

Practical 9

Space Invaders (Part 2)

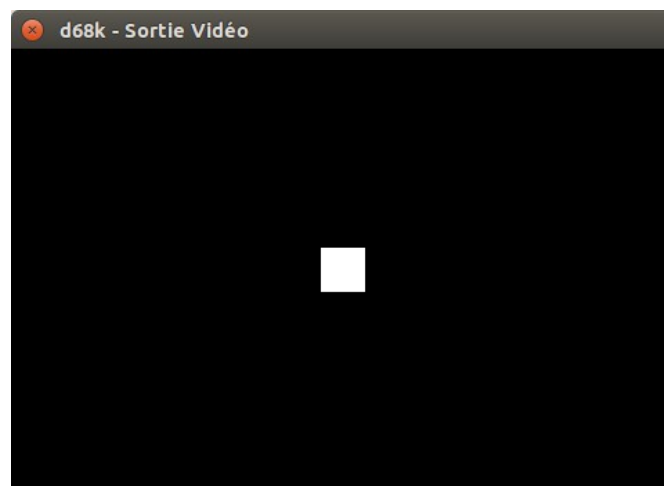
Step 1

Write the **WhiteSquare32** subroutine that draws a white square in the middle of the screen. The size of the square is 32 by 32 pixels.

Instructions:

The starting address of the square should not be computed dynamically (i.e. computed by the program running). You should compute it yourself and write its value directly in the source code.

Screenshot of the expected result:



Step 2

Write the **WhiteSquare128** subroutine that draws a white square in the middle of the screen. The size of the square is 128 by 128 pixels. Follow the same instructions given previously.

Step 3

Write the **WhiteSquare** subroutine that draws a white square in the middle of the screen. The size of the square will be passed as a parameter when calling the subroutine.

Input: **D0.W** = Size of the square in bytes (the size in pixels is then a multiple of 8).

- **Note:**

Since the height and width of the window are even, the square cannot be centered if **D0.W** is odd. For the time being, ignore this.

- **Instructions:**

In order to avoid nested loops and to limit the size of **WhiteSquare**, it is advisable to write the **WhiteLine** subroutine that draws a single horizontal white line. **WhiteLine** can then be called by **WhiteSquare** in a simple loop. The **WhiteLine** inputs are as follows:

Inputs: **A0.L** = Starting video address of the line.

D0.W = Size of the line in bytes.

Use the main program below in order to run and test **WhiteSquare**. Execute this program by pressing **[F10]** successively.

```
Main          move.w  #2,d0
\loop          jsr      WhiteSquare

               addq.w  #2,d0
               cmpi.w  #40,d0
               bls      \loop

               illegal
```