2017 Digital IC Design

Homework 6: Frequency Analysis System

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NAME | 沈育同 | | | |
| Student ID | P76061386 | | | |
| **Simulation Result** | | | | |
| **Testfixture 1** | | | | |
| Functional RANK | | A | | |
| Gate-level RANK | | A | | |
| Gate-level simulation time | | 16323.090 ns | | |
| **Testfixture 2** | | | | |
| Functional RANK | | A | | |
| Gate-level RANK | | A | | |
| Gate-level simulation time | | 16323.090 ns | | |
|  | | | |  |
|  | | | |  |
| **Synthesis Result** | | | | |
| Total logic elements | | | 3643 | |
| Total memory bit | | | 0 | |
| Embedded multiplier 9-bit element | | | 12 | |
|  | | | | |
| **Description of your design** | | | | |
| 本電路分為FIR、FFT、Analysis等3塊子模組作介紹：  **FIR:**    Fig 1 FIR架構圖  由於FIR的參數具有對稱性h(0) = h(31), h(1) = h(30)…，故採用Fig 1架構將原本須使用32個乘法器減半成16個乘法器，其運作原理為第1筆輸入資料data[0]與h[0]相乘後放入reg 0，下一個週期第2筆資料data[1]與h[1]相乘後並與reg 0進行累加放入reg1，一直到第17個週期data[16]與h[15]相乘後並與reg15進行累加放入reg16，最後第32個週期data[31]與h[0]相乘後並與reg30進行累加放入reg31，此時reg31即為第1筆fir\_d，接著之後的fir\_d會順應此pipeline架構不斷產出。  **FFT:**    Fig 2 FFT架構圖    Fig 3 PE架構圖  由於原本的FFT有4階段的交叉運算，故本電路賦予每1階段皆配有1個PE運算單元來處理單一個交叉運算，Fig 2中的5條Register File每個週期皆往上位移，每個PE以Pipeline形式，在適當的位置取出資料，並計算出下個階段的資料放置於適當的位置，此外由於w0與w4為1及-j，可以使用加法器輕易的算出Fig 3的fft\_b，因此PE3與PE4以更簡單的方式來取得fft\_b便可省下將近一半的乘法器。  **Analysis:**    Fig 4 Analysis架構圖  如Fig 4，當接收到fft\_valid = 1時，會將fft\_d0至fft\_d15存入Reg 0至Reg 15，max設為0，Counter設為0，接著每次週期Reg 0至Reg 15會往上位移，透過計算電路算出a\*a+b\*b並與max比較大小，如max較小則a\*a+b\*b存入max，同時Counter存入freq。 | | | | |

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

= (3643 + 0 + 9 \* 12) \* 16323.090 = 61227910.59