

Group Number: 33

Moshiur Howlader / 001316948 / howlam@mcmaster.ca

Ryan Ganeshan / 001322407 / ganeshr3@mcmaster.ca

January 17th, 2018

Exercise 1: This exercise was completed by modifying in-lab experiment 1b to 8 separate areas, while recognizing that the LCD and touch panels having different resolutions. Each region was assigned a 3 bit colour counter, allowing for 8 different colours. Every time a touch event was detected, this counter was incremented and thus, the colour would change. The hold and drag functionalities were also implemented using a 3 second clock that only ran if we were holding the touch panel. This clock was reset if we raised our finger, if we travelled across regions, or if we reached 3 seconds. If we reached 3 seconds in one region, we would increment the colour code of that respective region. We kept track of the current and previous touched regions by implementing a touch region buffer. If the touch regions differed between clock cycles, region has changed, and the timer would be reset. Flickering issues were fixed by implementing the v_sync blanking code as described in exercise 2 write up below. The BCD counter was implemented similarly to the 3 second clock. The counters would count 0-9 and then increment the BCD to the left, while rolling itself over. The counter would reset if the region was left or if touch was removed. The max value and 2nd max value was then found using two priority encoders, which were implemented in ascending RGB colour code. The RGB code was appended to the appearing frequency, making for a 7-bit signal for each colour. The same approach was used for the 2nd max, but we did not test the already found max value to avoid it. These values were then sent to the BCD's. If all regions were the same colour, we turned off the 2nd max BCD's.

Exercise 2: This exercise was completed by modifying in-lab experiment 3. Cases for switches='d6 and 'd7 were implemented. The five-tap filter was implemented for when the switch configuration equals 'd6. Two registers used for implementing the 5 tap filter were enabled to add extra delays to handle the extra terms for this equation (Y_{p2} and Y_{m2}). Case 7 was implemented by adding 1-second counting circuitry. A case statement was added within the case 'd7 case. Within this case were 6 cases. We started the 1 second counter when the switch case equals 'd7. After this, the cases were iterated through for 1 second each. A flag was added when the iteration was in case 6. During this case, the register arrangement for switch configuration 6 must be implemented. Thus, this flag was OR'd with the "switch-configuration='d6" conditions. We also tried to avoid the v_sync issues during case 7 by only updating the "inner case" iterations during the blanking period. Two approaches were taken, the submitted exercise 2 folder's approach used a counter as outlined above. An alternate approach explored was to use the VSync's falling edge as a frame reference. A count of 60 frames meant 1 second, which can be used to count and iterate through the images. Despite trying both attempts, the black screen showed up for Filter_config image 6, although the transitions were smooth between the images (no flickering).