

R11922137 賈皓中

用新的 data/inst. memory 做 pipeline

1.

EX: forwarding + ALU= 906ps <= critical path

```
ABC: Start-point = pi365 (\rd_mem_r [1]). End-point = po433 (\alu_out_w [11]).  
ABC: + write_blif <abc-temp-dir>/output.blif
```

```
ABC: WireLoad = "none" Gates = 2498 ( 26.1 %) Cap = 2.8 ff ( 1.1 %) Area = 2819.07 ( 92.1 %) Delay = 906.94 ps ( 4.7 %)
```

修改 ADDR_W & DATA_W

```
parameter ADDR_W = 12,  
parameter INST_W = 32,  
parameter DATA_W = 11
```

2. 從 CP=PC+4 之後->確認現在讀取的指令有沒有需要 stall/branch->然後將正確的指令位置寫入 PC

3. 如果 data hazard 包含 load use，用 forwarding unit 將以得到計算結果的 rd value 傳到 EX stage 的 rs1/rs2。如果是 load use hazard，stall 一個 cycle，然後將如 data mem 讀取到的參數用 forwarding unit 傳進去。

4. stall，如果讀到 branch 的指令，無論如何先 stall 1 cycle，再讀接下來的指令。

5. test case 8: branch

test case 9: load-use (data hazard) + branch

test case 10: branch

work load 2 應該可以透過 branch predictor 增加最多效能。

6. 增加 pipeline depth 可以將本來在同個 stage 裡有前後順序的 module，拆成不同 stage 執行，可以降低 latency。但 data/control hazard 會變得更難處理，所以增加 pipeline stage not always beneficial。