EE323: Microprocessor Systems Design
Lab N°1

Getting Started with MDA-WinZ80 Training Kit

1 Objective

The objective of this lab is to introduce theMDA-WinZ80 training kit and to get familiar with writing, assembling and executing assembly programs. MDA-WinZ80 kit is a low-cost single board computer designed for self-learning the Z80 microprocessor.

2 Before the Lab

1. Download and install the Z80 simulator IDE to your own computer (you can find the in EE321 course website). In the installed folder, open "*z80gettingstarted.pdf*" file and go through example 1.
2. Download from EE321course website "MDA-WinZ80 User Manual" and read sections 1 and 2 of the manual.
3. Give the HEX code of the programs of tasks 1 and 2. You can convert the assembly program using Z80 simulator or Z80 instruction table. Which operation is implemented by this program? What is the content of A at the end of the program?
4. Test the program of task 2 using Z80 simulator. Discuss the affected flags.
5. Write a program for the task 3 (32-bit addition). Test your program using Z80 simulator. Give the HEX code of your program.

3 Lab grading

This lab is scheduled for **1 week (3 hours)** and will be graded based on:

1) Preparation.
2) Completion of the experiment, obtained results and their interpretation. During the lab, you must show the obtained results to your teacher. Before moving to a new lab experiment (task), you have to demonstrate that your implementation is working correctly, and you must show your understanding of the experiment. To evaluate your comprehension; you may have to answer oral questions.
3) Answers to the questions.
4) Attendance to the lab.
5) Report sheet.

For this lab, a report sheet will be used as lab report. You must put all your experiment results and the answers to the questions on this sheet and **return it back at the end of the lab**. When you finish a task and before moving to the next task, you should get your teacher's signature for the results of the finished task. To evaluate your comprehension, during the demonstration, you may have to answer other questions not included in the lab assignment.

4 MDA-WinZ80 kit

The MDA-WinZ80 kit is a low-cost single board computer designed for self-learning the popular Z80 Microprocessor. The kit enables studying low level programming with direct machine code entering. A nice feature, single-step running, helps students learn the operation of microprocessor instructions quickly and clearly. The user registers provide simple means to verify the code execution. To interact with the MDA-WinZ80 kit, we can use the following main components: LCD display, keypad, Speaker, RS-232, dot matrix LED, AD and DA converters, DC motor, stepping motor, and I/O ports.

4.1 LCD Display

The MDA-WinZ80 kit has LCD display which shows the address/data content of the memory, and displays messages and the content of the Z80 registers.

4.2 Keypad

The MDA-WinZ80 kit has 16-key HEX digit (0-F) and 8 function keys:

- **RES:** This key causes a hardware reset and starts the monitor.
- **AD:** This key is used to set the current memory address**.**
- **DA:** This key is used to update the data of the current memory address**.**
- **REG:** This key allows examining the CPU Registers**.**
- **GO:** This key allow running the programs at full speed**.**
- **STP:** This key allows running the programs, one instruction at a time.
- **+/-:** These keys increment/decrement the current memory address.

4.3 I/O Port Map

Address	I/O port	Description
00H – 03H	LCD	00H: Instruction register 01H: Data register
04H – 0BH	Keypad	Read key input and write key flag
0CH – 0FH	8251A (RS-232)	0CH: Data register 0EH: Control/status register
10H – 13H	PPI (8255A)	10H/11H/12H: A/B/C port registers 13H: control register
14H – 17H	PIO	14H/15H: A/B port data registers 16H/17H: A/B port control registers
18H – 1BH	Counter Timer Circuit	18H/19H/1AH/1BH: Channels 1/2/3/4
1CH – 1FH	A/D converter and DOT-matrix	1CH/1DH/1EH: A/B/C port registers 1FH: control register
20H - 3FH		I/O extend connector
40H - FFH		User’s range

4.4 Memory Map

Location	Description
0000 – 1FFF	8K Monitor ROM
2000 – 3FFF	8K program and data RAM
4000 - FFFF	Users range

5 Tasks

5.1 Task 1: First Z80 Assembly Program

The objective of this task is to get started with the MDA-WINZ80 kit by loading and executing a simple Z80 assembly program.

```
LD A, 0FH
LD B, 0AH
LD C, 04H
ADD A, B
SUB C
HALT
```

Figure 1: First Z80 assembly program.

1) To load into the memory starting at address 2000H the HEX code of the simple assembly program, you should press the following sequence of the keys: **AD, 2000 DA 3E+0F+06+0A+0E+05+80+91+71+**
2) To check your program, press **AD 2000 + + … +**
3) To execute all the program, you should press **AD 2000 GO**.
4) To execute the program step by step, you should press **AD 2000 STP STP … STP**. After each step, the content of the registers are displayed and you can move between registers using **+** and **-** Keys.
5) Execute the program step by step and give the content of register A and the status of the flags after each instruction. Discuss the obtained results?
6) ***Get your results checked by your teacher.***
7) Give the address and the size of each instruction. What is the size of the program?
8) Which memory location should be modified to test your program with another value of register C. Which sequence of the keys we should press?

5.2 Task 2: Memory Load and Arithmetic Operations

The objective of this task is to get familiar with memory load instruction, arithmetic operations and flags.

```
LD  IX, 3000H
LD  A, (IX+0)
ADD A, (IX+1)
LD  (IX+2),A
HALT
```

Figure 2: Task 2 assembly program.

1) Explain the function of the program of Figure 2. Give the HEX code of the program
2) Load the program and execute it. Explain the obtained result.
3) Using **AD** and **DA** keys, modify the content of locations **3000H** and **3001H** to **0FH** and **01H** respectively. Test the program again and explain the obtained result.
4) Using step by step execution, give the status of the flags (C,V,N,Z and H) after by the execution of the ADD instruction. Justify the obtained result.
5) Repeat steps 3) and 4) with the values **FFH** and **01H**.
6) Repeat steps 3) and 4) with the values **72H** and **13H**.
7) ***Get your results checked by your teacher.***
8) Explain how we can modify this program to use HL pair register instead of IX register.

5.3 Task 3: 32-bit addition

Z80 is an 8-bit processor; it works mainly with 8-bit data and the arithmetic and logical instructions work with 8-bit data, the 16-bit arithmetic operations are provided to process 16-bit memory addresses. The objective of this task is to learn how to process multi-byte data. For the ADD operation,

1) Write a program to add two 32-bit numbers stored at locations 3000-3003H and 3004-3007H and save the result at location 3008-300BH. For the stored 32-bit numbers, you should use Little-Endian order, the least-significant byte is stored first (e.g. 3000H) and the most-Significant byte is stored last (e.g. 3003H).
2) Assemble and test your program using Z80 simulator.
3) Load the program and execute it. Explain the obtained result.
4) Using **AD** and **DA** keys, modify the content of locations **3000-3008H** to test your program with the following numbers: **0FFFFFFFH** and **00000002H**. Test the program with the new numbers and explain the obtained result.
5) Repeat step 3) with the values **89ABCDEFH** and **6789ABCDH**.
6) ***Get your results checked by your teacher.*** Put your assembly program and the HEX code on the report sheet. What is the size of your program?
7) Explain how you can modify your program to add two 32-bit numbers without using ADC instruction.