Digital Design and Computer Architecture LU

Lab Protocol

Exercise II

Vorname Nachname, Matr. Nr. 0123456
e0123456@student.tuwien.ac.at

Vienna, May 13, 2022

Task 1: VGA Graphics Controller

Subtask 1

VGA Oscilloscope Measurements

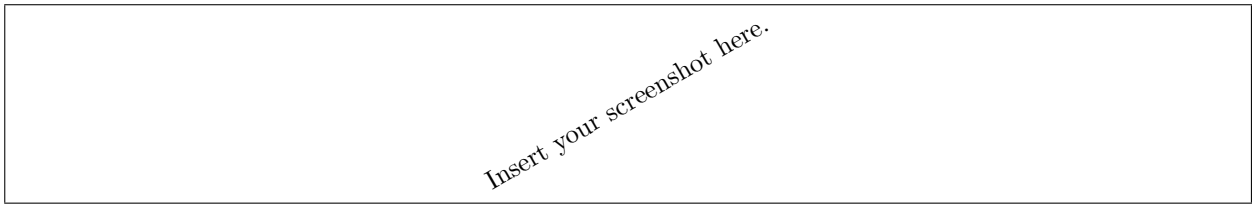

Figure 1: Line measurement with cursors marking the length (duration) of the whole line

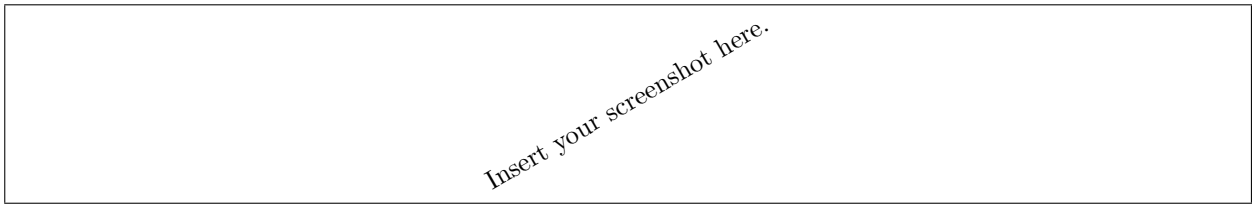

Figure 2: Line measurement with cursors marking the length (duration) of the horizontal synchronization pulse

END Subtask 1

Task 2: Tetris Game

Subtask 2

Briefly describe the architecture of your tetris_game module. Are there any submodules? What is their purpose? How many FSMs did you use?

add your explanation here (approximately 8-10 sentences, you can also include figures)

END Subtask 2

2.1 submodules for drawing

The submodules used for the drawing of blocks and strings on the screen are located in tetrisdrawerspkg.vhd The smallest unit of the implementation is a TBBBLOCK. It defines which bitblit to use when drawing it, and whether it is empty or not. A TBBROW is made of multiple TBBBLOCKS. A TBBMATRIX is made of multiple TBBROWS. The blockdrawer is used to draw a specific bitblit to a specified position on the screen. It is not used directly in the tetrisarchitecture. The blocklinedrawer is used to draw a line of blocks across the screen, provided a startingpoint, a tbbblock to use, a length, and a direction(horizontal or vertical). Inside the architecture, it used to draw the borders of the map. The blockmatrixdrawer is used to draw a tbbblockmatrix, starting at the specified x and y coordinates. The stringdrawer is used to print the labels for the score, removed-lines and next-tetromino.

2.2 submodules for game logic

The submodules used for the game logic are located in the tetrisgamelogicpkg.vhd The addtetrominohandler takes a blockmatrix, a tetromino, the coordinates of the tetromino inside the matrix, and the rotation of the tetromino as input. It adds all solid blocks of the tetromino to the matrix, and returns the new matrix. The rowsfullhandler is used for checking for full rows, removing them, and shifting the remaining rows accordingly. It also provides information about the number of rows that were removed. The timer is a simple counter that counts the number of clockcycles since it was last started, and sets its ouput to '1' after the specified amount of time has passed. It is both used for the automatic downwards movement of the tetromino, aswell as for measuring the time between different pitches for the synthesizers.

2.3 submodules for music

The methods and constants used for the music are located in the tetrisaudiopkg.vhd Here, a basic 2-channel arrangement of the tetris-theme is provided, aswell as some functions matching the different pitches to 8-bit vectors for the high- and lowtime of the synthesizers. Shouldn't you have tried implementing Task 1 before wasting time with adding music to the game? Yes, yes I should have.

2.4 submodules architecture

As for the tetrisarchitecture itself, its heart is the state.matrix object. It stores information about which blocks are solid on the map, aswell as which type of bitblit should be used when drawing them. The tetromino-collider acceses the matrix and the current tetromino. Couldn't you just have used the on-chip-memory to store the matrix? Yes, yes I could have. In hinsight, this would have been the better choice, as all submodules which have a matrix as output and input require a huge amount of data-connections and flip-flops, wheras storing the matrix inside the on-chip memory would have been the much more efficient option. When the collider registers a collision, the addtetrominohandler is started, which adds the current tetromino to the blockmatrix. After that, the rowsfullhandler is started, which checks for full rows and removes them accordingly.

Task 3: Bonus: SignalTap Measurement

Subtask 3

Trigger Condition

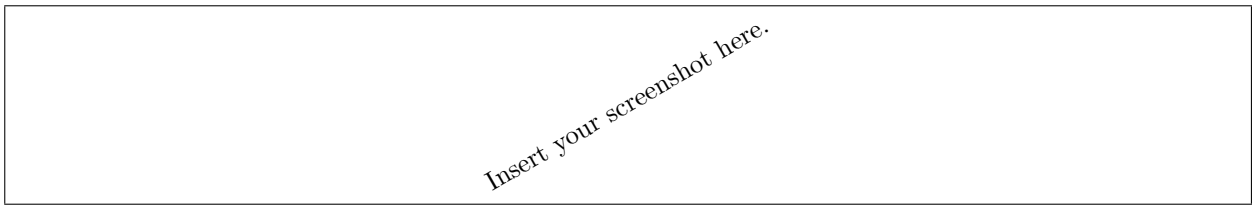

Figure 3: Screenshot showing the trigger condition

END Subtask 3

Subtask 4

Measurement Screenshot

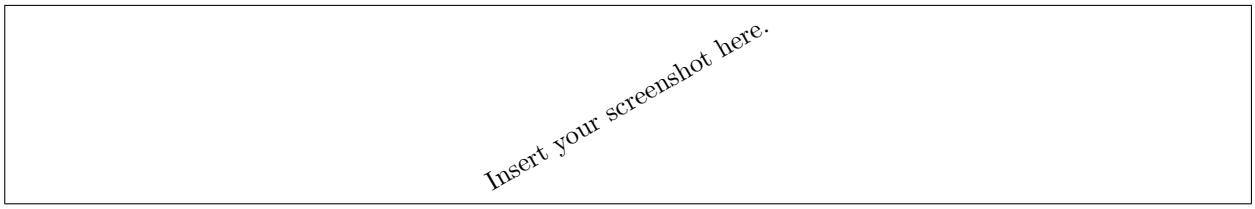

Figure 4: Screenshot showing at least the first 4 instructions (and their associated data items) issued to the graphics controller during one frame by the tetris_game module.

END Subtask 4

Subtask 5

Instruction Decoding

Command	Operands	Instruction Name	Description
0x..	• 0x0001	??	...
0x..	• 0x0001 • 0x0002	??	...
0x..	• 0x0001 • 0x0002 • 0x0003	??	...
0x..	• 0x0001 • 0x0002 • 0x0003 • 0x0004	??	...

END Subtask 5