# МИНОБРНАУКИ РОССИИ

**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

# «ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)

**Кафедра МО ЭВМ**

# ОТЧЕТ

**по лабораторной работе №1**

# по дисциплине «Организация ЭВМ и систем»

**Тема: Трансляция, отладка и выполнение программ на языке программирования Ассемблер.**

Студентка гр. 0382 Морева Е.С.

Преподаватель Ефремов М.А.

Санкт-Петербург 2021

# Цель работы.

Изучить работу программ на языке Ассемблер, выполнитьих трансляцию, отладку и компоновку.

# Задание.

Часть 1.

1. Просмотреть программу hello1.asm, которая формирует и выводит на экран приветствие пользователя с помощью функции ОС MSDOS, вызываемой через прерывание с номером 21H (команда Int 21h). Выполняемые функцией действия и задаваемые ей параметры — следующие:
   * обеспечивается вывод на экран строки символов, заканчивающейся знаком "$";
   * требуется задание в регистре ah номера функции, равного 09h, а в регистре dx
   * смещения адреса выводимой строки;
   * используется регистр ax и не сохраняется его содержимое.
2. Разобраться в структуре и реализации каждого сегмента программы. Непонятные фрагменты прояснить у преподавателя. Строку-приветствие преобразовать в соответствии со своими личными данными.
3. Загрузить файл hello1.asm из каталога Задания в каталог Masm.
4. Протранслировать программу с помощью строки

>masm hello1.asm

c созданием объектного файла и файла диагностических сообщений (файла листинга).

Объяснить и исправить синтаксические ошибки, если они будут обнаружены транслятором. Повторить трансляцию программы до получения объектного модуля.

1. Скомпоновать загрузочный модуль с помощью строки

>link hello1.obj

с созданием карты памяти и исполняемого файла hello1.exe.

1. Выполнить программу в автоматическом режиме путем набора строки

>hello1.exe

убедиться в корректности ее работы и зафиксировать результат выполнения в протоколе.

1. Запустить выполнение программы под управлением отладчика с помощью команды

>afd hello1.exe

Часть 2

Выполнить пункты 1 - 7 части 1 настоящего задания применительно к программе hello2.asm, приведенной в каталоге Задания, которая выводит на экран приветствие пользователя с помощью процедуры WriteMsg, а также использует полное определение сегментов. Сравнить результаты прогона под управлением отладчика программ hello1 и hello2 и объяснить различия в размещении сегментов.

# Выполнение работы.

Часть 1:

1. Прочитан и проанализирован код программы HELLO1.ASM.
2. Приветственная строка преобразована в соответствии с личными данными.
3. С помощью строки:

>masm hello1.asm

Выполнена трансляция HELLO1.ASM → HELLO1.OBJ, был создан файл листинга HELLO1.LST. Синтаксических ошибок не обнаружено.

1. Произведена компоновка созданного объектного файла, созданы HELLO1.MAP и исполняемый файл HELLO1.EXE. C помощью строки

>link hello1.obj

1. C помощью строки:

> hello1.exe

Был выполнен запуск файла в автоматическом режиме, в результате чего на экран вывелось «Hello world, by Moreva Elena».

1. C помощью команды:

>afdpro hello1.exe

был запущен отладчик, и пошагово отсмотрен исполняемый файл программы.

Часть 2:

1. Прочитан и проанализирован код программы HELLO2.ASM.
2. Приветственная строка преобразована в соответствии с личными данными.
3. С помощью строки:

>masm hello2.asm

Выполнена трансляция HELLO2.ASM → HELLO2.OBJ, был создан файл листинга HELLO2.LST. Синтаксических ошибок не обнаружено.

1. Произведена компоновка созданного объектного файла, созданы HELLO2.MAP и исполняемый файл HELLO2.EXE. C помощью строки

>link hello2.obj

1. C помощью строки:

>hello2.exe

Был выполнен запуск файла в автоматическом режиме, в результате чего на экран вывелось «Hello Worlds! \n Student from 0382 — Moreva Elena».

1. C помощью команды:

>afdpro hello2.exe

был запущен отладчик, и пошагово отсмотрен исполняемый файл программы.

Программный код см. в приложении А. Листинги программ см. в приложении Б.

Таблица 1 – Результаты отладки программы части 1 (HELLO1.EXE). Начальные значения сегментных регистров: (CS) = 1A05, (DS) = 19F5,(ES) = 19F5, (SS) = 1A0C.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Адрес команды | Символический код команды | 16-рич- ный код команды | Содержимое регистров и ячеек  памяти | |
| До выполнения | После выполнения |
|  |  |  | (AX) = 0000 | (AX) = 1A07 |
| (DX) = 0000 |
| 0010 | MOV AX, 1A07 | B8071A | (DX) = 0000 (DS) = 19F5 |
| (DS) = 19F5 |
|  |  |  | (IP) = 0010 | (IP) = 0013 |
|  |  |  | (AX) = 1A07 | (AX) = 1A07 |
| (DX) = 0000 |
| 0013 | MOV DS, AX | 8ED8 | (DX) = 0000  (DS) = 1A07 |
| (DS) = 19F5 |
|  |  |  | (IP) = 0013 | (IP) = 0015 |
|  |  |  | (AX) = 1A07 | (AX) = 1A07 |
| (DX) = 0000 |
| 0015 | MOV DX, 0000 | BA0000 | (DX) = 0000  (DS) = 1A07 |
| (DS) = 1A07 |
|  |  |  | (IP) = 0015 | (IP) = 0018 |
| 0018 | MOV AH, 09 | B409 | (AX) = 1A07 | (AX) = 0907 |
| (DX) = 0000 | (DX) = 0000 |
| (DS) = 1A07 | (DS) = 1A07 |
| (IP) = 0018 | (IP) = 001A |
| 001A | INT 21 | CD21 | (AX) = 0907 | (AX) = 0907 |
| (DX) = 0000 | (DX) = 0000 |
| (DS) = 1A07 | (DS) = 1A07 |
| (IP) = 001A | (IP) = 001C |
|  |  |  | (AX) = 0907 | (AX) = 4C07 |
| 001C | MOV AH, 4C | B44C | (DX) = 0000  (DS) = 1A07 (IP) = 001C | (DX) = 0000 (DS) = 1A07 (IP) = 001E |
|  |  |  | (AX) = 4C07 | (AX) = 0000 |
| 001E | INT 21 | CD21 | (DX) = 0000  (DS) = 1A07 | (DX)=0000  (DS)=19F5 |
|  |  |  | (IP) = 001E | (IP) = 0010 |

Таблица 2 – Результаты отладки программы части 2 (HELLO2.EXE). Начальные значения сегментных регистров: (CS) = 1A0B, (DS) = 19F5,(ES) = 19F5, (SS) = 1A05.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Адрес команды | Символическийкод  команды | 16-рич- ный код команды | жимое регистров и ячеекпамяти | |
| До выполнения | После выполнения |
| 0005 | PUSH DS | 1E | (AX) = 0000  (DX) = 0000 (DS) = 19F5 (IP) = 0005 (SS) = 1A05 (SP) = 0018  Stack: +0 0000 | (AX) = 0000  (DX) = 0000 (DS) = 19F5 (IP) = 0006 (SS) = 1A05 (SP) = 0016  Stack: +019F5 |
| 0006 | SUB AX, AX | 2BC0 | (AX) = 0000  (IP) = 0006 | (AX) = 0000  (IP) = 0008 |
| 0008 | PUSH AX | 50 | (AX) = 0000  (IP) = 0008  (SP) = 0016  Stack: +0 19F5 | (AX) = 0000  (IP) = 0009  (SP) = 0014  Stack: +0 0000 Stack: +219F5 |
| 0009 | MOV AX, 1A07 | B8071A | (AX) = 0000  (IP) = 0009 | (AX) = 1A07 (IP) = 000C |
| 000C | MOV DS, AX | 8ED8 | (AX) = 1A07 (DS) = 19F5 (IP) = 000C | (AX) = 1A07 (DS) = 1A07 (IP) = 000E |
| 000E | MOV DX, 0000 | BA0000 | (DX) = 0000 (IP) = 000E | (DX) = 0000  (IP) = 0011 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 0011 | CALL 0000 | E8ECFF | (IP) = 0011  (SP) = 0014  Stack: +0 0000  Stack: +2 19F5 | (IP) = 0000  (SP) = 0012  Stack: +0 0014  Stack: +2 0000  Stack: +4 19F5 |
| 0000 | MOV AH, 09 | B409 | (AX) = 1A07 (IP) = 0000 | (AX) = 0907  (IP) = 0002 |
| 0002 | INT 21 | CD21 | (IP) = 0002 | (IP) = 0004 |
| 0004 | RET | C3 | (IP) = 0004  (SP) = 0012  Stack: +0 0014  Stack: +2 0000  Stack: +4 19F5 | (IP) = 0014  (SP) = 0014  Stack: +0 0000  Stack: +2 19F5 |
| 0014 | MOV DX, 0010 | BA1000 | (DX) = 0000  (IP) = 0014 | (DX) = 0010  (IP) = 0017 |
| 0017 | CALL 0000 | E8E6FF | (IP) = 0017  (SP) = 0014  Stack: +0 0000  Stack: +2 19F5 | (IP) = 0000  (SP) = 0012  Stack: +0 001A  Stack: +2 0000  Stack: +4 19F5 |
| 0000 | MOV AH, 09 | B409 | (AX) = 0907  (IP) = 0000 | (AX) = 0907  (IP) = 0002 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 0000 | MOV AH, 09 | B409 | (AX) = 0907  (IP) = 0000 | (AX) = 0907  (IP) = 0002 |
| 0002 | INT 21 | CD21 | (IP) = 0002 | (IP) = 0004 |
| 0004 | RET | C3 | (IP) = 0004  (SP) = 0012  Stack: +0 001A  Stack: +2 0000  Stack: +4 19F5 | (IP) = 001A (SP) = 0014  Stack: +0 0000  Stack: +2 19F5 |
| 001A | RET Far | CB | (IP) = 001A (SP) = 0014 (CS) =1A0A  Stack: +0 0000  Stack: +2 19F5  Stack: +4 0000 | (IP) = 0000  (SP) = 0018 (CS) =19F5  Stack: +0 0000  Stack: +2 0000  Stack: +4 0000 |
| 0000 | INT 20 | CD20 | (AX) = 0907  (DX) = 0010 (CX) = 007B (DS) = 1A07 (IP) = 0000 (CS) = 19F5 | (AX) = 0000  (DX) = 0000  (CX) = 0000 (DS) = 19F5 (IP) = 0005 (CS) = 1A0B |

# Выводы.

Были освоены навыки работы с программами на языке Ассемблер: их трансляция, отладка, компоновка и выполнение.

# ПРИЛОЖЕНИЕ А ИСХОДНЫЙ КОД ПРОГРАММЫ

Название файла: HELLO1.ASM

; HELLO1.ASM - упрощенная версия учебной программы лаб.раб. N1

; по дисциплине "Архитектура компьютера"

; \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

; Назначение: Программа формирует и выводит на экран приветствие

; пользователя с помощью функции ДОС "Вывод строки"

; (номер 09 прерывание 21h), которая:

; - обеспечивает вывод на экран строки символов,

; заканчивающейся знаком "$";

; - требует задания в регистре ah номера функции=09h,

; а в регистре dx - смещения адреса выводимой

; строки;

; - использует регистр ax и не сохраняет его

; содержимое.

; \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

DOSSEG ; Задание сегментов под

ДОС

.MODEL SMALL ; Модель памяти-

SMALL(Малая)

.STACK 100h ; Отвести под Стек 256 байт

.DATA ; Начало сегмента

данных

Greeting LABEL BYTE ; Текст приветствия DB 'Hello world, by Moreva Elena',13,10,'$'

.CODE ; Начало сегмента кода

mov ax, @data ; Загрузка в DS адреса начала

mov ds, ax ; сегмента данных

mov dx, OFFSET Greeting ; Загрузка в dx смещения

; адреса текста приветствия

DisplayGreeting:

mov ah, 9 ; # функции ДОС печати строки

int 21h ; вывод на экран приветствия

mov ah, 4ch ; # функции ДОС завершения программы

int 21h ; завершение программы и выход в ДОС

END

Название файла: HELLO2.ASM

; HELLO2 - Учебная программа N2 лаб.раб.#1 по дисциплине "Архитектура компьютера"

; Программа использует процедуру для печати строки

;

; ТЕКСТ ПРОГРАММЫEOFLine EQU '$' ; Определение символьной константы

; "Конец строки"

; Стек программы

ASSUME CS:CODE, SS:AStack AStack SEGMENT STACK

DW 12 DUP('!') ; Отводится 12 слов памяти AStack ENDS

; Данные программы DATA SEGMENT

; Директивы описания данных

HELLO DB 'Hello Worlds!', 0AH, 0DH,EOFLine GREETING DB 'Student from 0382 — Moreva Elena$'

DATA ENDS

; Код программы CODE SEGMENT

; Процедура печати строки

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| WriteMsg | PROC | NEAR |  |  |  |  |  |  |  |
|  | mov | AH,9 |  |  |  |  |  |  |  |
|  | int | 21h ; | Вызов | | функции DOS по прерыванию | | | |  |
|  | ret |  |  |  |  |  |  |  |  |
| WriteMsg | ENDP |  |  |  |  |  |  |  |  |
| ; Головная процедура | | | | | | | | | |
| Main | PROC | FAR |  |  |  |  |  |  |  |
|  | push | DS | ;\ |  | Сохранение адреса начала PSP | | | | в стеке |
|  | sub  push | AX,AX  AX | ; >  ;/ | | для последующего восстановления по  команде ret, завершающей процедуру. | | | | |
|  | mov | AX,DATA |  |  |  | ; | Загрузка | сегментного | |
|  | mov | DS,AX |  |  |  | ; | регистра | данных. |  |
|  | mov | DX, OFFSET | | HELLO | | ; | Вывод на | экран первой | |

call WriteMsg ; строки приветствия. mov DX, OFFSET GREETING ; Вывод на экран второй call WriteMsg ; строки приветствия.

ret ; Выход в DOS по команде,

; находящейся в 1-ом слове PSP.

Main ENDP

CODE ENDS

END Main

# ПРИЛОЖЕНИЕ Б

**ДИАГНОСТИЧЕСКИЕ СООБЩЕНИЯ** **(ЛИСТИНГИ)**

Название файла: HELLO1.LST

#Microsoft (R) Macro Assembler Version 5.10 9/21/21 24:07:19

Page 1-1

ы

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ; | HELLO1.ASM - упрощенная версия учебной программы лаб.раб. N1 | | | | | | | |
| ; | по дисциплине "Архитектура компьютера" | | | | | | | |
| ; | \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* | | | | | | | |
| ; | Назначение: Программа формирует и выводит на экран приветствие | | | | | | | |
| ; | пользователя с помощью функции ДОС "Вывод строки" | | | | | | | |
| ; | (номер 09 прерывание 21h), которая: | | | | | | | |
| ; | - обеспечивает выод на экран строки символов, | | | | | | | |
| ; | заканчивающейся знаком "$"; | | | | | | | |
| ; | - требует задания в регистре ah номера функции=09h, | | | | | | | |
| ; | а в регистре dx - смещения адреса выводимой | | | | | | | |
| ; | строки; | | | | | | | |
| ; | - использует регистрax и не сохраняет его | | | | | | | |
| ; | содержимое. | | | | | | | |
| ; | \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* | | | | | | | |
| DOSSEG | | | | |  | |  |  |
| ; Задание сегментов | | | | | под ДОС | |  |  |
| .MODEL SMALL | | | | |  | |  | ; Модель памяти-SMALL(Малая) |
| STACK 100h | | | | |  | |  | ; Отвести под Стек 256 байт |
| .DATA  данных | | | | |  | |  | ; Начало сегмента |
| 0000 | | | | | Greeting | | LABEL | BYTE ; Текст приветствия |
| 0000 D0 | | 92 | D0 | B0 D1  ,'$' | | 81 DB 'Hello world, by Moreva Elena',13,10 | | |
| 20 | | D0 | BF | D1 80 | | D0 | | |
| B8 | | D0 | B2 | D0 B5 | | D1 | | |
| 82 | | D1 | 81 | D1 82 | | D0 | | |
| B2 | | D1 | 83 | D0 B5 | | D1 | | |
| 82 | | 20 | D1 | 81 D1 | | 82 | | |
| 2E | | D0 | B3 | D1 80 | | 2E | | |
| 37 | | 33 | 30 | 33 20 | | 2D | | |

.CODE ; Начало

сегмента кода

#Microsoft (R) Macro Assembler Version 5.10 9/21/21 24:07:19

Page 1-2

0000 B8 ---- R mov ax, @data ; Загрузка в DS адреса

0003 8E D8 mov ds, ax ; сегмента данных

0005 BA 0000 R mov dx, OFFSET Greeting ; Загрузка в dx смещения

; адреса

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| текста | | приветствия | | |  | | |  | | | |
| 0008 | |  | | | DisplayGreeting: | | |
| 0008  С„ | | B4 09 | | | mov ah, | | | 9 | | ; # | |
| 000A | CD | | 21 | функции ДОС | | печати  int | 21h | | ; | | |
| выход  000C | на экран  B4 4C | | | приветствия | | mov | ah, | | 4ch | | ; # |

функции ДОС завершения

000E CD 21 int 21h ;

завершение программы и выход в ДОС

END

#Microsoft (R) Macro Assembler Version 5.10 9/21/21 24:07:19

Symbols-1

Segments and Groups:

N a m e Length Align Combine Class DGROUP . . . . . . . . . . . . . GROUP

\_DATA . . . . . . . . . . . . 0047 WORD PUBLIC 'DATA'

STACK . . . . . . . . . . . . 0100 PARA STACK 'STACK'

\_TEXT . . . . . . . . . . . . . 0010 WORD PUBLIC 'CODE'

Symbols:

N a m e Type Value Attr DIS- PLAYGREETING . . . . . . . . L NEAR 0008 \_TEXT GREETING . . . . . . . . . . . . L BYTE 0000 \_DATA

|  |  |  |  |
| --- | --- | --- | --- |
| @CODE . . . . . . . . . . . . .  @CODESIZE . . . . . . . . . . . | | TEXT  TEXT | \_TEXT  0 |
| @CPU . . . . . . . . . . . . . . | | TEXT | 0101h |
| @DATASIZE . . . . . . . . . . . | | TEXT | 0 |
| @FILENAME . . . . . . . . . . . | | TEXT | hello1 |
| @VERSION . . . . . . . . . . . . | | TEXT | 510 |
| 33 | Source Lines |  | |
| 33 | Total Lines |  | |
| 19 | Symbols |  | |
| 47994 | + 459266 Bytes symbol space | free | |
| 0 | Warning Errors |  | |

0 Severe Errors

Название файла: HELLO2.LST

HELLO2.LST

#Microsoft (R) Macro Assembler Version 5.10 9/21/21 24:14:44

Page 1-1

; HELLO2 - Учебная программа N2 лаб.раб.#1 по дисциплине "архитектура компьютера"

; Программа использует процедуру для печати строки

; ТЕКСТ ПРОГРАММЫ

EOFLine EQU '$' ; Определение символьной константы

; "Конец строки"

; Стек программы

|  |  |  |
| --- | --- | --- |
| 0000 | AStack SEGMENT STACK |  |
| 0000 | 000C [DW 12 DUP(?) ; Отводится | 12 слов памяти] |
| 0018 | AStack ENDS ; Данные программы |  |

0000 DATA SEGMENT ; Директивы описания данных

0000 48 65 6C 6C 6F 20 HELLO DB 'Hello Worlds!', 0AH,

0DH,EOFLine

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 57 | 6F | 72 | 6C | 64 | 73 |  |  |
|  | 21 | 0A | 0D | 24 |  |  |  |  |
| 0010 | 53 | 74 | 75 | 64 | 65 | 6E | GREETING | DB 'Student from 0382 — Moreva |
| Elena$'  74 | | 20 | 66 | 72 | 6F | 65 |  |  |
|  | 20  2D | 30  20 | 33  4D | 38  6F | 32  72 | 20  65 |  |  |
|  | 76 | 61 | 20 | 45 | 6C | 65 |  |  |
| 6e  0031 | | 61 | 24 |  |  |  |  |  |
|  |  |  |  | DATA | ENDS |  |

; Код программы

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 0000 |  |  | CODE | SEGMENT |  |  |
|  |  | ; | Процедура печати строки  WriteMsg PROC NEAR | |  |  |
| 0000 |  |  |
| 0000 | B4 | 09 |  | mov | AH,9 |  |
| 0002 | CD | 21 |  | int | 21h | ; Вызов функции DOS по |
| прерыванию  0004 C3 ret | | | | |  |  |
| 0005 |  |  | WriteMsg | ENDP |  |  |

; Головная процедура

#Microsoft (R) Macro Assembler Version 5.10 9/10/21 00:50:29

Page 1-2

0005 Main PROC FAR

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 0005 | 1E | push DS ;\ Сохранени е адреса начала PSP в стеке | | | | |
| 0006 | 2B | C0 sub AX,AX ; > для последующего восстановления по | | | | |
| 0008 | 50 | push AX ;/ команде ret, завершающей процедуру. | | | | |
| 0009 | B8 | ---- R mov AX,DATA ; Загрузка сегментного | | | | |
| 000C | 8E | D8 | mov DS,AX |  | ; | регистра данных. |
| 000E | BA | 0000 | R mov DX, OFFSET HELLO | | ; | Вывов на экран первой |
| 0011 | E8 | 0000 | R call WriteMsg | |  | ; строеи приветствия. |
| 0014 | BA | 0010 | R mov DX, OFFSET GREETING | | ; | Вывов на экран второй |
| 0017  001A | E8  CB | 0000 | R call WriteMsg  ret | |  | ; строки приветствия.  Выход в DOS по команде, |
|  | ; |
| 001B |  |  | Main | ; находящейся в 1-ом слове PSP.  ENDP | | |
| 001B |  |  | CODE | ENDS |  |  |

END Main

#Microsoft (R) Macro Assembler Version 5.10 9/21/21 24:14:44

Symbols-1

Segments and Groups:

N a m e Length Align Combine Class ASTACK . . . . . . . . . . . . . 0018 PARA STACK

CODE . . . . . . . . . . . . . . 001B PARA NONE

DATA . . . . . . . . . . . . . . 0031 PARA NONE

Symbols:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| N a m e Type | | Value | | Attr | |
| EOFLINE . . . . . . . . . . . . | | NUMBER | | 0024 | |
| GREETING . . . . . . . . . . . . | L | | BYTE | 0010 | DATA |

HELLO . . . . . . . . . . . . . L BYTE 0000 DATA

MAIN . . . . . . . . . . . . . . F PROC 0005 CODE Length = 0016

WRITEMSG . . . . . . . . . . . . N PROC 0000 CODE Length = 0005

@CPU . . . . . . . . . . . . . . TEXT 0101h

@FILENAME . . . . . . . . . . . TEXT hello2

@VERSION . . . . . . . . . . . . TEXT 510

|  |  |  |
| --- | --- | --- |
| 52 | Source Lines |  |
| 52 | Total Lines |  |
| 13 | Symbols |  |
| 47994 | + 459263 Bytes symbol space | free |

0 Warning Errors

0 Severe Errors