CAB202 Assignment 2 Documentation

Jarod Lam
n9625607

October 2018

Contents

1 Assignment implementation summary

Table 1: Assignment implementation summary

Item number	Item description	Implementation level
1	Intro	Fully implemented
2	Pause game	Fully implemented
3	Player size	Fully implemented
4	Block size	Fully implemented
5	Random blocks	Fully implemented
6	Player movement	Fully implemented
7	Treasure	Fully implemented
8	Basic game mechanics	Partially implemented
9	Block movement	Fully implemented
10	Player velocity	Partially implemented
11	Player jumping	Not implemented
12	Player inventory	Not implemented
13	Zombies	Not implemented
14	Pause screen advanced	Not implemented
15	ADC for block speed	Fully implemented
16	Switch debouncing	Fully implemented
17	LED warning	Not implemented
18	Direct control of LCD	Not implemented
19	Multiple timers	Partially implemented
20	Program (flash) memory	Partially implemented
21	PWM controlled visual effects	Partially implemented
22	Pixel level collision	Not implemented
23	Serial communication events	Partially implemented
24	Serial communication game control	Partially implemented

2 Basic functionality test plan

2.1 Intro

Table 2: Intro test plan

Test of specific functionality	**Test setup and actions**	**Expected result**	**Actual result**
Program displays student name and number initially.	Load the program and check that name and student number are displaying as expected.	Student name and student number are displayed.	As expected.
Pressing SW2 starts the game.	After loading game, press SW2.	Intro screen clears and game screen is drawn.	As expected.

2.2 Pause game

Table 3: Pause game test plan

Test of specific functionality	**Test setup and actions**	**Expected result**	**Actual result**
When joystick centre is pressed once, the game pauses.	Press the joystick centre while the game is running.	All sprites stop moving and are unable to move by using the normal controls, the game screen is cleared, and information is displayed.	As expected.
Game information is displayed on the pause screen.	Press the joystick centre to view the pause screen.	Lives remaining, current score, and game time in mm:ss format are displayed.	As expected.
The game is resumed when joystick centre is pressed again.	Press the joystick centre while on the pause screen.	Pause screen disappears and game reappears on screen.	As expected.

2.3 Player size

Table 4: Player size test plan

Test of specific functionality	**Test setup and actions**	**Expected result**	**Actual result**
The player initially appears on a 'starting block' in the top row.	Start the game.	The player begins on a stationary safe starting block in the top row.	As expected.
The player's sprite is at least 3 pixels high and 3 pixels wide.	Run the game and observe the player sprite.	The player's sprite is 8 pixels high and 9 pixels wide.	As expected.

2.4 Block size

Table 5: Block size test plan

Test of specific functionality	**Test setup and actions**	**Expected result**	**Actual result**
All blocks are at least 2 pixels high.	Start the game and observe the block sprites.	All blocks are 2 pixels high.	As expected.
All blocks are at least 10 pixels wide.	Start the game and observe the block sprites.	All blocks are 10 pixels wide.	As expected.
Blocks are clearly distinguished from each other.	Start the game and observe the block sprites.	All blocks have visible horizontal spacing between each other.	As expected.
All blocks are always at least `player sprite height + 2` pixels vertically separated from other blocks.	Start the game and observe the block sprites.	All blocks are 10 pixels vertically separated from other blocks.	As expected.
There are at least 7 safe blocks on the screen at one time.	Start the game and observe the block sprites.	There are at least 7 safe blocks on the screen	As expected.
There are at least 2 forbidden blocks on the screen at one time.	Start the game and observe the block sprites.	There are at least 2 safe blocks on the screen.	As expected.

2.5 Random blocks

Table 6: Random blocks test plan

Test of specific functionality	**Test setup and actions**	**Expected result**	**Actual result**
Blocks have no consistent observable pattern.	Start the game and observe the blocks.	Each block appears in a randomly selected row and column.	As expected.
Blocks do not overlap other blocks.	Start the game and observe the blocks.	All blocks stay within their respective rows and columns.	As expected.

2.6 Player movement

This functionality has been redefined under advanced functionality. See "*Player velocity*".

2.7 Treasure

Table 7: Treasure test plan

Test of specific functionality	**Test setup and actions**	**Expected result**	**Actual result**
The treasure sprite is no larger than the player's sprite.	Start the game and observe the treasure sprite.	The treasure sprite is 8 pixels high and 8 pixels wide.	As expected.
The treasure sprite does not overlap any of the blocks.	Start the game and observe the treasure sprite movement.	The treasure sprite never overlaps a block.	As expected.
The treasure sprite spawns in the bottom half of the screen.	Start the game and observe the treasure sprite.	The treasure sprite spawns above the bottom row of blocks.	As expected.
The treasure sprite moves back and forward, changing horizontal direction when it reaches the edges of the screen.	Start the game and observe the treasure sprite movement.	The treasure sprite moves back and forward horizontally and 'bounces' off the edges of the screen.	As expected.
The treasure sprite stops moving when SW3 is pressed and starts moving again if SW3 is pressed again.	Press SW3 while the treasure sprite is visible, then press it again.	The treasure sprite will stop moving, then start moving again.	As expected.
The treasure sprite disappears when the player collides with it and gives the player 2 more lives and returns the player to the 'starting block'.	Guide the player to the treasure sprite and collide with it. Press joystick centre to check lives.	The treasure sprite disappears, the player gains 2 more lives, and is returned to the 'starting block'.	As expected.

2.8 Basic game mechanics

Table 8: Basic game mechanics test plan

Test of specific functionality	**Test setup and actions**	**Expected result**	**Actual result**
The player starts with 10 lives.	Start the game and press joystick centre to view the player's lives on the pause screen.	The player has 10 lives.	As expected.
A point is scored every time the player lands on a safe block.	Move the player around onto multiple safe blocks, then press joystick centre to check the score.	The player's score goes up when landing on a safe block.	Not implemented.
The player dies if any part of the player sprite moves off the screen in any direction or manner.	Guide the player off the sides or bottom of the screen.	The player dies when it hits the edges of the screen.	As expected.
On death, the player respawns on the 'starting block'	Kill the player using any method imaginable while having 2 lives or more.	The player dies and respawns on the stationary 'starting block'.	As expected.
When the player loses all their lives, the game over screen is displayed which displays a game over message, total score, and game play time.	Kill the player repeatedly until all lives are gone.	The game over screen is displayed showing a message, total score, and game play time in mm:ss format.	Not implemented.
The game over screen allows the player to restart by pressing SW3 and score, lives, time, and player position all reset.	Press SW3 on the game over screen.	The game screen disappears and score, lives, time, and player position all reset.	Not implemented.
The game over screen allows the player to end the game by pressing SW2 which clears everything and just displays student number on the screen.	Press SW2 on the game over screen.	The screen is cleared and the student number is displayed on the screen.	Not implemented.

3 Advanced functionality test plan

3.1 Block movement

Table 9: Block movement test plan

Test of specific functionality	**Test setup and actions**	**Expected result**	**Actual result**
All blocks move at a constant horizontal speed.	Start the game and observe block movement.	Blocks move at the same constant speed and do not accelerate by themselves, except for the starting block.	As expected.
Each row of blocks must move in the opposite direction to the row above it.	Start the game and observe block movement.	Adjacent rows move in opposite directions, except for the starting block.	As expected.

3.2 Player velocity

Table 10: Player velocity test plan

Test of specific functionality	Test setup and actions	Expected result	Actual result
Pressing the joystick left or right while the player is supported by a block sets the player in continuous horizontal motion at a constant speed relative to the block in the appropriate direction.	Move the player onto a moving safe block and press joystick left and right.	The player will move left when the left and right when the left and right joysticks are pressed, respectively, at a speed relative to the block.	As expected.
When in horizontal motion, the player's horizontal velocity must be greater than that of the block.	Move the player onto a moving safe block and press the joystick against the direction of motion.	The player makes progress against the direction of the block.	As expected.
If the player is moving horizontally on a supporting block and the joystick is pressed in the opposite direction, the player stops moving relative to the supporting block.	Move the player onto a moving safe block, then press joystick left and then right.	The player begins moving left relative to the block, then stops moving relative to the block and is carried by its motion alone.	As expected.
If the player is moving horizontally on a supporting block, and the joystick is pressed in the same direction as current movement, then the player continues to move at the same speed in the current direction.	When the player is stationary, press joystick left or right, then press the joystick the same direction again.	The player moves in the direction of the joystick and does not speed up or slow down.	As expected.
If the player is moving horizontally on a supporting block and the joystick is not pressed, then the player continues to move at the same speed in the current direction.	When the player is stationary, press and release either joystick left or right.	The player continues moving in the direction the joystick is pressed after it is released.	As expected.
If the player is not supported by a block, then it will commence to accelerate downwards.	Guide the player off a safe block until it is not supported by any block.	The player will start to accelerate downwards.	As expected.
If the player is moving horizontally before leaving the support of a block, then the player must continue to move horizontally at the same speed while accelerating downwards, so that a parabolic flight path will be observed.	Move the player off a moving safe block.	The player will keep its horizontal velocity but accelerate downwards according to gravity.	As expected.
If the player is moving without support of a block and the player lands on a safe block, it will then immediately begin to move horizontally in the same speed and direction as the block, and all vertical motion will cease.	Move the player off a moving safe block so that it falls and lands on another moving safe block.	When the player lands, it will immediately begin to move horizontally in the same speed and direction as the block, and all vertical motion will cease.	As expected.
If the player is moving without support of a block, and its	Make the player		

3.3 Player jumping

Table 11: Player jumping test plan

Test of specific functionality	**Test setup and actions**	**Expected result**	**Actual result**
Pressing the joystick up while the player is supported by a block causes the player to jump.	Press the joystick up while the player is supported by a block.	The player is given an upward velocity.	Not implemented.
After UP is pressed, the player should commence to move upwards. Any horizontal motion should continue, and the acceleration provision when the player is not supported by a block will take effect.	Press joystick up while on a safe block with horizontal motion.	The player will retain the previous horizontal motion of the block while being given a new upwards velocity.	Not implemented.

3.4 Player inventory

Table 12: Player inventory test plan

Test of specific functionality	**Test setup and actions**	**Expected result**	**Actual result**

3.5 Zombies

Table 13: Zombies test plan

Test of specific functionality	**Test setup and actions**	**Expected result**	**Actual result**

3.6 Pause screen advanced

Table 14: Pause screen advanced test plan

Test of specific functionality	**Test setup and actions**	**Expected result**	**Actual result**

4 Specialised Teensy functionality test plan

4.1 ADC for block speed

Table 15: ADC for block speed test plan

Test of specific functionality	Test setup and actions	Expected result	Actual result

4.2 LED warning

Table 16: LED warning test plan

Test of specific functionality	Test setup and actions	Expected result	Actual result

4.3 Multiple timers

Table 17: Multiple timers test plan

Test of specific functionality	Test setup and actions	Expected result	Actual result

4.4 PWM controlled visual effects

Table 18: PWM controlled visual effects test plan

Test of specific functionality	Test setup and actions	Expected result	Actual result

4.5 Pixel level collision

Table 19: Pixel level collision test plan

Test of specific functionality	Test setup and actions	Expected result	Actual result

4.6 Serial communication events

Table 20: Serial communication events test plan

Test of specific functionality	Test setup and actions	Expected result	Actual result

4.7 Serial communication game control

Table 21: Serial communication game control test plan

Test of specific functionality	Test setup and actions	Expected result	Actual result

5 Specialised Teensy functionality justification

5.1 Switch debouncing

5.2 Direct control of LCD write

5.3 Timers

5.4 Program memory