**Цель:** освоение средств обеспечивающих модульное программирование.

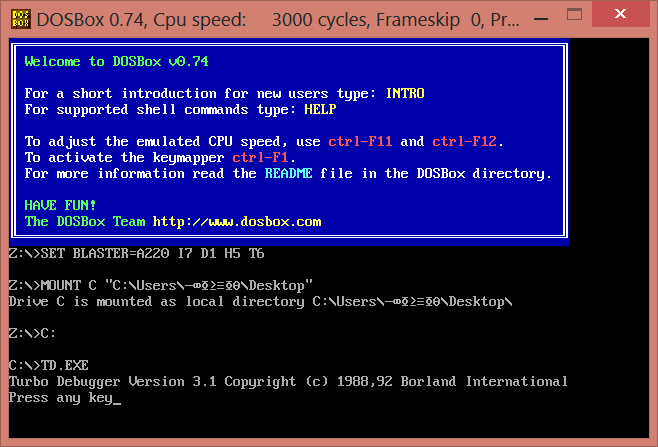
Код основного модуля main, который после запуска выводит на экран сообщение "Press any key". Затем управление передается модулю input, который осуществляет ожидание нажатия клавиши. после нажатия клавиши:

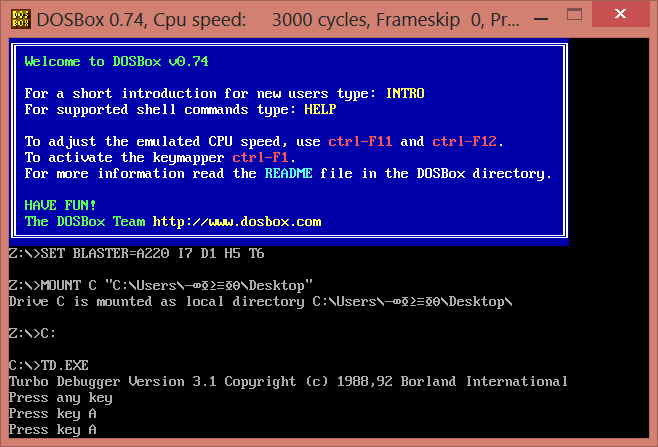
* если была нажата клавиша А, управление передается модулю input1, в котором процедура output1 выводит на экран сообщение ""
* если была нажата любая другая клавиша, управление передается модулю input2, в котором процедура output2 выводит на экран сообщение ""

**Код модулей:**

|  |  |
| --- | --- |
| TITLE MAIN  EXTRN INPUT:PROC  EXTRN OUTPUT1:PROC  EXTRN OUTPUT2:PROC  .Model Small  .stack 100h  .Data  message db 13,10, 'Press any key$'  .Code  begin:  mov ax, @data  mov ds, ax  mov ah,09  lea dx, message  int 21h  cond:  call INPUT ; вызов процедуры INPUT  cmp al,61h ; сравнение нажатой клавиши с А  je lend  call OUTPUT2 ; вызов процедуры OUTPUT2  jmp cond  lend:  call OUTPUT1 ; вызов процедуры OUTPUT1  mov ah,1h  int 21h  mov ax,4c00h  int 21h  end begin | TITLE INPUT  PUBLIC INPUT  .Model Small  .Stack 100h  .Data  .Code  INPUT PROC  mov ax,@data  mov ds,ax  ;ожидание нажатия клавиши  mov ah,1h  int 21h  ret  INPUT endp  End |

|  |  |
| --- | --- |
| TITLE OUTPUT1  PUBLIC OUTPUT1  .model small  .stack 100h  .data  mes1 db 10,13, 'Key A was pressed. Exiting now$'  .code  OUTPUT1 PROC  mov ax,@data  mov ds,ax  ;вывод сообщения 'Key A  was pressed. Exiting now'  mov ah,09h  lea dx,mes1  int 21h  ret  OUTPUT1 endp  end | TITLE OUTPUT2  PUBLIC OUTPUT2  .model small  .stack 100h  .data  mes2 db 10,13, 'Press key A$'  .code  OUTPUT2 PROC  mov ax,@data  mov ds,ax  mov ah,09h  ;вывод сообщения  "Press key A"  lea dx,mes2  int 21h  ret  OUTPUT2 endp  end |

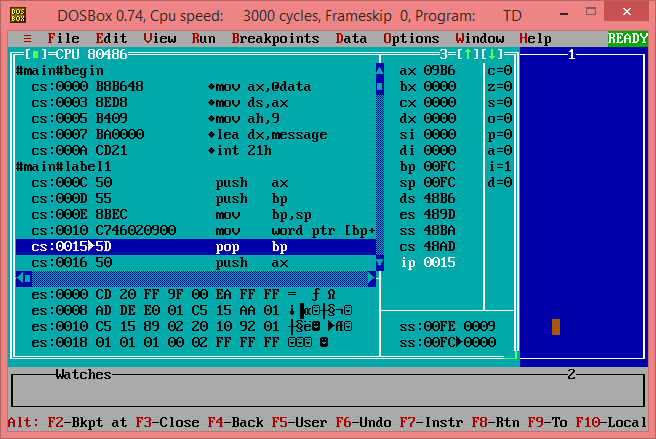




**Модернизация:**

После разработки и отладки рассмотренной выше многомодульной программы требуется модернизировать текст модулей созданной программы таким образом, чтобы обеспечить демонстрацию передачи параметров процедурам через стек. Модернизированную программу необходимо отладить и составить отчет о проделанной работе.

|  |  |
| --- | --- |
| **TITLE MAIN**  EXTRN INPUT:PROC  EXTRN OUTPUT1:PROC  EXTRN OUTPUT2:PROC  .ModelSmall  .stack 100h  .Data  message db 'Press any key$'  .Code  begin:  mov ax,@data  mov ds,ax  mov ah,9  lea dx,message  int 21h  label1:  push 9h ;заносим в стек значение, для того чтобы  push 8h потом извлечь его в регистр ax  **call INPUT**  add sp,2 ;отметка 2  cmp al,61h ;сравнение с ASCII-кодом кнопки «a»  je label\_end  push 9h ;заносим в стек значение, для того чтобы  потом извлечь его в регистр ax  **call OUTPUT2**  add sp,2  jmp label1  label\_end:  push 9h ;заносим в стек значение, для того чтобы  потом извлечь его в регистр ax  **call OUTPUT1**  add sp,2  mov ah,4Ch  int 21h  end begin  **Комментарий:**  В процессе, когда программа дойдет до отметки 2, значение sp будет равно 00FC и для того, чтобы вернуть стек в исходное состояние нам необходимо использовать команду “add sp, 2”. После ее использования, видно, что стек вернулся в исходное состояние:  **OUTPUT 1**  TITLE OUTPUT1  PUBLIC OUTPUT1  .Model small  .stack 100h  .data  mes1 db 10,13,'Key A was pressed. Exiting now$'  .code  OUTPUT1 PROC ;отметка 3  mov ax,@data  mov ds,ax  push bp  mov bp, sp  mov ah,[bp+4]  lea dx, mes1  int 21h  pop bp  ret  OUTPUT1 endp  end  **Комментарий:**  В процессе, когда программа дойдет до отметки 3, значение sp будет равно 00FC , так как происходят аналогичные действия, что и в предыдущем случае. | **TITLE INPUT**  PUBLIC INPUT  .model small  .stack 100h  .data  .code  INPUT PROC ; отметка 1  mov ax,@data  mov ds,ax  push bp  mov bp, sp  mov ah,[bp+4]  int 21h  pop bp  ret  INPUT endp  end  **Комментарий:**  В процессе, когда программа дойдет до отметки 1, значение sp будет равно  **OUTPUT 2**  TITLE OUTPUT2  PUBLIC OUTPUT2  .model small  .stack 100h  .data  mes2 db 10,13,'Press key A$'  .code  OUTPUT2 PROC ;отметка 4  mov ax,@data  mov ds,ax  push bp  mov bp, sp  mov ah,[bp+4]  lea dx, mes2  int 21h  pop bp  ret  OUTPUT2 endp  End  **Комментарий:**  В процессе, когда программа дойдет до отметки 4, значение sp будет равно 00FC , так как происходят аналогичные действия, что и в предыдущих случаях (отметки 2 и 3) |



**Вывод:** в проделанной лабораторной работе было частично изучено модульное программирование и вызов отдельных модулей программы (директивы PUBLIC и EXTRN).

Для вызова процедуры необходима команда call (в процедуре обязательно должна быть команда ret – она возращает программу в то место , где была выполнена команда call). Для реализации этой задачи мы использовали стек. Для работы со стеком мы использовали операции push (положить данные в стек) и pop (взять данные из стека)*.*