Лабораторная работа №11

***СИСТЕМНЫЙ ТАЙМЕР***

Выполнил ст. гр. ІО-83, Гулевский Денис

Написать программу для вывода бегущего символа времени(заданного вариантом) с заданным цветом из часов реального времени на экран. Для упрощения задания должен быть выведен только один символ шрифтом ASCII, как показано на Рис. 1 в примере. Задержка при движении бегущего символа(скорость его движения) задана в варианте. Для вывода на экран использовать способ, заданный в варианте. (Если задан способ вывода «Прямая запись в видеопамять», то для вывода нужно использовать, непосредственно запись нужных символов в память видеоадаптера для текстового режима. Адрес видеобуфера как уже известно в данном случае B8000h. задан способ вывода «Посредством прерываний видеосервиса», то для вывода необходимо использовать функции прерывания видео-сервиса INT 10h, как показано в примере.). Выход из программы должен происходить по событию, заданному вариантом.

## Варіанти виконання роботи: 306

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 306 | 2 символ часов | gray | 2 | Посредством прерываний видео сервиса | Через 7 сдвигов символа |

ASSUME CS:CODE, DS:DATA, SS:MYSTACK

.286

ASSUME CS:CODE,DS:DATA,SS:STK

STK SEGMENT STACK

STKD DW 64 DUP(?)

STK ENDS

CODE SEGMENT

\_beg: MOV AX,DATA

MOV DS,AX

mov cx, 7

\_lop:

MOV AX,0003h ;init text mode

INT 10h ;80x25

push cx

MOV AH,02h ;get system time from RTC

INT 1Ah ;CH-hours, CL-minutes DL-seconds

XOR AH, AH

MOV AL, CH

AND AL, 00001111b

MOV SI, AX

ADD X, 5 ;shift of the X coordinate to the next digit

CALL outDigit ;out digit

MOV AH, 86h ;DELAY FOR 2 SEC

MOV DX, 04240h ;DX:CX = 000F4240h

MOV CX, 000Fh ;in microSeconds

INT 15h

pop cx

loop \_lop

mov AX, 4c00h

int 21h

;PROCEDURE outDigit

; SI : CHAR NUMBER OF DIGIT TO OUT

; X,Y : COORDINATES

;OUTPUT THE DIGIT WITH A NUMBER SI AND

;COORDINATES X AND Y

outDigit PROC NEAR

PUSHA

MOV AX,SI ; CALCULATES THE OFFSET

MOV BX,25 ; AFTER THE BEGIN OF

MUL BX ; OUTPUT TABLE.

MOV SI,AX ;CHAR OFFSET

MOV DH,Y ;ROW TO OUT

MOV DL,0 ;COL TO OUT

MOV CX,25

\_out:

PUSH CX

PUSH DX

ADD DL,X

MOV AH,02h ;SET CURSOR

MOV BH,00h ;VIDEO PAGE

INT 10h

MOV AH,09h ;OUT CHAR AT CURSOR

MOV AL,zer+SI ;CHAR

MOV CX,01h ;COUNT

MOV BL,08h ;TEXT ATTRIBUTE(WHITE)

;ADD BL,80h ;FLASHING COLOR

INT 10h

INC SI

POP DX

INC DL

CMP DL, 5

JE \_inc

JMP \_next

\_inc: INC DH ;INCREASE Y POSITION

MOV DL,0 ;"RETURN CARET"

\_next: POP CX

LOOP \_out

POPA

RET

outDigit ENDP

CODE ENDS

DATA SEGMENT

X DB 2 ;X coordinate of the text

Y DB 3 ;Y coordinate of the text

SWITCH DW 10

zer DB '---- '

DB '- - '

DB '- - '

DB '- - '

DB '---- '

one DB '-- '

DB ' - '

DB ' - '

DB ' - '

DB '--- '

two DB '---- '

DB ' - '

DB '---- '

DB '- '

DB '---- '

thr DB '---- '

DB ' - '

DB '---- '

DB ' - '

DB '---- '

fou DB '- - '

DB '- - '

DB '---- '

DB ' - '

DB ' - '

fiv DB '---- '

DB '- '

DB '---- '

DB ' - '

DB '---- '

six DB '---- '

DB '- '

DB '---- '

DB '- - '

DB '---- '

sev DB '---- '

DB ' - '

DB ' - '

DB ' - '

DB ' - '

eig DB '---- '

DB '- - '

DB '---- '

DB '- - '

DB '---- '

nin DB '---- '

DB '- - '

DB '---- '

DB ' - '

DB '---- '

dts DB ' '

DB ' - '

DB ' '

DB ' - '

DB ' '

emp DB ' '

DB ' '

DB ' '

DB ' '

DB ' '

DATA ENDS

END \_beg