2022 Digital IC Design

Homework 4: Edge-Based Line Average interpolation

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| NAME | | 何立洋 | | | | | | |
| Student ID | | Q36101066 | | | | | | |
| **Simulation Result** | | | | | | | | |
| Functional simulation | Pass | | Gate-level simulation | Pass | Clock  width | 57442(ns) | Gate-level simulation time | 571902(ns) |
|  | | | | |  | | | |
| **Synthesis Result** | | | | | | | | |
| Total logic elements | | | | | 2308/68416(3%) | | | |
| Total memory bit | | | | | 0/1152000(0%) | | | |
| Embedded multiplier 9-bit element | | | | | 0/300(0%) | | | |
|  | | | | | | | | |
| **Description of your design** | | | | | | | | |
| 這次作業我的方法是將in\_data依序在電路處理後，再存進result image memory，因此，沒有使用到addr。  **程式描述:**  基本上還是使用上兩次作業的骨幹，循序電路設四個state。分別對應:  State0:reset時，不做存和算的動作，但預先準備計數值。  State1:讀取in\_data。我將其分為上行和下行(第一筆對應第一行和第三行)，  State2:計算中行。除了計數值，其他部分寫在組合電路，包括資料比較後D1,D2,D3給值。  State3:存入data\_wr和準備新的行。將上行和中行存入data\_wr後，將下行存入上行，之後回到State1，此時，讀取in\_data僅存入下行。  為了方便，我設置需多的計數，作業遇到比較大的問題是在State2比較數值時，經常會延一個clk。  **感想:**  這次作業沒有上次那麼複雜，有可能是因為我的做法是屬於比較簡單的那種，也因此我的performance比較普通，但這是目前的我想到的最直接、最直觀的方法。這段時間我對verilog漸漸熟悉了，但我在寫程式時，還是比較容易從撰寫軟體的方式去思考。  **改進:**  我發現我用datamem\_middle[caculate-1]，似乎會花費相當多時間，我在撰寫時都沒有注意到這個問題。之後做類似的儲存動作時，我應該會改用shfit。 | | | | | | | | |

*Scoring = (Total logic elements + total memory bit + 9\*embedded multiplier 9-bit element) (longest gate-level simulation time in ns)*