**Project introduction**

**Catcher Game**

**Xi Cui**

1. How to Run
2. Assemble MIPS
3. Set the Tools -> Bitmap Display:

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)

Connect to MIPS

1. Set the Tools -> Keyboard and Display MMIO Simulator

Connect to MIPS

1. Make sure speaker is on, the game will play a simple music at the end. ^\_^
2. Keyboard Control: move catcher

"a": move left;

"s": move down;

"d": move right;

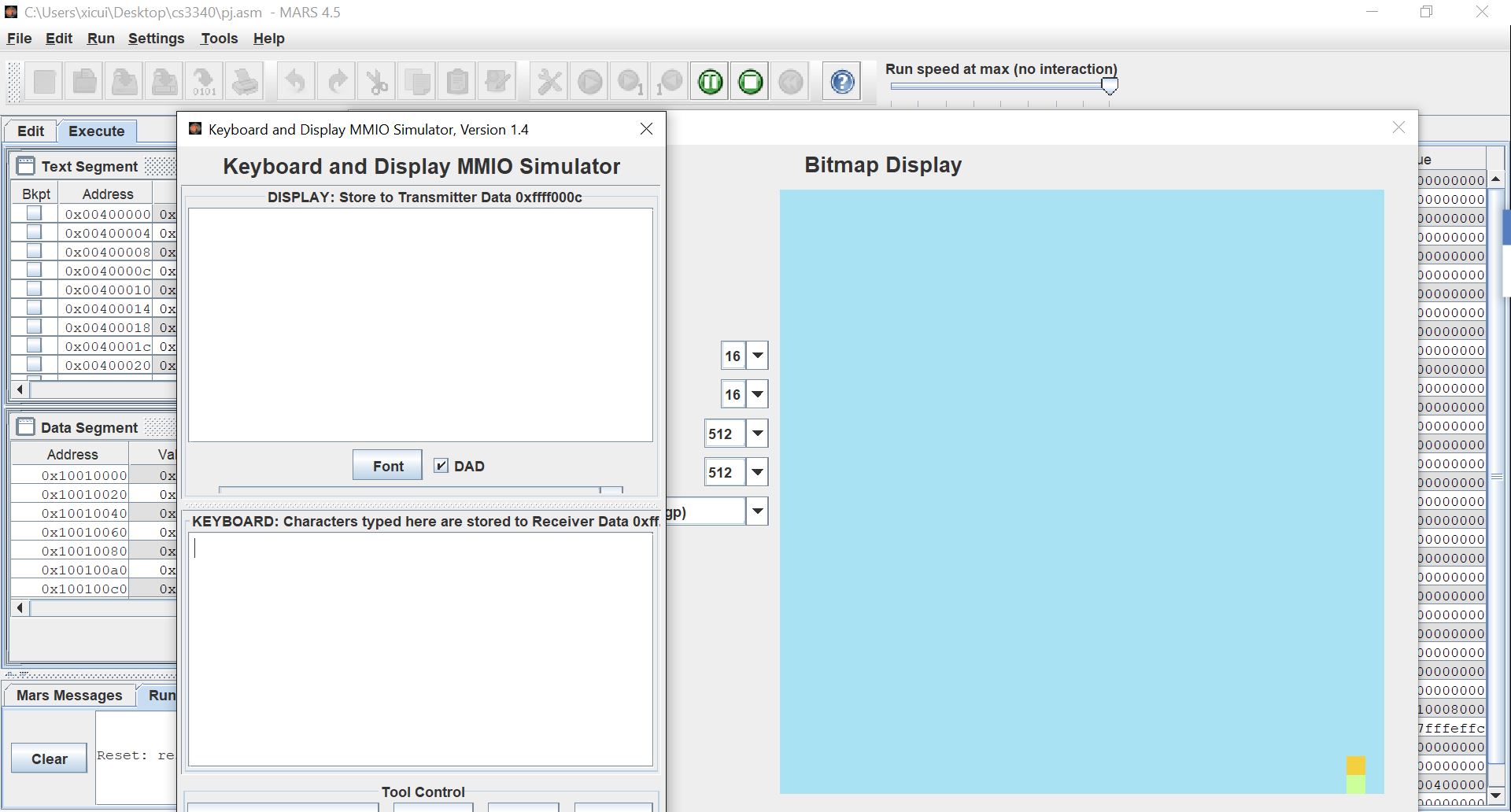
"w": move up;

"e": end the game;

“j": change colour upside-down

1. Run the program and click at Keyboard Display window to make sure cursor work. Then press one of control key above to start the game.

A Readied Screen Shot:



1. Rules

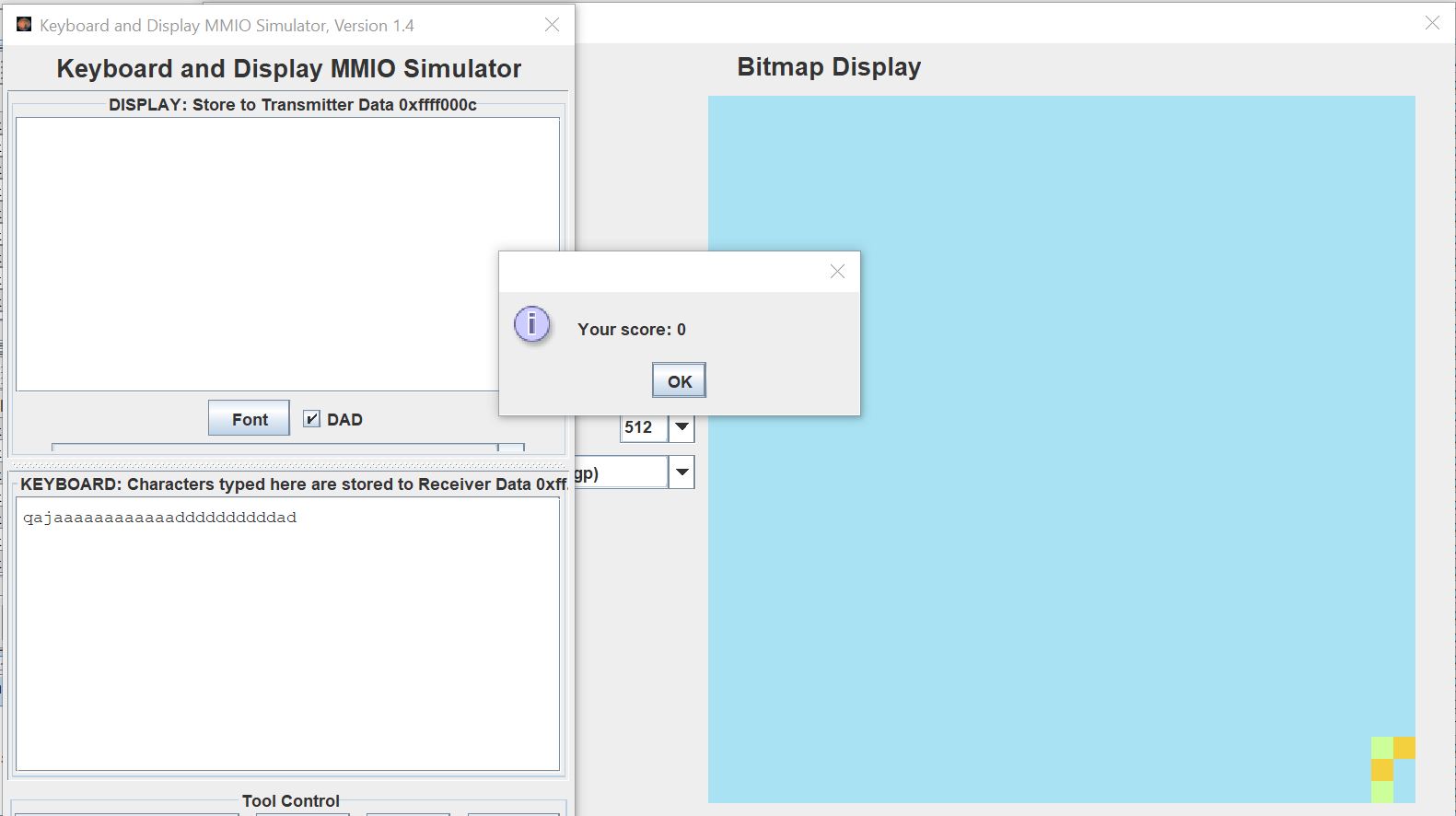
The Target will fall from random position of the screen, Catcher below need to catch the Target to gain the point.



Catcher is vertical: Target is horizontal: 

1. Target falls distance is 1 pixel, Catcher move 2 pixels in one movement.
2. Catcher should catch Target in correspondence color.
3. One success catching gain 1 score. Then a new target will be created and start falling.
4. Game Over:
5. Failed to catch the target. i.e. The target falls out.
6. Catch the Target with wrong colour side.
7. Press “e” to end the game.
8. Score will show after Game Over music

A Game Over Screen Shot:



b. Catch the Target with wrong colour side.

1. Idea

The idea comes from the “Snake” game of snake chasing and getting food. And I also played game to catch falling down apple. This game is very simple, but I hope you could enjoy.

1. Paint

Paint color on bitmap displayer is simply save color code to the $gp global pointer of the bitmap. But the difficulty is to calculate the correct position.

1. Move

There are two kinds of moving happened simultaneously, Target falling down and player controlling the movement of catcher.

The colored pixel’s movement on bitmap displayer is actually about changing color of its old and new position. Paint the old position as background color, then paint the new position with pixel’s self-color. The difficulty is to calculate the new position. To calculate the position, horizontal direction is to add or sub 1 word, but vertical direction need to multiply the screen width then do add or sub. To make sure two pixel move together, the color should change at same situation.

1. Branch

Target keeps falling down until drop out or catch by catcher. So, the keyboard control movement of catcher need to return to falling loop no matter it does or not. The use of branch “b”, give chance to branch two processes.

1. Use of Nop

This reduce the breakdown chance, and convenient to debug.

1. Make simple music

I really have fun at this part as I learnt to play piano at childhood. The system call 33 contains many kinds of instrument sound, and the pitch includes a standard 88 key piano begins at 9-A and ends at 108-C. Hope you have fun when music play.

I use the pitch code as follow:

