Instructions-

The settings to run the program are 16 x 16 pixels, and a screen width of 256 with a screen height of 512. The base address is $gp.

Once these settings are correct, all that is needed to run the program is to connect the bitmap keyboard, and use A and D to move left and right.

The initialization of $s3 done at line 301 is what controls the speed of the blocks descending, and it’s adjustment will lead to the blocks falling slower if it’s at a larger number, or faster if it’s at a higher number.

Overview-

This program creates a playable game of tetris on a bitmap, although the pieces cannot be turned. In more detail, the program creates a box in black to serve as the game screen, then chooses a random tetris piece to begin lowering from the top. The piece can be moved left or right using the A and D keys, although it will automatically stop once it reaches the edge of the box, not allowing the player to move it any further. The piece will also automatically move downwards periodically, until it detects that there is something beneath one of its pixels, which will cause it to check for a completed line. If there is a completed line (a full row that has every location taken by a piece), then the program will automatically clear the line, and move everything above said cleared line down. If there is not a completed line once a piece reaches the bottom, the program will return to the top of the playing field, where it checks that there is room for a new piece. If there is, the above process is repeated until there is not, which will cause the game to end.

Warnings-

While I was coding I realized that if I started changing the run speed of the program and pausing it, it would permanently slow the run of the program. I’m not sure if this is exclusive to my computer or something, but regardless, if this happens it can be fixed by simply closing mips and opening it again.

