

CHAPTER IV ANALYSIS AND DESIGN

4.1. Analysis

To create this program I've analyzed several things required to complete it. First I need to draw the board (the game field). Then creating the tetras used in this game, there's seven in all. Then be able to move down the tetras in a fixed delay. Then I need to be able to move the tetras left and right with input from keyboard, and then clearing the line by stacking tetras into 1 line or more. And a score function to track the scores, detailed explanation can be found in 4.2 Design.

4.2. Design

The tetris.exe program will look like this,

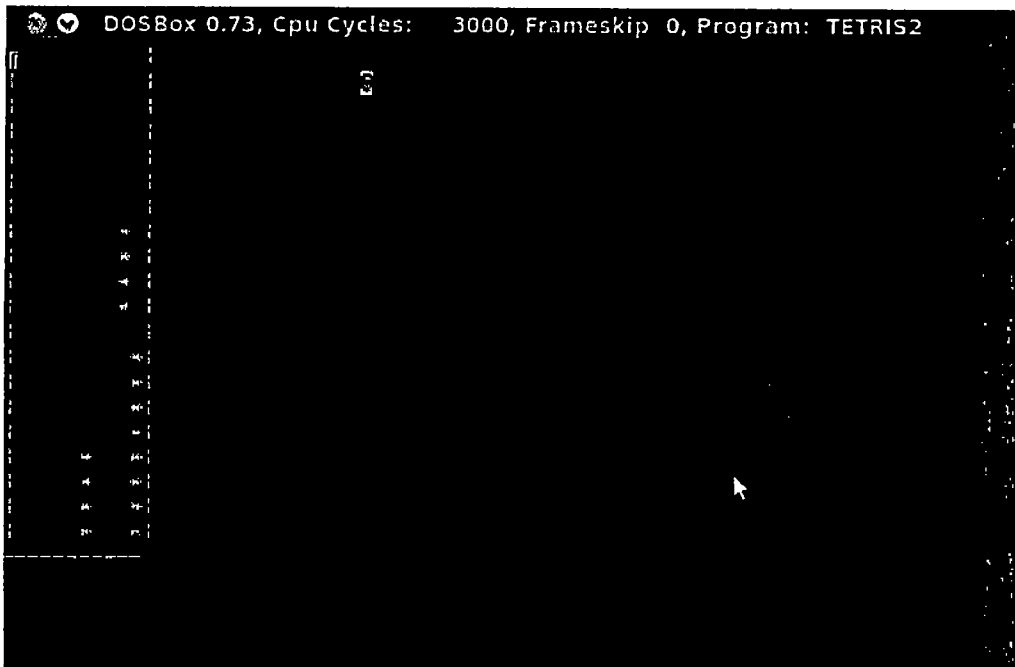


Figure 4.1 Design Program

the glowing 0 on the top middle represent the score, the asterisks (*) represent the tetras, when the tetras fill a line, the line that was filled will clear and the score

will go up depending on the number of lines cleared. Pressing A or D will move the tetras left or right. The board and the tetras is drawn directly on the memory address using ax as a register. Line clearing is done by replacing the tetras in the memory with blanks.

To draw the board game first I need to set the VRAM address to display the board where I want, and then storing the wall in a register, next display the board by printing them to the screen by looping the print function of the wall several times depending on the size of the board. Next creating the tetras, there are 7 tetra in total such as the L shaped tetra, the reverse L, the S shaped tetra, the reverse S, the T shaped tetra, the Box shaped tetra, and last the I shaped tetra, to create the tetras I need to set the shape as a variable as a x,y format, it's done like this example `piece_linex db 0,0,0,0` `piece_liney db 0,1,2,3` it will create the I shaped tetra. Next in order to move the tetra down in a fixed interval first the function needs to be looped, to make it move downward I altered the tetra's address each time it is looped. Next is implementing keyboard input to move the tetras, in order to do this, a function that detects input from keyboard is necessary this is done by adding an input detector in the function that move the tetra downward, if there is a button pressed the tetras will move depending on the input before continuing to move down in the loop, this is done by "in" function in assembly, continued by comparing if there's an input the function will jump to further detect which button is pressed and move according to the button pressed. Next step is to clear the line when tetras filled the line, this is done by replacing the asterisks (*) with blanks when the asterisks form a line. Next is the scoring function, to do this a function that tracks the number of lines cleared is needed, this is done by adding the function inside the clearline function loop, next new variable called score is needed to store the information about the score, this function works by using al as a register to show the score variable by displaying it on the screen then increment it for each loop done and then storing al in the score variable.