



## **Laboratory work №3**

### Week 3

---

Use LMC

# Software: machine code

- Software - code that runs on the hardware
- CPU implements "machine code" instructions
- Each machine code instruction is extremely simple
  - e.g. add 2 numbers
  - e.g. compare 2 numbers

# Software: machine code

The language of the machine code is hardwired into the design of the CPU

- it is not something that can be changed
- Each family of compatible CPUs (e.g. the very popular Intel x86 family) has its own machine code which is not compatible with the machine code of other CPU families.

# Lyrics

CPU is capable of performing simple instructions if you load them into RAM. For example addition/subtraction of two numbers, jumping to another instruction. While CPU is performing an instruction the temporary results are stored inside CPU itself in the fastest, smallest memory locations – registers. CPU starts by fetching an instruction from RAM and then executes/performs it. Then again fetches and executes... This is all CPU does. Initially all programs are on hard disk. When user double-clicks a program icon the machine instructions of that program get loaded to RAM where CPU can access them and execute.

# What are the instructions?

CPU has list of defined instructions, such as:

- add values
- store values
- copy values
- increment value
- go to command

## Example

- Signals sent to CPU
- (10110000 01100001) (read in hex B0 61)
- B0 means “Move a copy of the following value into AL (place in memory)”
- Value in AL memory is equal to 61 in hexadecimal

# What is assembly language

- CPU understands only electrical signals, such as:  
(10110000 01100001)
- But to be understandable to programmers, assembly languages were created.
- Assembly languages use words instead of binary commands, as: MOV AL, 61h

# Little Man Computer

- The Little Man Computer (LMC) is an instructional model of a computer.
- The LMC is generally used to teach students, because it models a simple architecture computer - which has all of the basic features of a modern computer.
- It can be programmed in machine code or assembly code.



# LMC

## Assembly Language Code

## OUTPUT

V1.3 **Little Man Computer**

**CPU**

00 PROGRAM COUNTER

---

INSTRUCTION REGISTER

ADDRESS REGISTER

---

ACCUMULATOR

000

**ARITH-METIC UNIT**

**INPUT**

**RAM**

0	1	2	3	4	5	6	7	8	9
000	000	000	000	000	000	000	000	000	000
10	11	12	13	14	15	16	17	18	19
000	000	000	000	000	000	000	000	000	000
20	21	22	23	24	25	26	27	28	29
000	000	000	000	000	000	000	000	000	000
30	31	32	33	34	35	36	37	38	39
000	000	000	000	000	000	000	000	000	000
40	41	42	43	44	45	46	47	48	49
000	000	000	000	000	000	000	000	000	000
50	51	52	53	54	55	56	57	58	59
000	000	000	000	000	000	000	000	000	000
60	61	62	63	64	65	66	67	68	69
000	000	000	000	000	000	000	000	000	000
70	71	72	73	74	75	76	77	78	79
000	000	000	000	000	000	000	000	000	000
80	81	82	83	84	85	86	87	88	89
000	000	000	000	000	000	000	000	000	000
90	91	92	93	94	95	96	97	98	99
000	000	000	000	000	000	000	000	000	000

**Modifying Program Area**

©GCSEcomputing.org.uk and Peter Higginson

# Set of instructions:

Mnemonic Code	Numeric Code	Instruction
INP	901	Input data
ADD	1XX	Add data
SUB	2XX	Subtract data
STA	3XX	Store data
LDA	5XX	Load data
BRA	6XX	Branch to specified cell
BRZ	7XX	If 0, branch to a specified cell
BRP	8XX	If 0 or positive, branch to a specified cell
OUT	902	Output data

# Try it: Little Man Computer

- <http://robewriter.info/little-man-computer/>
- <http://peterhigginson.co.uk/LMC/>

Battle Tank:

- <http://pddring.github.io/cpu-battle-tank/>

# Practice

- T1: Read two numbers and output their sum and difference
- T2: Enter two numbers, and output them in ascending order
- T3: Read three numbers and output them in ascending order

# Deadline

- Laboratory work 3 deadline 07.02.21 at 23.59 pm.

# Questions?

