

Programming in Assembler

Laboratory manual

Exercise 1

Installing MASM 6.14 with Programmers Workbench environment



During the Exercise No.1 students are to install and configure Microsoft assembler MASM 6.14 with full environment for program writing and debugging.

Environment consists of:

- The Microsoft Macro Assembler (MASM)
- Editor
- Debbger CodeView
- A project management („make”) utility
- A source-level browser
- A complete online reference system

All parts of this environment should be installed. They all are intergated with one shell tool - The Programmer's WorkBench (PWB).

To install the MASM 6.14 environment the following are needed:

- Instalation version of the MASM 6.11
- Patch to update the MASM 6.11 to 6.14
- Documentation of the environment

All needed materials can be accessed from location **D:\lab_assembler**

Step by step guide for installation:

1. Read the documents in **Getting Started** folder – they describe installation process
2. Read the packing.txt file – it describes files to be installed
3. Install the MASM 6.11 running **setup.exe** from **DISK1** folder with default parameters
4. Refer to documentation for help about installing parameters
5. Unpack the MASM 6.14 update running the **ml614.exe**
6. Refer to documentation in **readme.txt** file for updating process
7. Update MASM 6.11 to 6.14 running the **patch.exe**

To run the PWB environment first set the environment variables using the **new-vars.bat** located in the BINR directory. Now it is possible to correctly run the **pwb.exe**.

After installation test the functions of the tools writing simple assembler program:

```
TITLE HELLO
.MODEL small, c, os_dos           ; Could be any model except flat
.DOSSEG                          ; Force DOS segment order
.STACK
.DATA                             ; Data segment
    msg BYTE "Hello, world.", 13, 10, "$"
.CODE                             ; Code segment
.STARTUP                          ; Initialize data segment and
                                ; set SS = DS
    mov ah, 09h                   ; Request DOS Function 9
    mov dx, OFFSET msg            ; Load DX with offset of string
                                ; (segment already in DS)
    int 21h                       ; Display String to Standard Out
.EXIT 0                           ; Exit with return code 0
END
```