**Лабораторная работа №6**

**Цель работы**

Реализация программы, вводящей с клавиатуры значения двух 16-битных регистров (AX и BX) в шестнадцатеричном виде и выводящей их двоичное представление на экран. Реализовать ввод через процедуру с контролем допустимости ввода.

**Решение**  
start:  
 call get\_word ; получить значение для AX  
 mov ax, dx  
 call get\_word ; получить значение для BX  
 mov bx, dx  
 call print\_binary\_ax  
 call print\_binary\_bx  
 int 20h  
  
; Процедура ввода 16-битного числа (4 hex-цифры)  
get\_word:  
 xor dx, dx ; очистить DX  
 mov cx, 4  
next\_digit:  
 call get\_hex ; ввод цифры в AL  
 shl dx, 4 ; сдвиг числа влево на 4 бита  
 or dl, al ; добавить новую цифру  
 loop next\_digit  
 ret  
  
; Процедура get\_hex (ввод hex-цифры с проверкой допустимости)  
get\_hex:  
 push ax  
 push dx  
.wait:  
 mov ah, 8  
 int 21h ; ввод без эха, AL ← символ  
 mov dl, al  
 cmp al, '0'  
 jb .wait  
 cmp al, '9'  
 jbe .dec  
 cmp al, 'A'  
 jb .wait  
 cmp al, 'F'  
 ja .wait  
 sub al, 37h ; A-F → 10–15  
 jmp .ok  
.dec:  
 sub al, 30h ; 0–9 → 0–9  
.ok:  
 mov ah, 2  
 mov dl, dl  
 int 21h ; вывод символа  
 pop dx  
 pop ax  
 ret  
  
; Вывод AX в двоичном виде  
print\_binary\_ax:  
 push ax  
 mov cx, 16  
 mov dx, ax  
 call print\_binary  
 pop ax  
 ret  
  
; Вывод BX в двоичном виде  
print\_binary\_bx:  
 push bx  
 mov cx, 16  
 mov dx, bx  
 call print\_binary  
 pop bx  
 ret  
  
; Универсальная процедура двоичного вывода  
print\_binary:  
 mov ah, 2  
.print\_bit:  
 shl dx, 1  
 jc .one  
 mov dl, '0'  
 jmp .out  
.one:  
 mov dl, '1'  
.out:  
 int 21h  
 loop .print\_bit  
 ret

**Результат**

ВВЕДИТЕ СОДЕРЖИМОЕ РЕГИСТРА AX F46B

ВВЕДИТЕ СОДЕРЖИМОЕ РЕГИСТРА BX 5A0C

(AX) = 1111010001101011

(BX) = 0101101000001100