*Пример программы*

// Компиляция:

// nasm -f bin test.asm -o test.com

global main

bits 16

org 0x100

main:

mov ax, cs

mov ds, ax // mov ds,cs

mov ax, 0xB800

mov es, ax // mov es, 0xb800

mov ax, 0x1F20

mov di, 0

mov cx, 80\*25

rep stosw

mov si, message

mov di, 208

m1: lodsb

test al, al

jz stop

stosw

jmp m1

stop: ret

message db "Hello World!!!", 0x00

NOP

EAX

int M[16];

BYTE – 8 bit

WORD – 2 BYTE – 16 BYTE

DWORD – 4 BYTE – 32 BYTE

CS:EIP – адрес кода операции

SS:ESP – адрес вершины стека

DS:<адрес> - данные

ES:<адрес> - данные

if (x==1)

{

<op>

}

cmp DS:<x>, 5 // ZF=1, ZF=0

jz <adr>

int x; // x <- 0x00000000

int y; // y <- 0x00000004

int M[16] // M <- 0x0000008

int z; // z <- 0x00000018

Операционная система – level = 0

пользовательских\е – level = 3

add

mov

команда директива

NASM

MASM

[<метка>] <операция> [<операнды>]

nop

push eax

mov eax, 1

mov DS:[756], 0x56

mov ax, 1

mov ah, 1

1. непосредственный операнд
2. регистр EAX EBX ECX EDX EDI ESI EBP ESP CS SS DS ES FS GS
3. адресный операнд ES:[EBP+ESI+BX+0x100]

mov ebx, [5+esi]

push ebx

m1: push [1000]

jmp m1

команды:

1. арифметические и логические
2. пересылки данных
3. передачи управления
4. организация цикла

asm16.txt

stosw =>

mov ES:[DI], ax

add di, 2

lodsb =>

mov al, ds:[si]

inc si