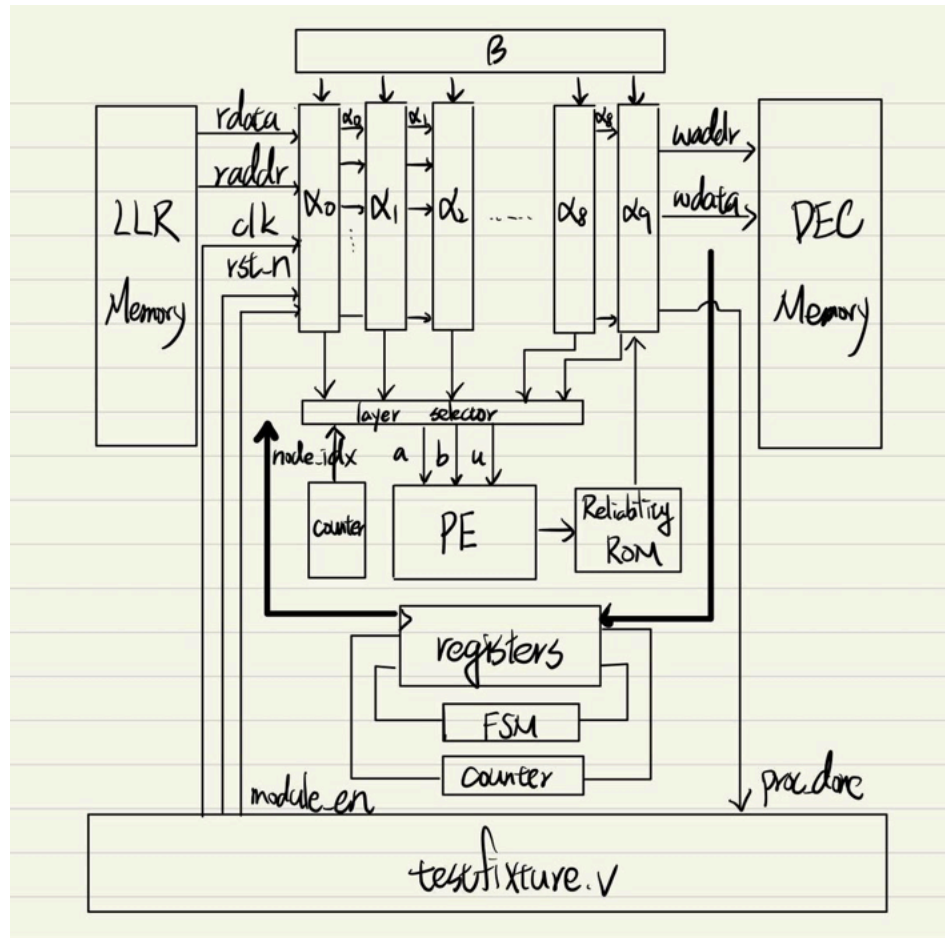


Team 28 Report

R11943119 童子瑜

R11943103 李彥儒

1. Block Diagram



- $\alpha_0 \sim \alpha_9$
用來表示各個 stage 算出來的值，由於每個 stage 算完後就可以直接更新，因此分別只有存 $2^{(10-\text{stage})}$ 個 bits。
- β
我們將 G function 所需要的參數 u 記做 β ，而因為 u 是需要從後往前更新的，因此每次算到最後一個 stage 我們就會更新 β 一次。
- Layer selector
因為 polar decoder 具有順序性（須從較低 stage 往較高 stage 去計算），因此需要 selector 決定現在要更新在哪個 stage。
- PE

主要做 F G function 的運算，輸入 a b u 並判斷要做的是 F 還是 G 後，output 出結果送回給 alpha。

e. Reliability_ROM

將三種長度的 Reliability 放進 ROM 裏面，輸入長度以及 index 就會輸出該位置的 reliability 決定是不是 frozen bit。

f. Node_idx counter

由於前面會有許多 frozen bit，很多的 u 都會是 0，因此我們會有一個 counter 用來記錄該 bit 到底需不需要被計算，減少 cycle 的數量。

2. Optimization Technique

a. 減少 alpha 的數量

原本我們設計 design 會在讀進所有 data 後才開始運算，但後來發現其實可以邊讀就可以變放進第一個 stage 了，因此可以省掉第一層 512*11 bits 的 reg。

b. 減少 beta 的數量

在計算 beta 時，需要用到其他位置的結果，所以原本我們會將全部得 beta 都記下來全部更新一次，後來經過推導可發現每次新的 beta 會是上次計算出的 beta XOR 下一個 stage 的結果，且每次會用到的 beta 其實只有固定數量，因此我們將 beta8~beta0 分別只存入 256 bit ~1 bit，且更新只會在下一次要計算的 stage，可減少一半的 reg。

c. 利用 node_idx 減少 cycle

從 reliability 裡可看出前面的 bits 其實很多都是 frozen bits (u 為 0) 因此我們透過計算 weight 的方式，決定出該 node 到底需不需要計算，如果不用就會跳過他，經實驗整體大概看以少掉 1/2~1/3 的 cycle