

CHAPTER V

Implementation and Testing

5.1. Implementation

This code is the header for the program, the header depends on the compiler used to compile the program, in this case it's TASM.

```
1 assume cs:code, ds:code
2 code segment
3 start:
4     cli
5     mov     ax, code
6     mov     ds, ax
7     push   ds
8     call   initialize ;; draws the blank board, blah blah blah
9     push   ax
0     mov     score,30h
1     call   scoring
2     pop    ax
3     pop    ds
```

Figure 5.1 Header

The start code contains code to call the function initialize (drawing the board) and score to keep track of the score.

This code is delay loop

```
wastesometime:
    push    cx
    mov     cx,0
wasteloop:
    add     cx,1
    cmp     cx,60000
    jnz     wasteloop
    mov     cx,0
wasteloop2:
    add     cx,1
    cmp     cx,60000
    jnz     wasteloop2
    pop     cx
    ret
```

Figure 5.2 Delay Loop

This loop functions as a delay, as the tetra will drop down too fast if the delay is not implemented. The loop works by using a register as a counter to count (in this

case CX) from 0 to 60000 twice in a row thus forcing the computer to complete the process before continuing to the main loop.

This code is inside the main loop, it's function is to check whether the tetra can move down or not.

```
canmovedown:  
    push    si  
    push    cx  
    push    bx  
    push    dx  
    call   showcurrentpiece  
    mov     si,offset currentpiecex  
    mov     cx, 0  
canmovedownloop:  
    mov     bl,[si+4]  
    pop     dx  
    add     bl,dx  
    mov     bh,[si]  
    cmp     dh,2  
    jz     decrementbh  
    add     bh,dh
```

Figure 5.3 Move Down Loop

This code is used to clear the lines when the tetras fill the game box.

```
clearfulls:
    push    si
    push    dx
    push    ax
    push    cx
    push    bx
    mov     bl,19
    mov     bh,1
clearrowloop:
    push    bx
    call    getpixel
    pop     bx
    cmp     al,7
    jnz    failrow
    add     bh,1
    cmp     bh,11
    jnz    clearrowloop
    jmp     sucrow
failrow:
    mov     bh,1
    dec     bl
    cmp     bl,0
    jnz    clearrowloop
    jmp     endclearfulls
sucrow:
    push    bx
clearline:
    call    scoring
    jmp     sucloop
```

Figure 5.4 Clearline Function 1

```

sucloop:
    mov     bh,10
    call   copyabovebelow
    dec    bl
    cmp    bl,0
    jnz    sucloop
    pop    bx
    mov    bh,1
    mov    bl,19
    jmp    clearrowloop
endclearfulls:
    pop    bx
    pop    cx
    pop    ax
    pop    dx
    pop    si
    ret
copyabovebelow:
    push   bx
    dec    bl
    call   getpixel
    inc    bl
    call   setpixel
    pop    bx
    dec    bh
    cmp    bh,0
    jnz    copyabovebelow
    ret

```

Figure 5.5 Clearline Function 2

This is the scoring function

```
scoring:
    push cx
    push ax
    mov ax,0B801h
    mov es,ax
    mov di, 200
    mov al, [score]

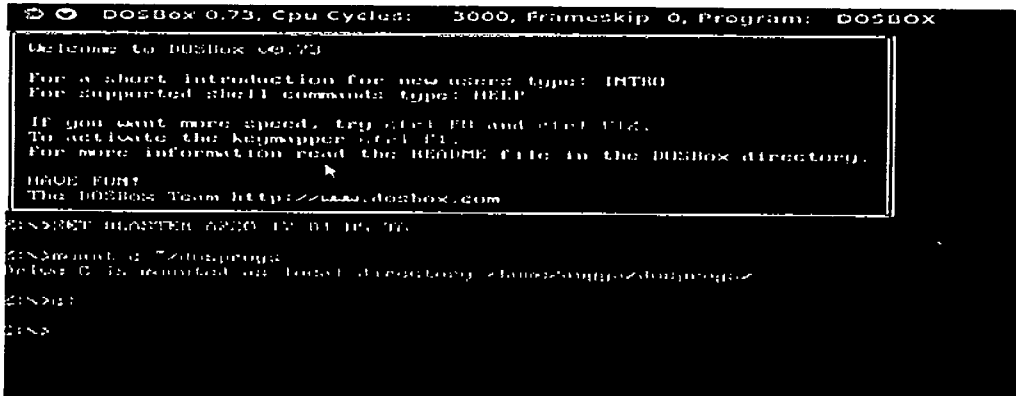
    mov es:[di], ax

    inc al
    cmp al,39h
    jg zeroscore
    mov score, al
    int 21h
    pop ax
    pop cx
    ret
zeroscore:
    mov al,30h
    mov score, al
    pop ax
    pop cx
    ret
```

Figure 5.6 Scoring Function

5.2. Testing

To launch the program the OS must be DOS based (up until windows 98), or a DOS emulator can be used (such as DOSBox). In case of using DOSBox first you have to mount the folder where the program is located like the picture below.



```
DOSBox 0.73, Cpu Cycles: 3000, Frameskip 0, Program: DOSBOX

Welcome to DOSBox 0.73

For a short introduction for new users type: INTRO
For supported shell commands type: HELP

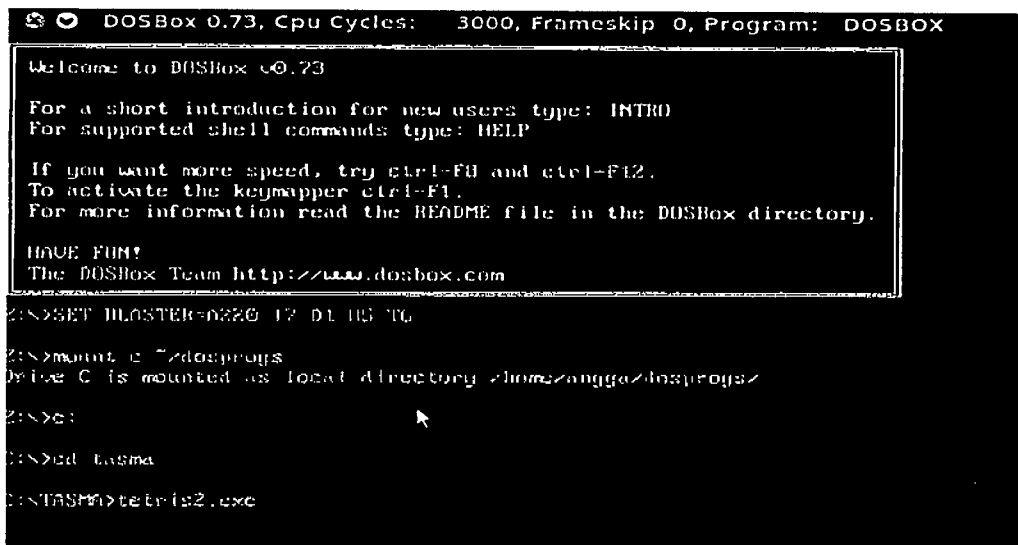
If you want more speed, try ctrl-F8 and ctrl-F12.
To activate the keymapper ctrl-F1.
For more information read the README file in the DOSBox directory.

HAVE FUN!
The DOSBox Team http://www.dosbox.com

Z:\>SET BLASTER=A220 I7 D1 H5 J6
Z:\>mount c /zdosprogs
Drive C is mounted as local directory /home/zangga/zdosprogs/
Z:\>
Z:\>
Z:\>
```

Figure 5.7 Mounting

And then executing the program.



```
DOSBox 0.73, Cpu Cycles: 3000, Frameskip 0, Program: DOSBOX

Welcome to DOSBox 0.73

For a short introduction for new users type: INTRO
For supported shell commands type: HELP

If you want more speed, try ctrl-F8 and ctrl-F12.
To activate the keymapper ctrl-F1.
For more information read the README file in the DOSBox directory.

HAVE FUN!
The DOSBox Team http://www.dosbox.com

Z:\>SET BLASTER=A220 I7 D1 H5 J6
Z:\>mount c /zdosprogs
Drive C is mounted as local directory /home/zangga/zdosprogs/
Z:\>
Z:\>
Z:\>cd tasma
Z:\TASMA>tetris2.exe
```

Figure 5.8 Executing Program

The program will look like this, use A and D to move the tetras and when the tetras fill a line the line will be cleared thus increasing the score



Figure 5.9 Tetris 2