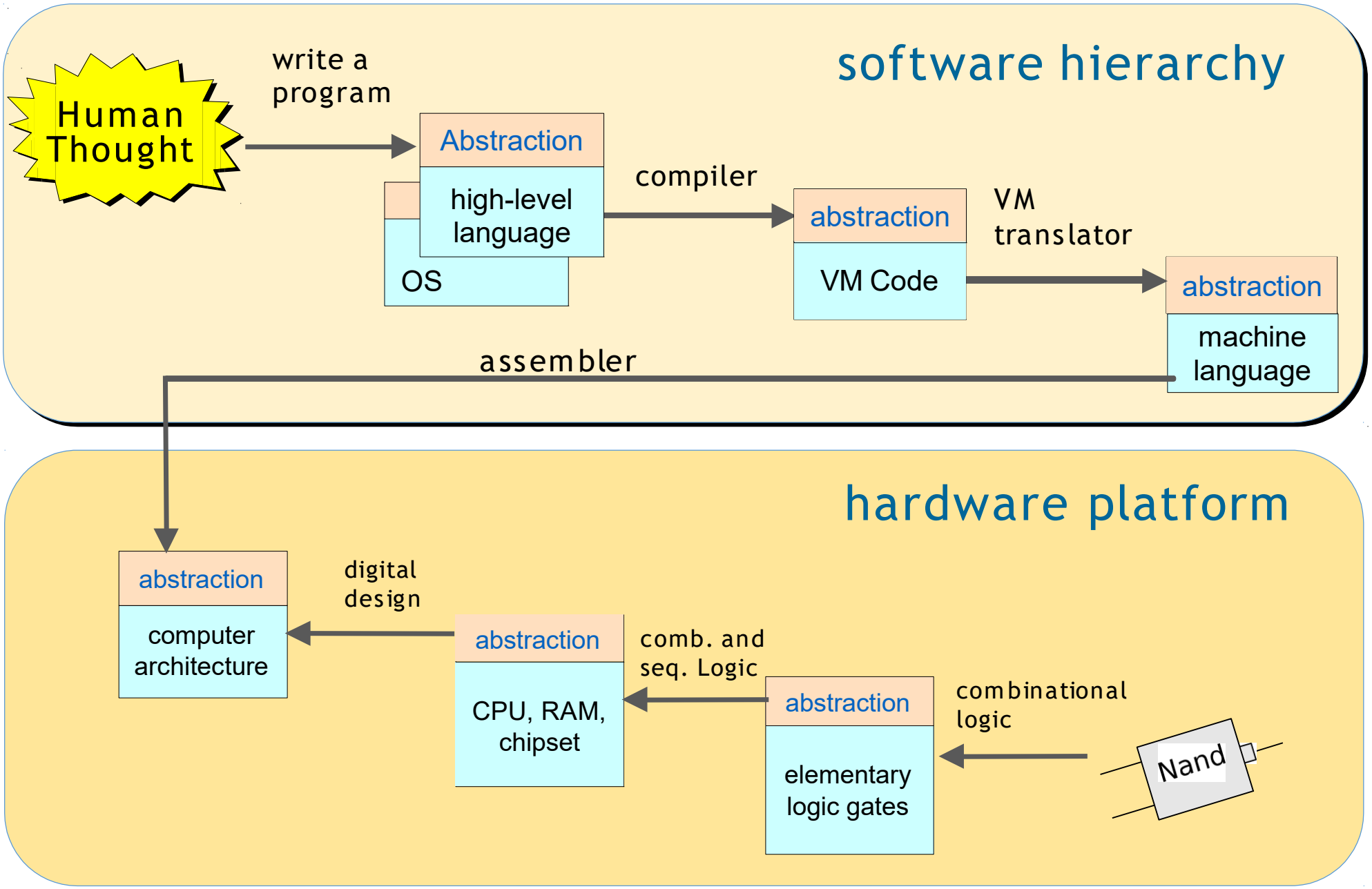
UNIVERSITY OF LEEDS



HOW TO BUILD THE COMPUTER?

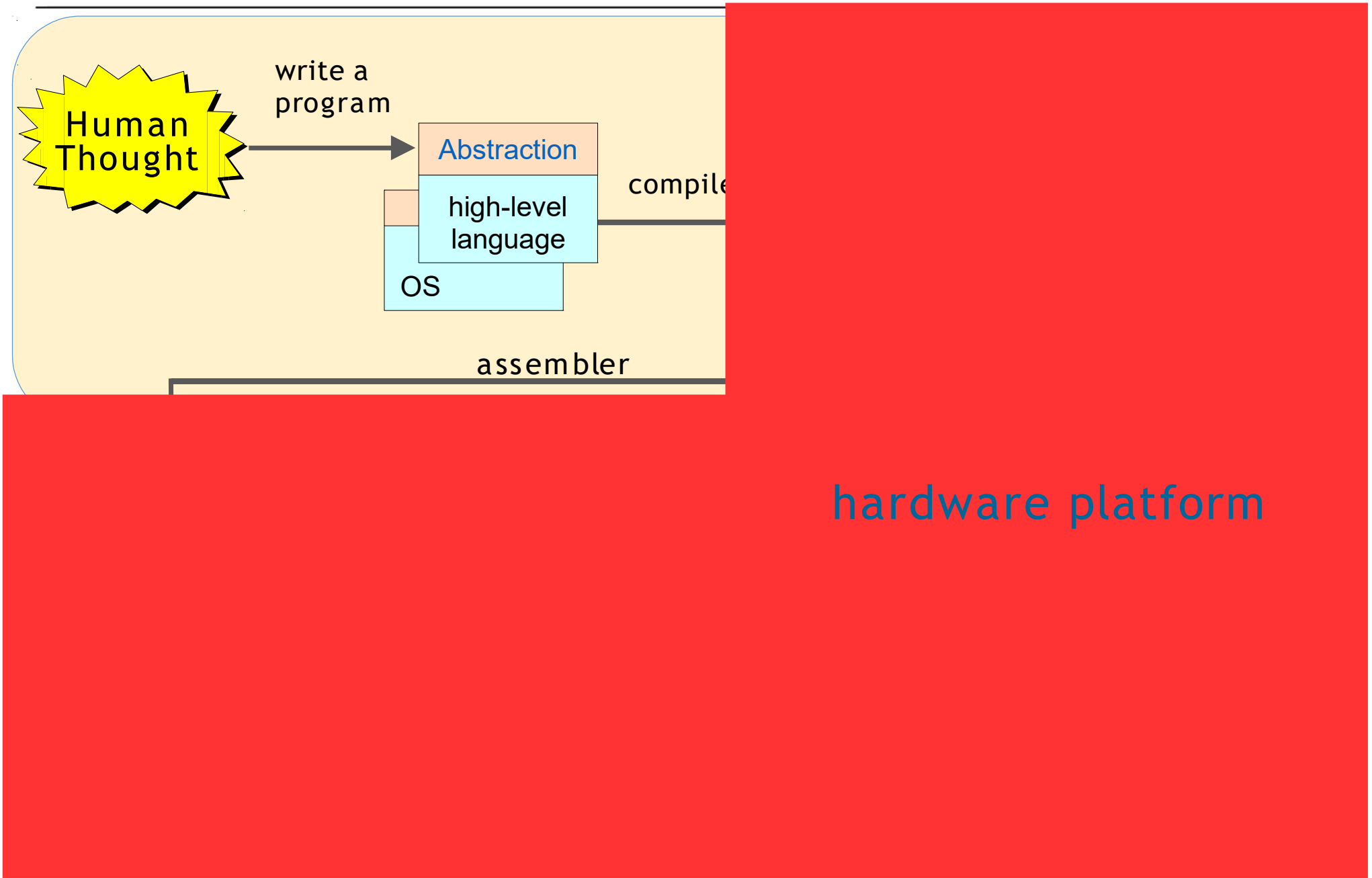
About this Module

UNIVERSITY OF LEEDS



About this Module

UNIVERSITY OF LEEDS

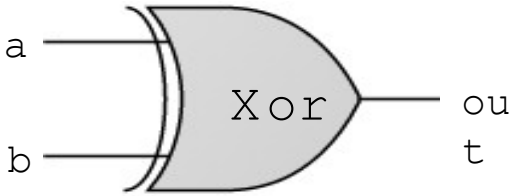


About this Module

UNIVERSITY OF LEEDS

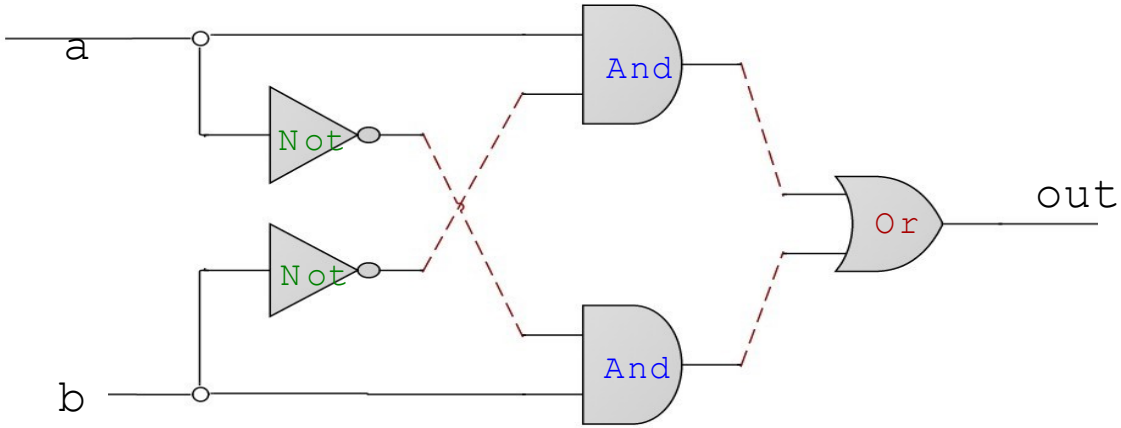
PART 1:

- Understanding Boolean Logic and Boolean Gates
- Hardware Description Language



outputs 1 if one, and only one, of its inputs, is 1.

a	b	out
0	0	**0**
0	1	**1**
1	0	**1**
1	1	**0**

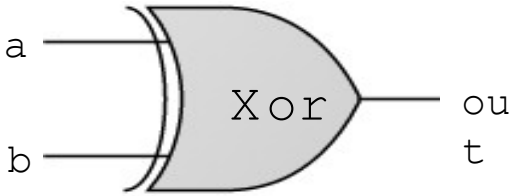


About this Module

UNIVERSITY OF LEEDS

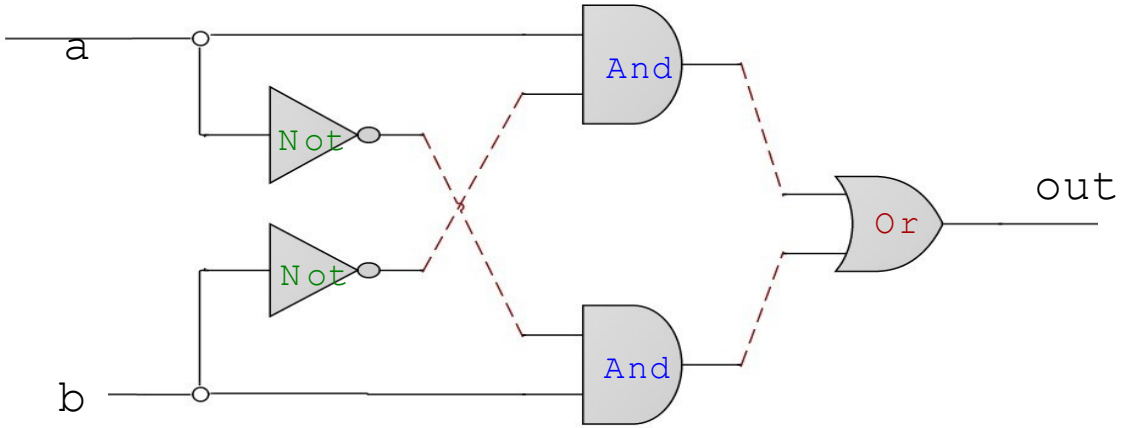
PART 1:

- Understanding Boolean Logic and Boolean Gates
- Hardware Description Language



outputs 1 if one, and only one, of its inputs, is 1.

a	b	out
0	0	**0**
0	1	**1**
1	0	**1**
1	1	**0**



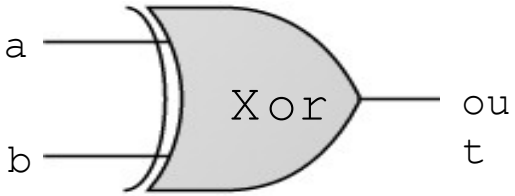

```
/** Xor gate: out = (a And Not(b)) Or (Not(a) And
    b)) */

CHIP Xor {
    IN a, b;
    OUT out;

    PARTS:
    // Implementation missing
}
```

About this Module

UNIVERSITY OF LEEDS

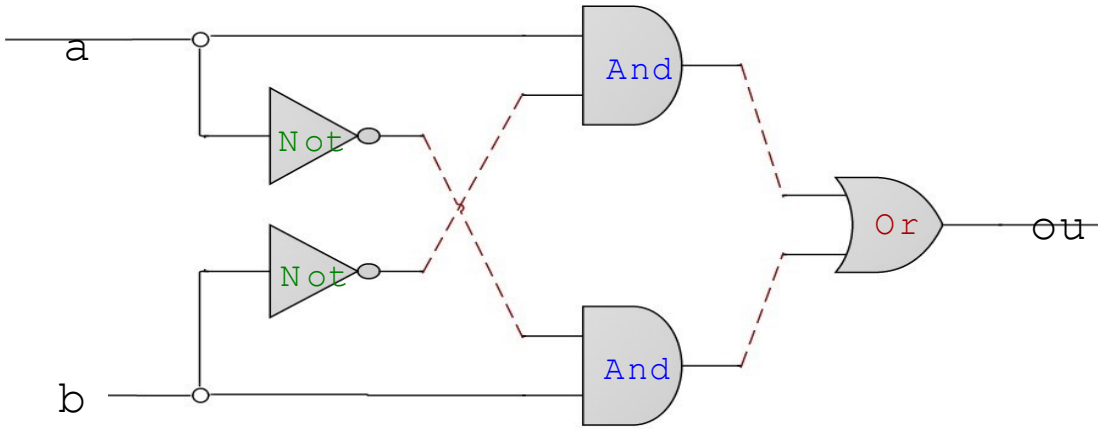
PART 1:

- Understanding Boolean Logic and Boolean Gates
- Hardware Description Language



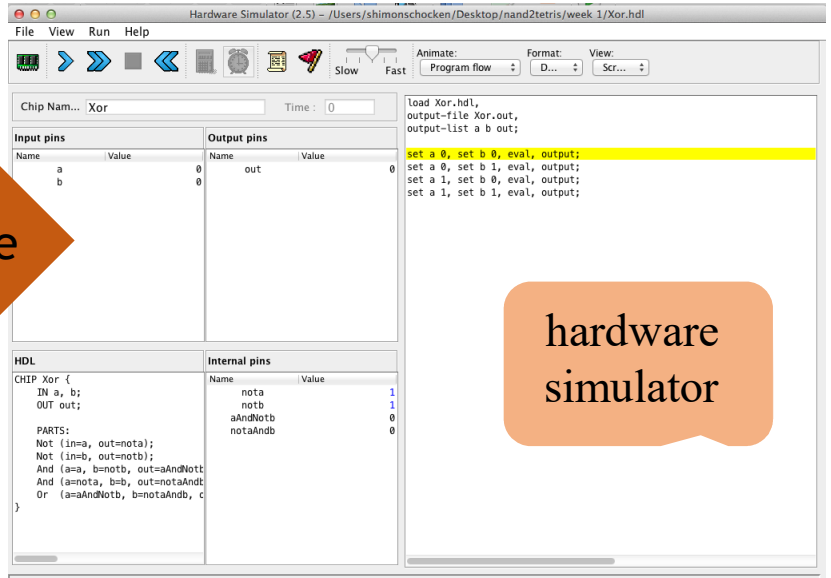
outputs 1 if one, and only one, of its inputs, is 1.

a	b	out
0	0	**0**
0	1	**1**
1	0	**1**
1	1	**0**




```
/** Xor gate: out = (a And Not(b)) Or (Not(a) And
    b)) */

CHIP Xor {
    IN a, b;
    OUT out;

    PARTS:
    // Implementation missing
}
```

simulate



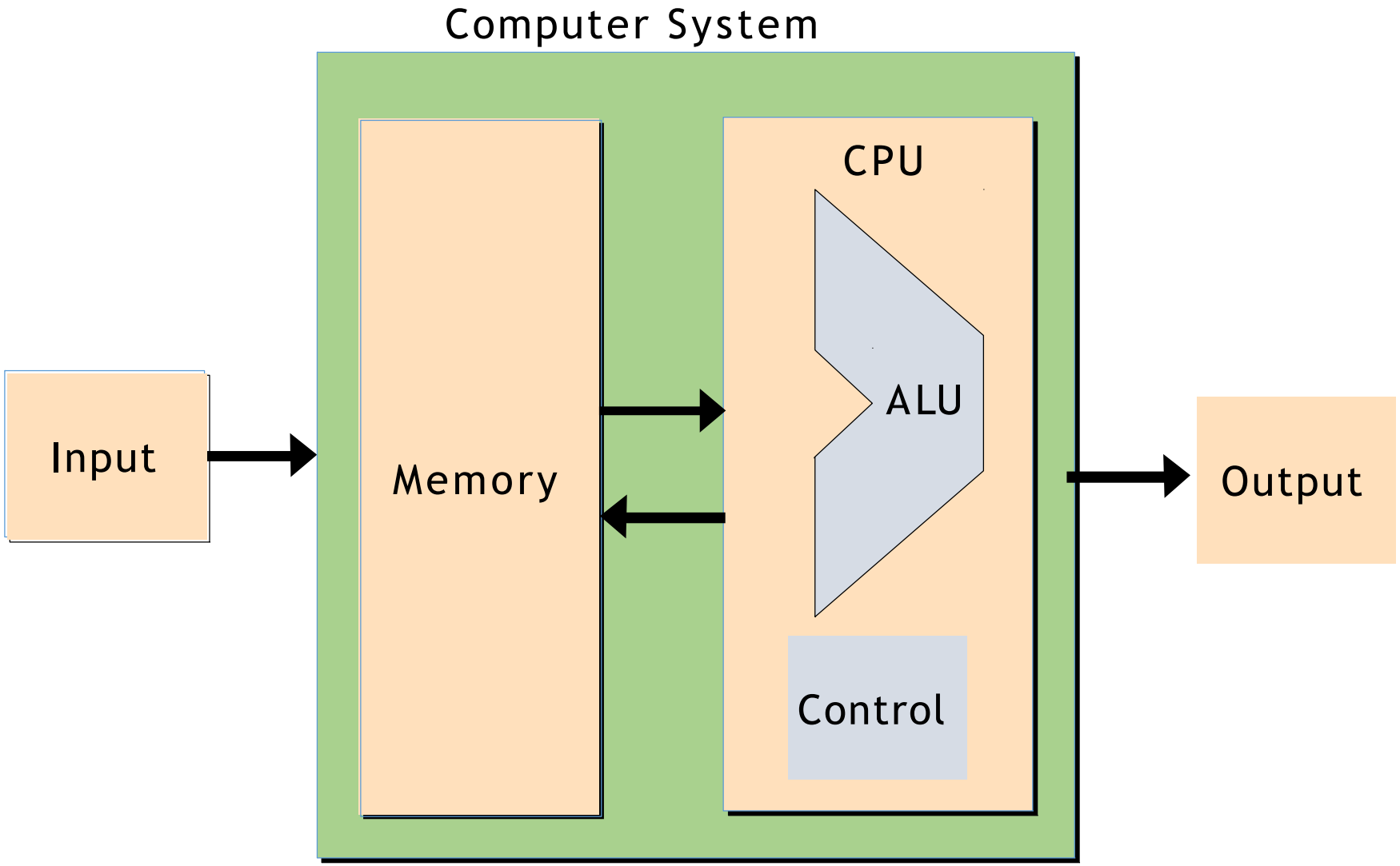
hardware simulator

About this Module

UNIVERSITY OF LEEDS

PART 2:

- Building blocks of the computer (Arithmetic Logic Unit, Memory, CPU)

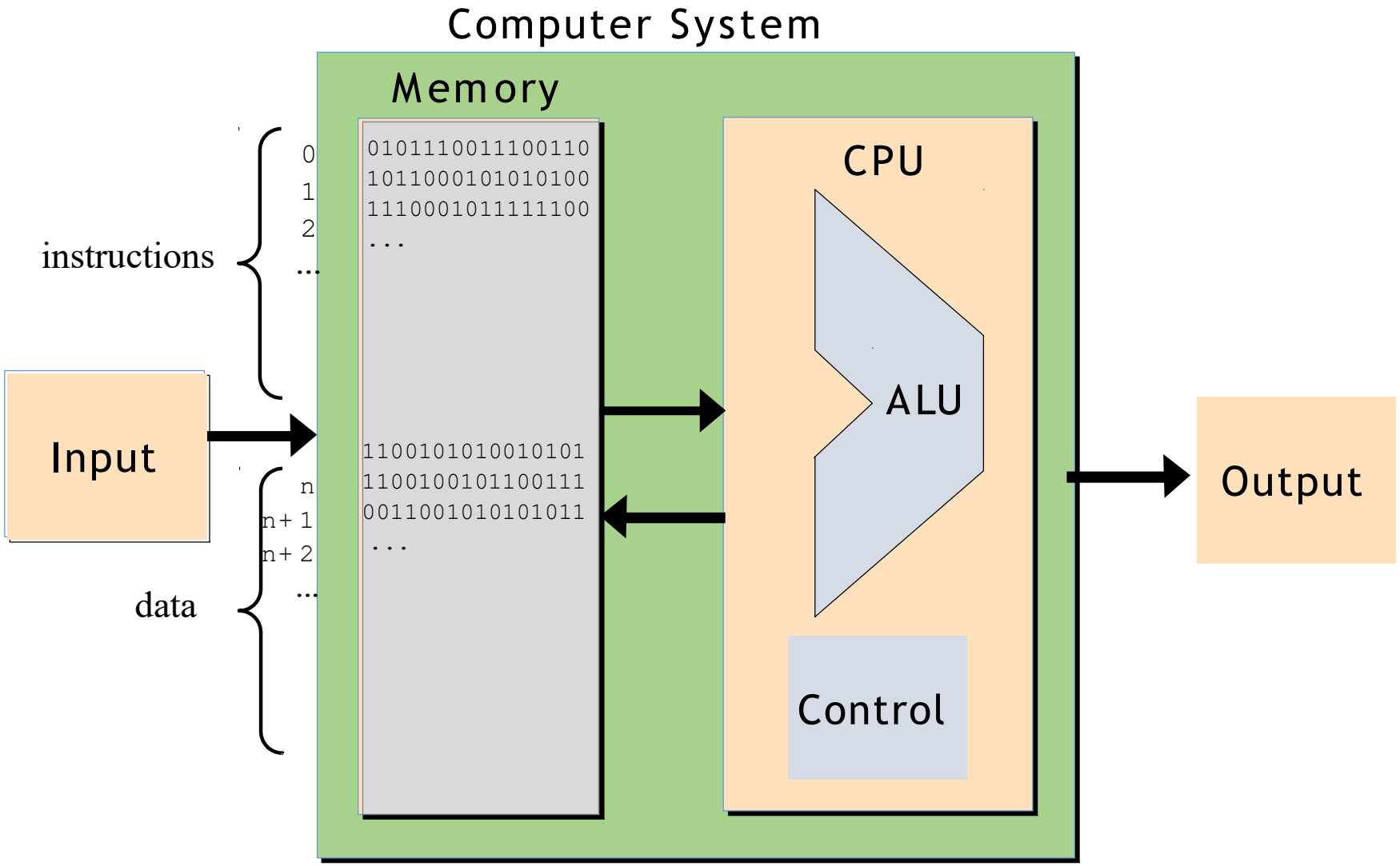


About this Module

UNIVERSITY OF LEEDS

PART 3:

- Machine Language (binary language understood by our computer)

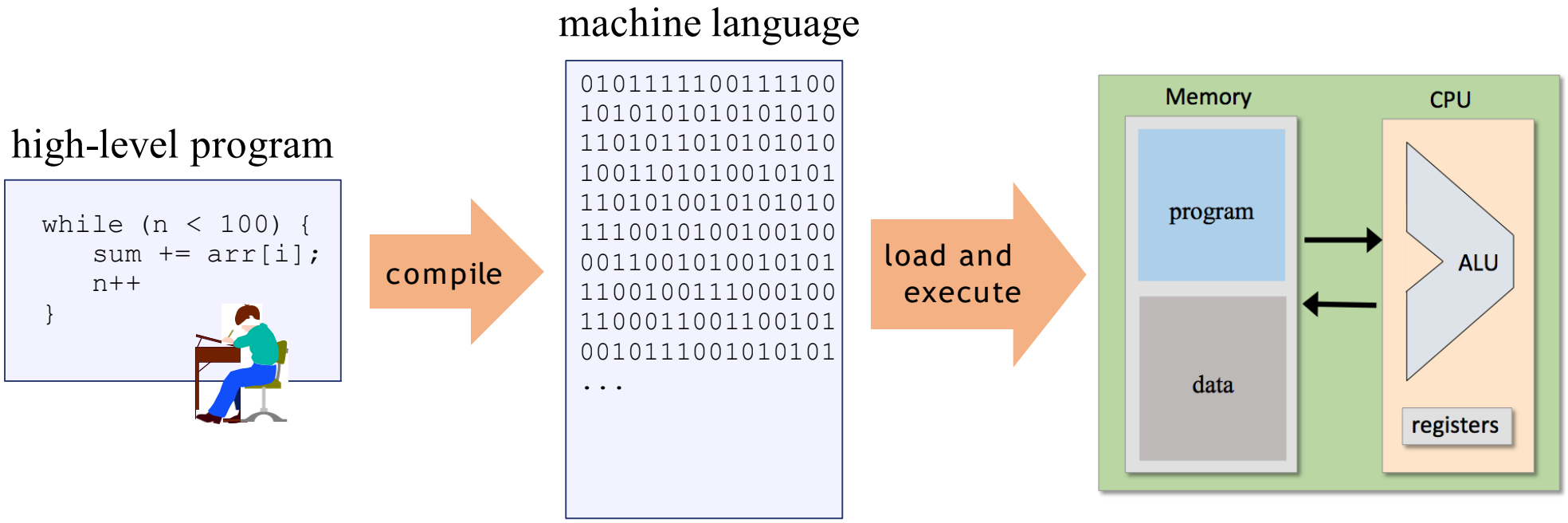


About this Module

UNIVERSITY OF LEEDS

PART 3:

- Machine Language (binary language understood by our computer)

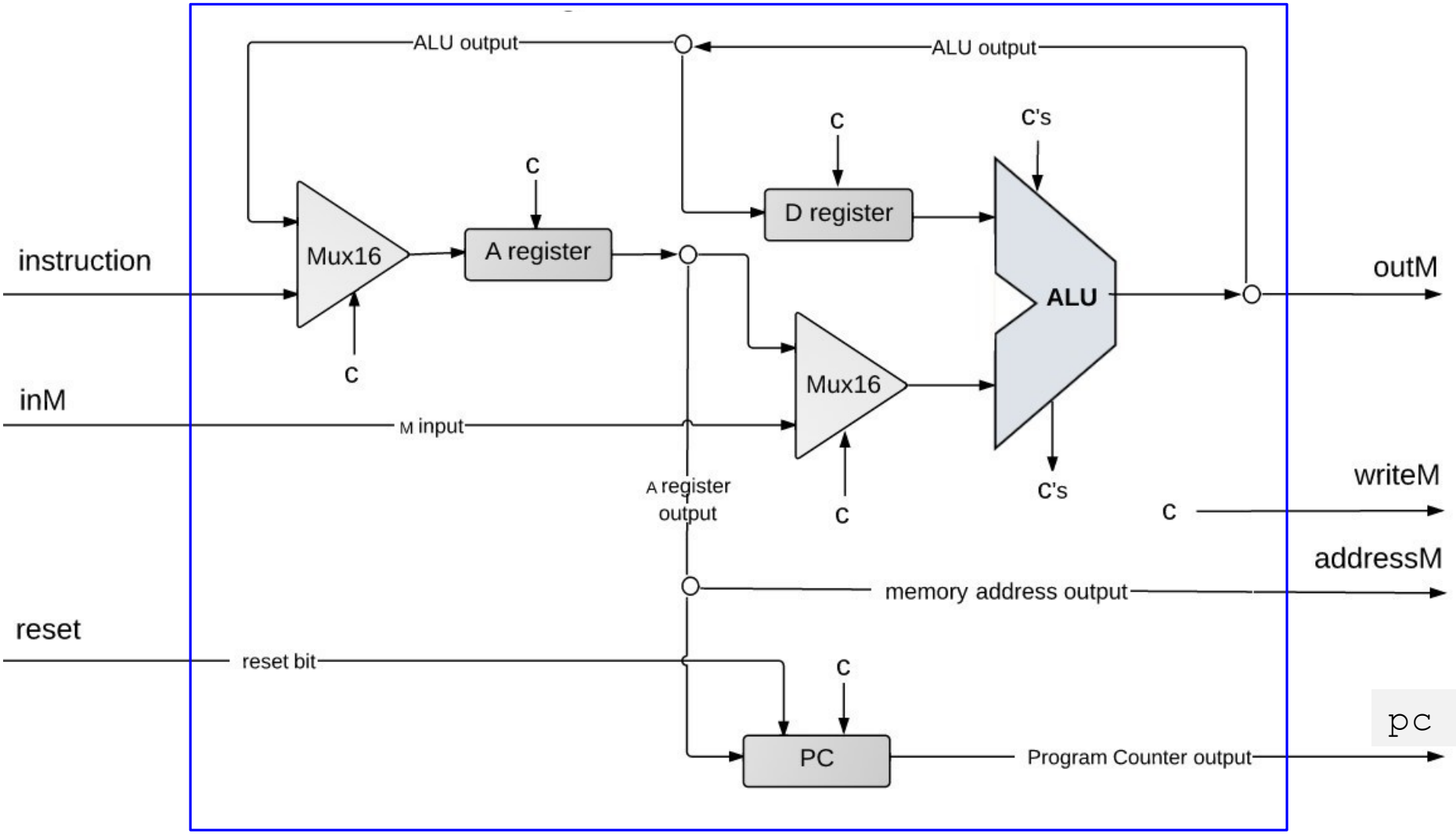


About this Module

UNIVERSITY OF LEEDS

PART 4:

- Building our computer (computer architecture)

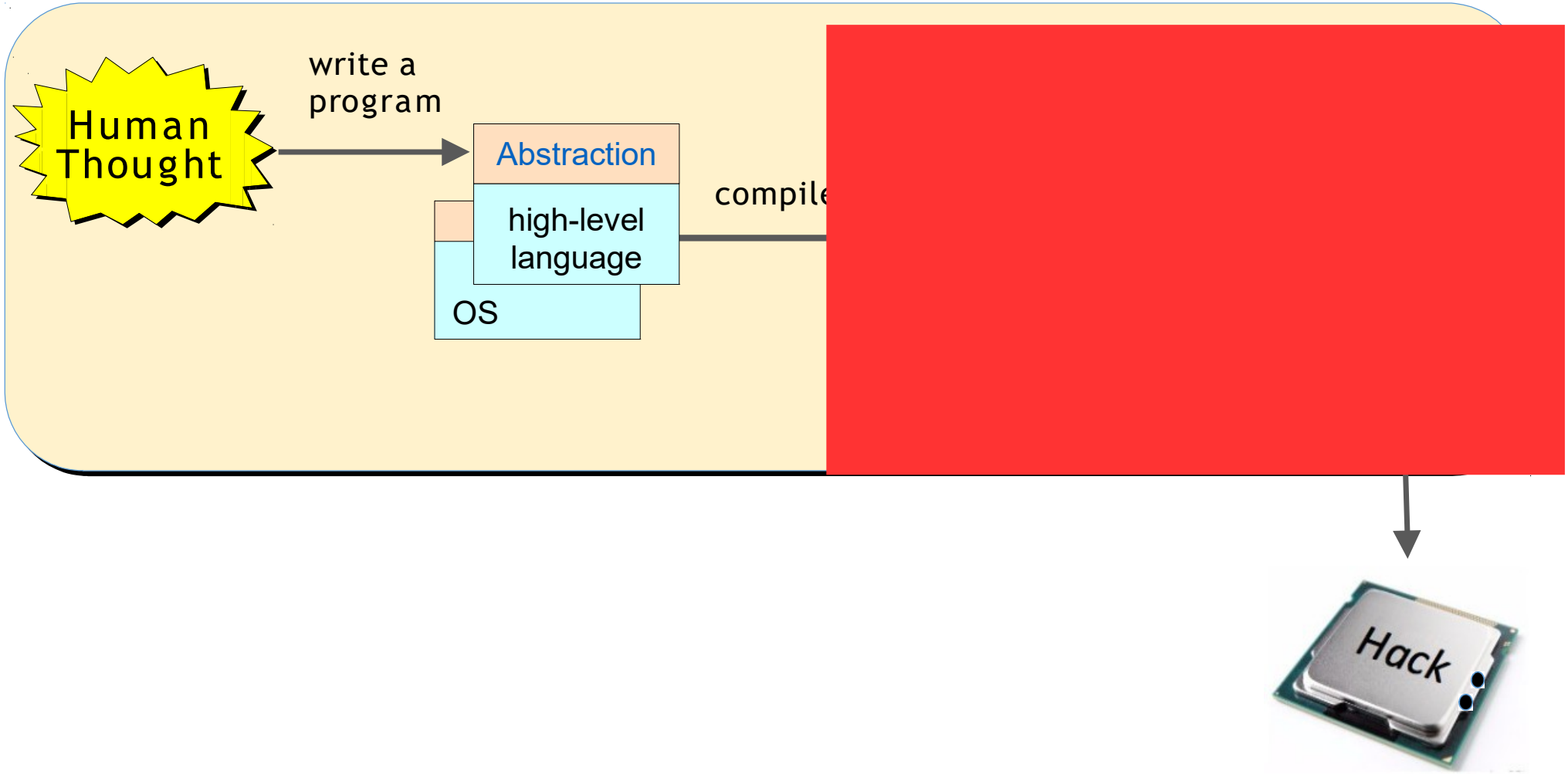


About this Module

UNIVERSITY OF LEEDS

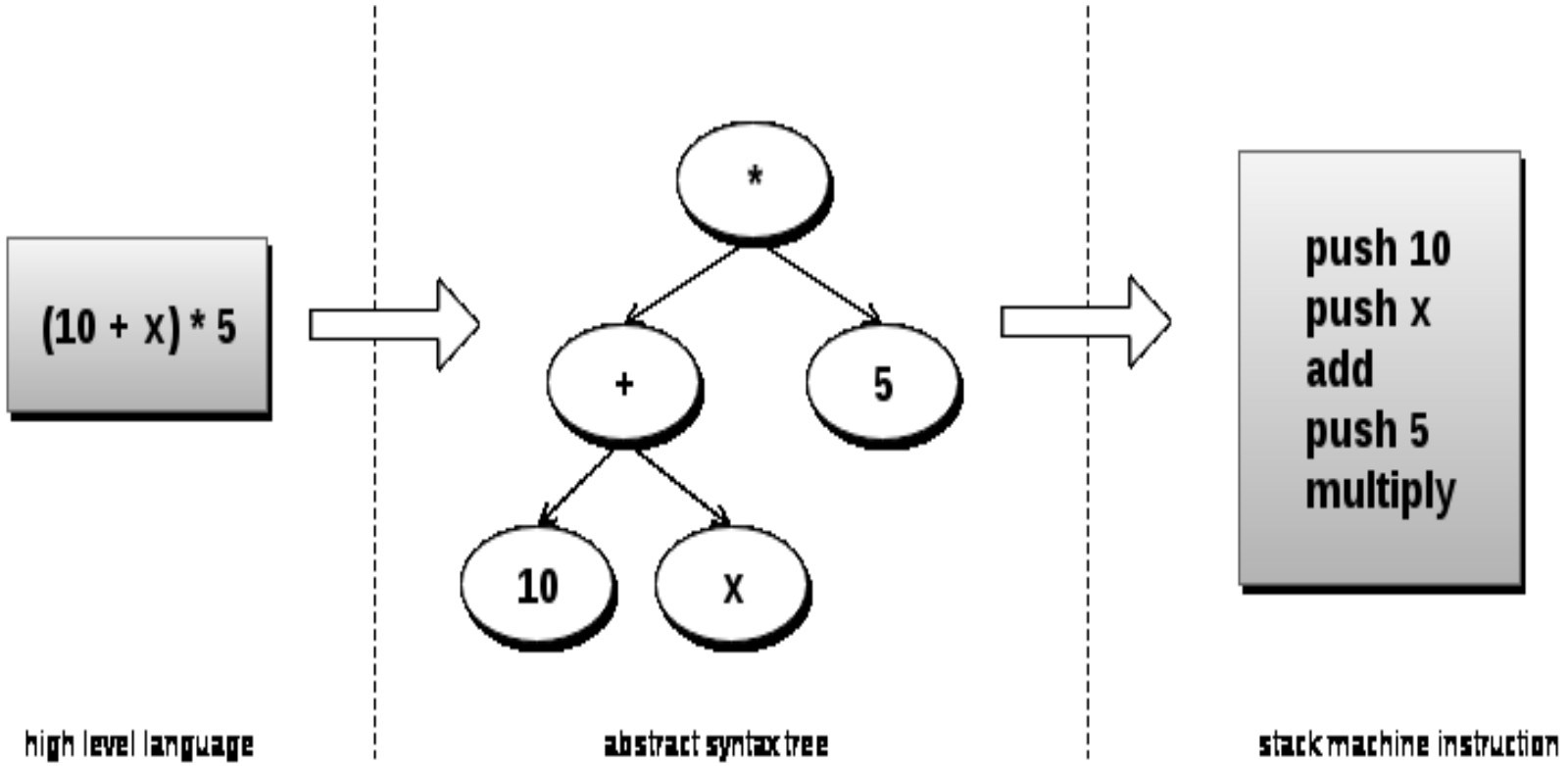
PART 5:

- Connecting the software to our computer (Assembler, Virtual Machine)



Compilation overview

UNIVERSITY OF LEEDS

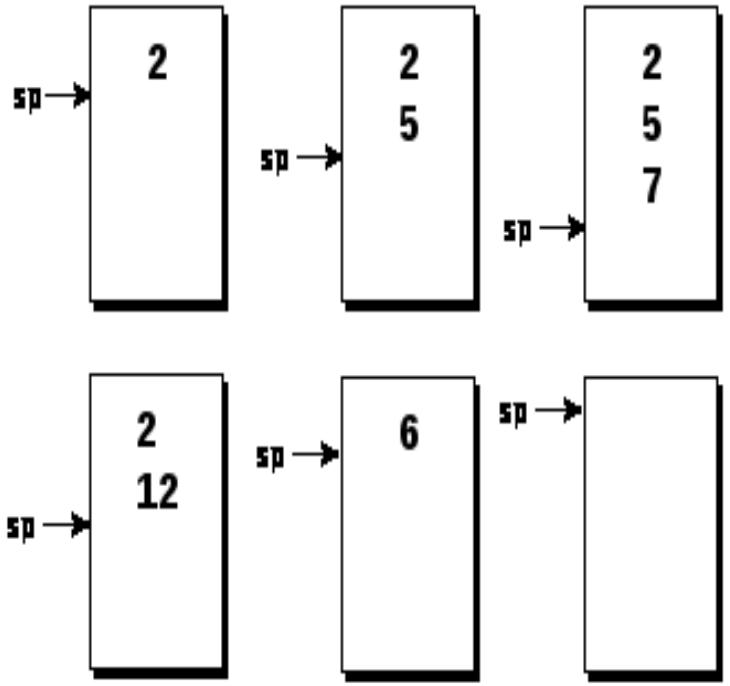


Virtual Machine (VM)

UNIVERSITY OF LEEDS

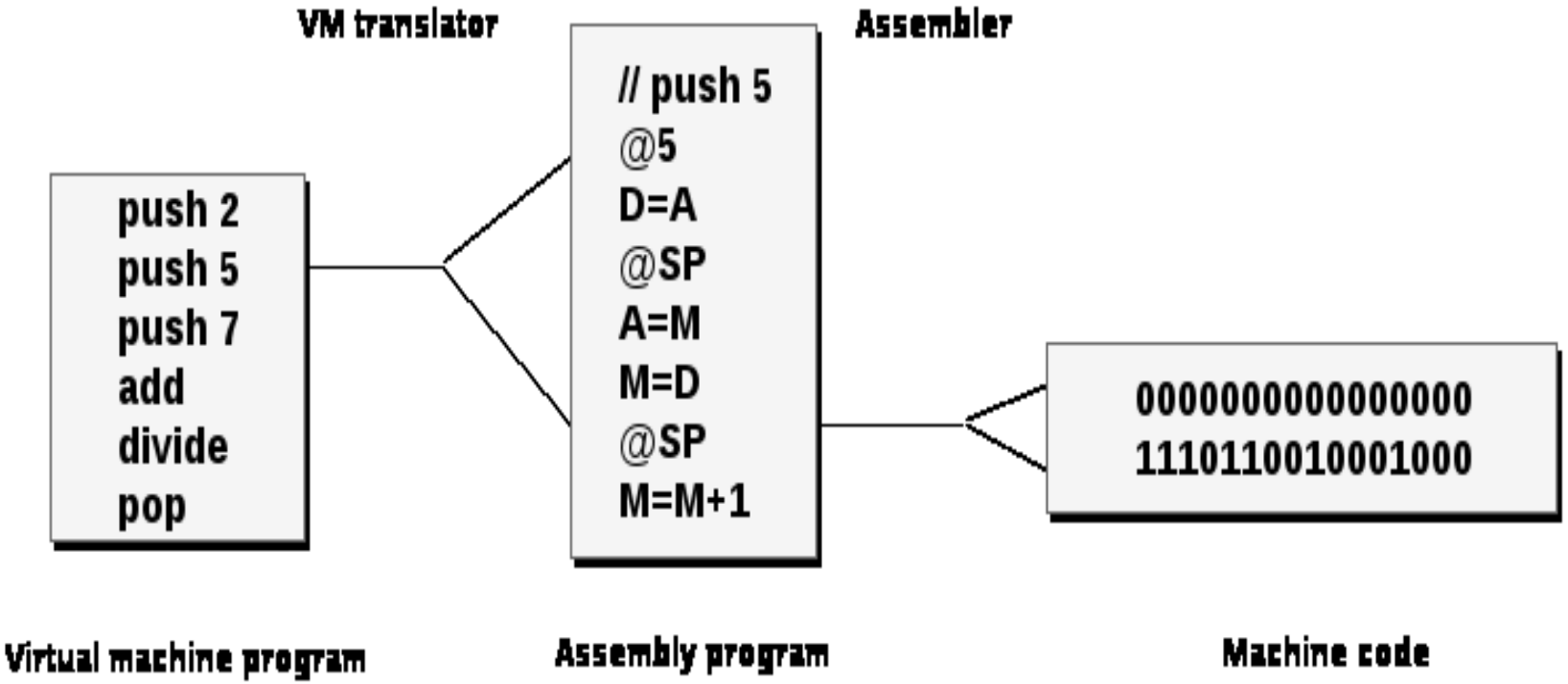
- Stack based virtual machine
  - All operations manipulate the stack
- Operands are pushed onto the stack
- Results are popped off the stack
- Operations consume items from the stack and push their result onto the stack.

```
push 2
push 5
push 7
add
divide
pop
```



Low level programming

UNIVERSITY OF LEEDS



About this Module

UNIVERSITY OF LEEDS

PART 1:

- Understanding Boolean Logic and Boolean Gates
- Hardware Description Language

PART 2:

- Building blocks of the computer (Arithmetic Logic Unit, Memory, CPU)

PART 3:

- Machine Language (binary language understood by our computer)

PART 4:

- Building our computer (computer architecture)

PART 5:

- Connecting the software to our computer (Assembler, Virtual Machine)