



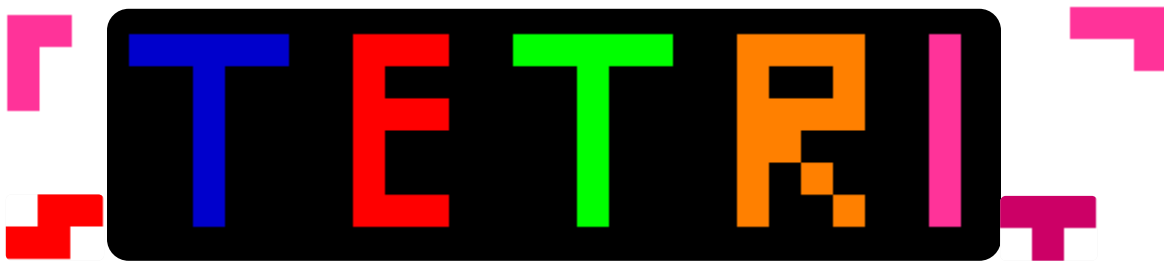
UNIVERSIDAD METROPOLITANA

Caracas, Venezuela 21 de marzo de 2016

Miguel Moretón

Proyecto N°1

Organización del Computador



TETRI

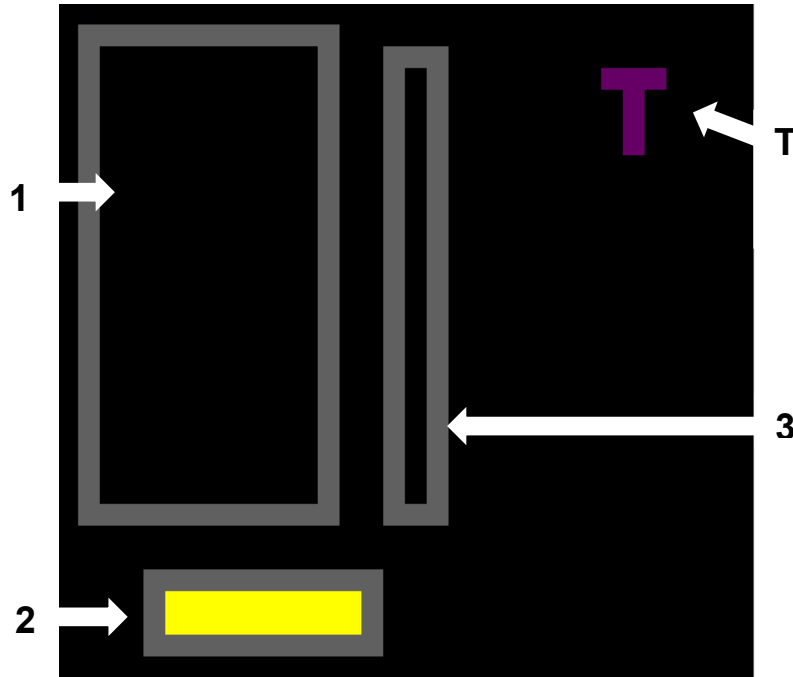
TETRI es un simulador del famoso juego TETRIS programado en MARS (MIPS Assembler and Runtime Simulator). Posee 7 tipos de piezas distintas, puntuación, vidas y niveles de dificultad variable. Todas las piezas se mueven en tres direcciones, rotan hacia la izquierda indefinidas veces y caen a velocidad “constante”. El objetivo eliminar cuantas líneas sea posible organizando las piezas para que formen una línea horizontal continua sin que se llegue al techo del tablero. El juego termina cuando el jugador se queda sin vidas.

Tablero

El tablero de juego consta de tres partes distintas y una letra **T** de color morado “titilante”:

1. La zona de juego
2. La barra de energía (o vidas)

3. La barra de nivel

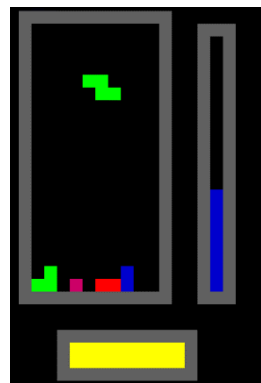


Descripción:

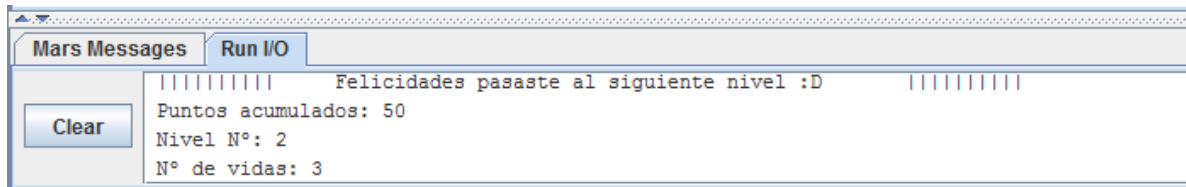
- La letra T indica que se tiene que presionar una tecla para iniciar el juego o la partida
- 1) La zona de juego es donde van a aparecer las piezas
- 2) La barra de vida muestra la cantidad de vidas disponibles (máximo 3 vidas).



- 3) La barra de nivel se llenará cada vez que se elimine una línea de piezas en la zona de juego. La velocidad con que se llene dependerá del nivel del juego. Una vez llena se pasa al siguiente nivel.



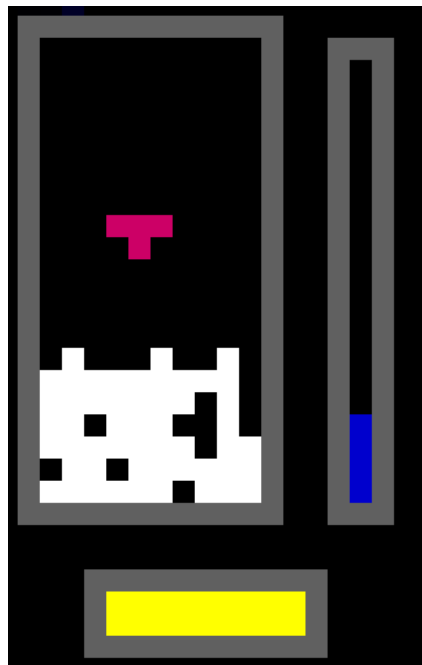
- Adicionalmente se tiene una barra de notificaciones que muestra instrucciones del sistema, vidas, puntaje por nivel, etc...



Nivel Fantasma



El juego cuenta con un nivel fantasma que luego de superar el nivel 3 se activa haciendo que todas las piezas cuando “choquen” se transformen en fantasmas, pero la mecánica sigue siendo la misma.



Configuración del juego:

- Abrir MARS V4.5 o superior
- Abrir el Archivo Proyecto
- Ensamblar
- Darle a la pestaña Tools y seleccionar “Bitmap Display” y luego en la misma pestaña seleccionar “Keyboard and Display MMIO Simulator”
 - Configurar el Bitmap dependiendo de las dimensiones de tu pantalla.

Ambas con la dirección base especificada

Para pantallas grandes

Unit Width in Pixels	16 ▼
Unit Height in Pixels	16 ▼
Display Width in Pixels	512 ▼
Display Height in Pixels	512 ▼
Base address for display	0x10008000 (\$gp) ▼


Para pantallas pequeñas

Unit Width in Pixels	8 ▼
Unit Height in Pixels	8 ▼
Display Width in Pixels	256 ▼
Display Height in Pixels	256 ▼
Base address for display	0x10008000 (\$gp) ▼

- Configurar el Keyboard and Display MMIO Simulator **cuidando que el Delay este en 1**

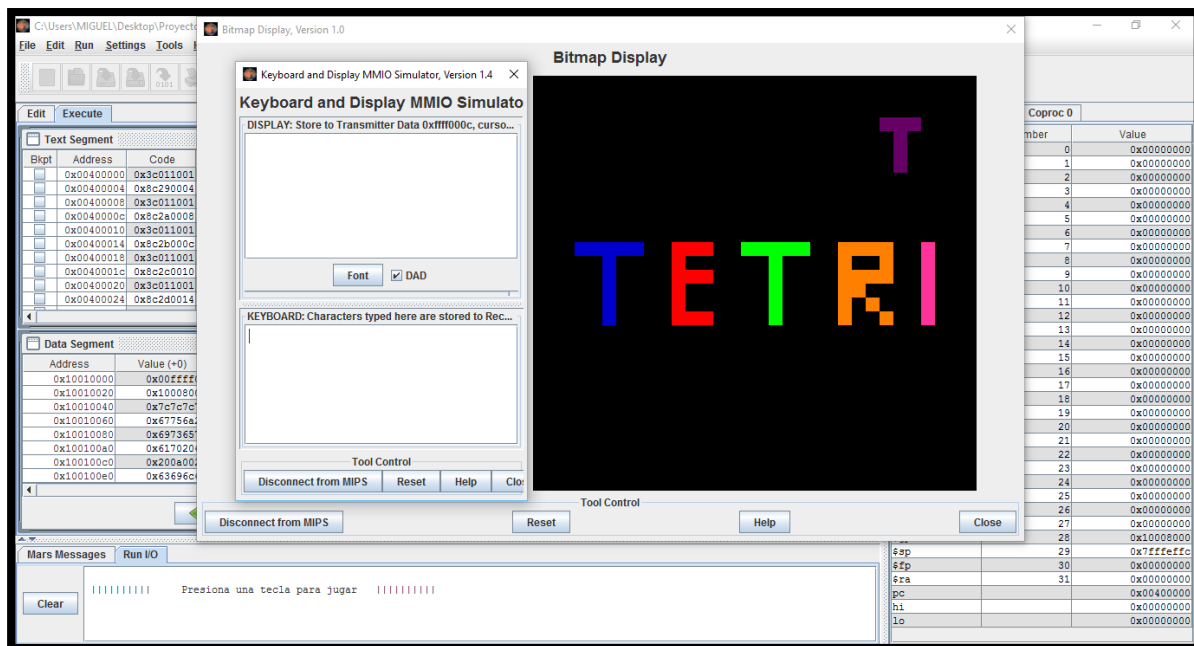


Es importante dejar tanto el Display como el Keyboard abierto antes de compilar.

- Conectar ambas herramientas a Mips (pestaña inferior izquierda de las ventanas)
- Darle a correr 
- Deberías tener las pestañas organizadas como aparece en la imagen de la siguiente página cuidando de que no se corte el tablero de juego o se oculte la barra de notificaciones “Run I/O”

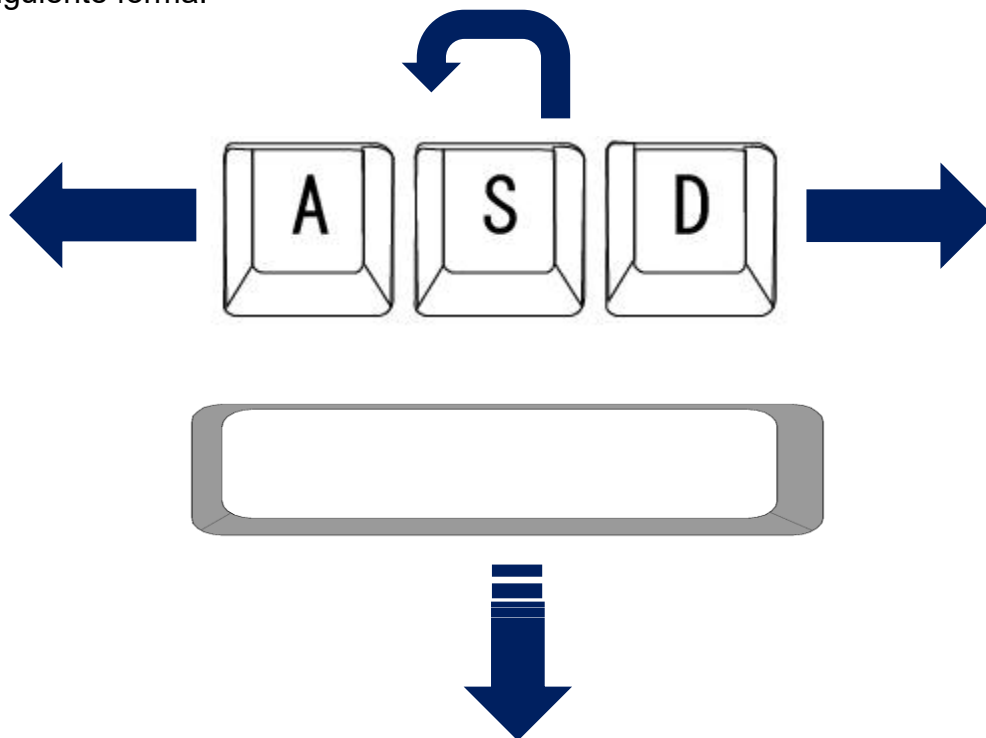
TETRI

- Por último Posicionarte en el Keyboard y seguir las instrucciones del juego



Instrucciones del juego:

Las piezas de mueven utilizando las teclas **A**, **S**, **D** y la **barra espaciadora** de la siguiente forma:



- A: mueve la pieza hacia la izquierda
- D: mueve la pieza hacia la derecha
- S: rota la pieza hacia la izquierda
- Space: hace que la pieza caiga rápidamente

Vidas

Cuando se comienza el juego originalmente se tiene tres vidas, cada vez que se pasa de nivel se agrega una más hasta un máximo de tres.

Niveles

El juego consta de infinitos* niveles y un nivel fantasma. Para pasar de nivel se tiene que llenar la barra de puntos eliminando filas horizontales continuas generadas por las piezas del juego.

Bugs *hasta la fecha*

A veces cuando se presiona la tecla de movimiento o rotación no se mueve la figura y eso tiene que ver por la manera como se estructuro el programa. Se pueden usar métodos alternativos de direccionamiento, pero el “Keyboard and Display MMIO Simulator” de MARS tiene varias limitaciones y fallos, por ende, la manera como fue implementado en este programa es las que asegura que haya menos errores.

Otro error también relaciona con el Keyboard es que si se presiona repetidas veces una tecla de dirección (A, S o D) el programa se para, nuevamente generado por limitaciones de MARS. Pero tranquilo el problema es menos evidente cuando se avanza de nivel.

