

# CAD/VLSI Circuit Design VLSI 期末構想報告書補充

## Fixed point QR decomposition using CORDIC

### Algorithms on FPGA with systolic array

7111064109 林軒宇 指導教授 范志鵬

#### 一、 簡介

QR 分解是數值線性代數中具備多種用途的計算工具，主要應用於線性方程、最小平方法和特徵值問題。常見的 QR 分解的計算方法包括 Householder 變換、Givens rotation 以及 Gram-Schmidt 正交法。本文使用 given rotation 搭配 CORDIC Algorithms。

本次實作採用  $8 \times 4$  矩陣，每個數字大小定義在  $\pm 0.25 \sim \pm 1$ ，預期得到一組  $8 \times 4$  的上三角矩陣 R。實驗流程為先使用 MATLAB 估算預期使用 **定點數(fixed point)** 的長度(浮點數與定點數的誤差需足夠小)以及 iteration 的次數，再將 MATLAB 生成的隨機  $8 \times 4$  矩陣以定點數格式匯入 verilog，並將 verilog 算出答案與 matlab 算出答案做比較，最後使用 FPGA 做驗證。

#### 二、 理論

##### ➤ given rotation 說明

$$x' = x \cos \phi - y \sin \phi$$

$$y' = y \cos \phi + x \sin \phi$$

找到角度  $\phi$  使得  $y' = 0$ ，以圖一為例， $a_{1,1}$  為  $x'$ 、 $a_{2,1}$  為  $y'$ ，找出角度  $\phi$  後，需將右側同一列的數值皆經過同樣的旋轉矩陣運算(原理同基本矩陣第二定理)，之後依序將  $a_{3,1}$ 、 $a_{4,1}$ ...變成 0，直到第一行除了  $a_{1,1}$  外均變成 0。同理第二行，將  $a_{2,2}$  為  $x'$ 、 $a_{3,2}$  為  $y'$  進行相同動作，直到第二行除了  $a_{2,2}$  外皆變 0。持續到第 n 行(最後一行)，即可得到上三角矩陣 R。

$$\begin{bmatrix} 1 & a_{1,1} & a_{1,2} & a_{1,3} & \dots \\ 2 & a_{2,1} & a_{2,2} & a_{2,3} & \dots \\ 3 & a_{3,1} & a_{3,2} & a_{3,3} & \dots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

(圖一)given rotation 範例圖

➤ 使用 CORDIC 達成 given rotation

$$x' = \cos \phi \cdot [x - y \tan \phi]$$

$$y' = \cos \phi \cdot [y + x \tan \phi]$$

先將  $\cos \phi$  提出，接著限制  $\tan(\phi) = \pm 2^{-i}$ ，即可將算式簡化如下：

$$x_{i+1} = K_i [x_i - y_i \cdot d_i \cdot 2^{-i}] \quad K_i = \cos(\tan^{-1} 2^{-i}) = 1/\sqrt{1+2^{-2i}}$$

$$y_{i+1} = K_i [y_i + x_i \cdot d_i \cdot 2^{-i}] \quad d_i = \pm 1$$

為了簡化運算量，將每次的運量係數省略，最後再乘上所有系數的乘積和 ( $A_N$ )

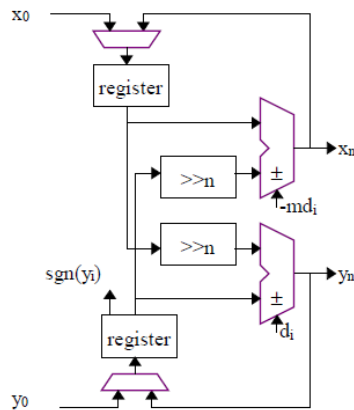
$$A_n = \prod_n \sqrt{1+2^{-2i}}$$

n 為疊代的次數

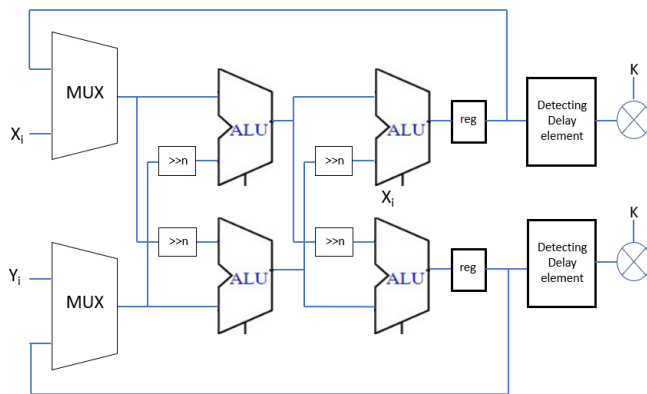
### 三、 架構改良

➤ 改良一： PE Element 改良

(圖二)為論文中一顆 PE 元件的架構，(圖三)為改良後一顆 PE 元件的架構，最大的改進在於完成一次計算所需花的 CLK 數變原先的一半(unfolding factor(J)=2)。缺點為會增加 Critical path，但此架構的 Critical path 為乘法器，因此可以近乎無代價的提高 Performance



(圖二)



(圖三)

➤ 改良二：input size 改良

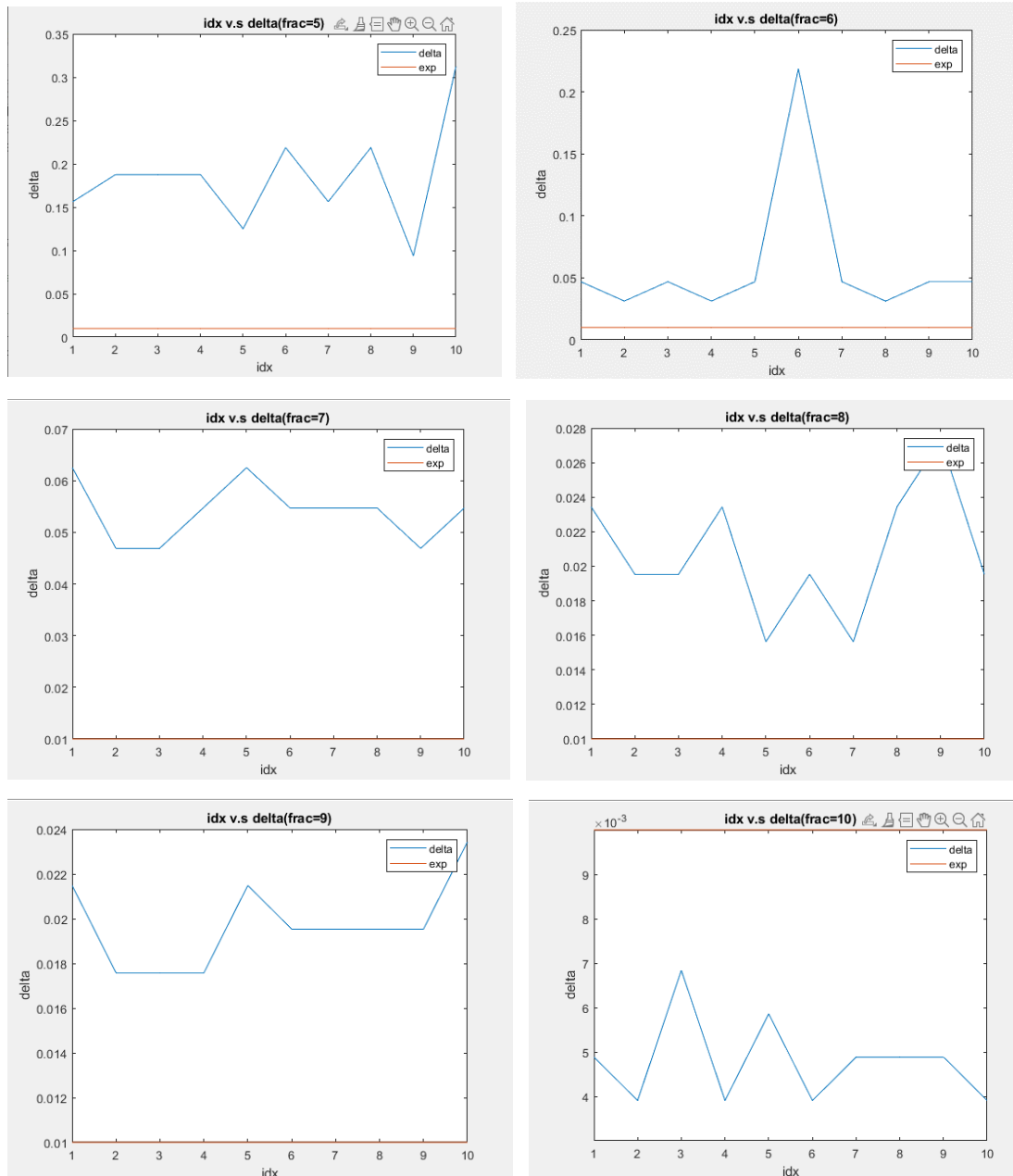
原論文疊代次數 n 為 9，為了使用上文圖三架構，我們須選擇偶數次疊代次數，因此此文選用疊代次數(n)為 8。此外原論文每個 input 為 16bits，此文將每個 input 改為 13bits(1 sign bit, 2 decimal bits, 10 fraction bits)，找法如下文。

首先定義 delta 函式如下：

$$\delta = \frac{\sqrt{\sum (r_{ij} - \hat{r}_{ij})^2}}{\sqrt{\sum r_{ij}^2}}$$

接著使用 matlab 畫圖，做法如下：

先將 Fraction bit 設定為 5 bit，皆著連續測試 10 組  $\delta$  值，如果不滿足  $\delta < 0.01$ ，就將 Fraction bit 加 1，結果如下圖：



橫軸為  $\delta$  值，縱軸為 index(從 1~10)，由上圖可以看出唯有 Fraction bit=10 時， $\delta$  值會小於 0.01，最終測出 Fraction bit 最小需要 10 bit。

➤ 改良三：systolic array 改良

(圖四)為原論文架構，(圖五)為改良版架構，改良版為 pipeline 版本，在高頻下也可以成功運作。由於 R22 與 R12 間有資料相依，中間需要加 delay，delay 數由疊代次數(n)與 J(unfolding factor)有關，算式如下：

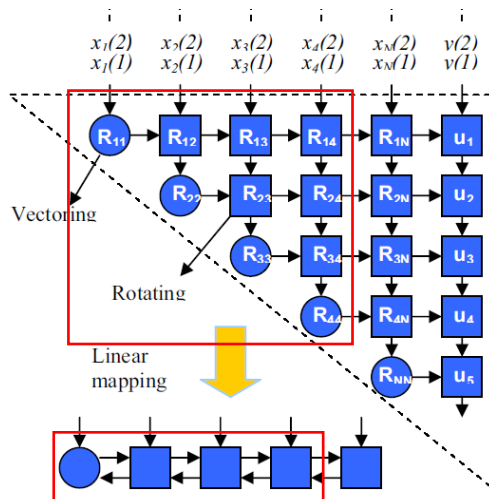
$$\text{Delay} \geq (n/J+1)*2$$

$n/J$  為  $X_{ij}$  執行 **Rotation mode** 的次數，加 1 為**乘法器**，乘 2 為有兩級資料相依。

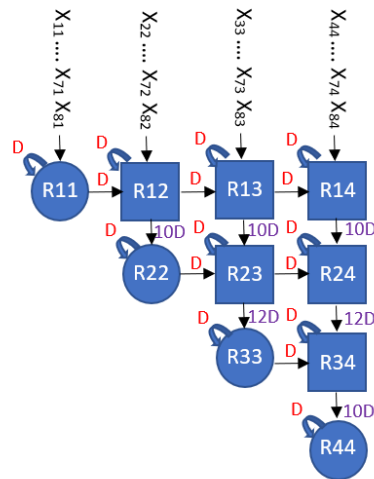
舉例：以  $X_{72}$ 、 $X_{82}$  為例，R22 執行需等到 R12 執行完  $X_{72}$ 、 $X_{82}$  才能計算。

同理(R13,R23)、(R14,R24)、(R34,R44)。

(R23,R33)、(R24,R34)多 delay 兩級目的為使用 pipeline 技巧，讓原先需要八顆乘法器降成 4 顆乘法器。最後結果如(圖五)。



(圖四)



(圖五)

➤ 改良四：重新定義 K

從前文理論中，我們可以得知  $A_n$  如下：

$$A_n = \prod_n \sqrt{1+2^{-2i}}$$

其中 n 為疊代次數，將  $n=8$  帶入，並將其重新定義成 K，可得結果如下：

$$K=0.6074$$

最後將其轉為 FIXED POINT(共 11 bits, sign bit：1, fraction bit：10)

#### 四、 參考文獻

- [1] FPGA based Embedded Processing Architecture for the QRD-RLS Algorithm Deepak Boppana, Kully Dhanoa, Jesse Kempa Altera Corporation, San Jose CA
- [2] A survey of CORDIC algorithms for FPGA based computers Ray Andraka Andraka Consulting Group, Inc 16 Arcadia Drive North Kingstown, RI 02852 401/884-7930 FAX 401/884-7950