



應用於OFDM系統之FFT演算法硬體設計與比較

指導教授：朱元三 教授

專題生：徐碩廷、陳景琪

摘要

在現今的無線通訊中，正交分頻多工(Orthogonal frequency-division multiplexing, OFDM)傳輸有相當廣泛的應用，相較於傳統的FDM可以節省頻寬來提高頻寬的使用效率。而快速傅立葉轉換(Fast Fourier transform, FFT)在OFDM架構中用來取代振盪器為運算複雜度最高的部分，所以本專題在16-point的情況下透過不同之FFT演算法的硬體實現來進行比較，在考量精準度的情況下以降低功耗面積。

快速傅立葉轉換(FFT)演算法

離散傅立葉轉換 (Discrete Fourier transform, DFT)，是根據傅立葉轉換的定義來直接計算，而FFT則是在限制訊號數量為 N (N 為2的次方)的狀況下去進行迭代運算，將原本的DFT透過拆解來加快運算的速度，在硬體和軟體上都能節省資源。

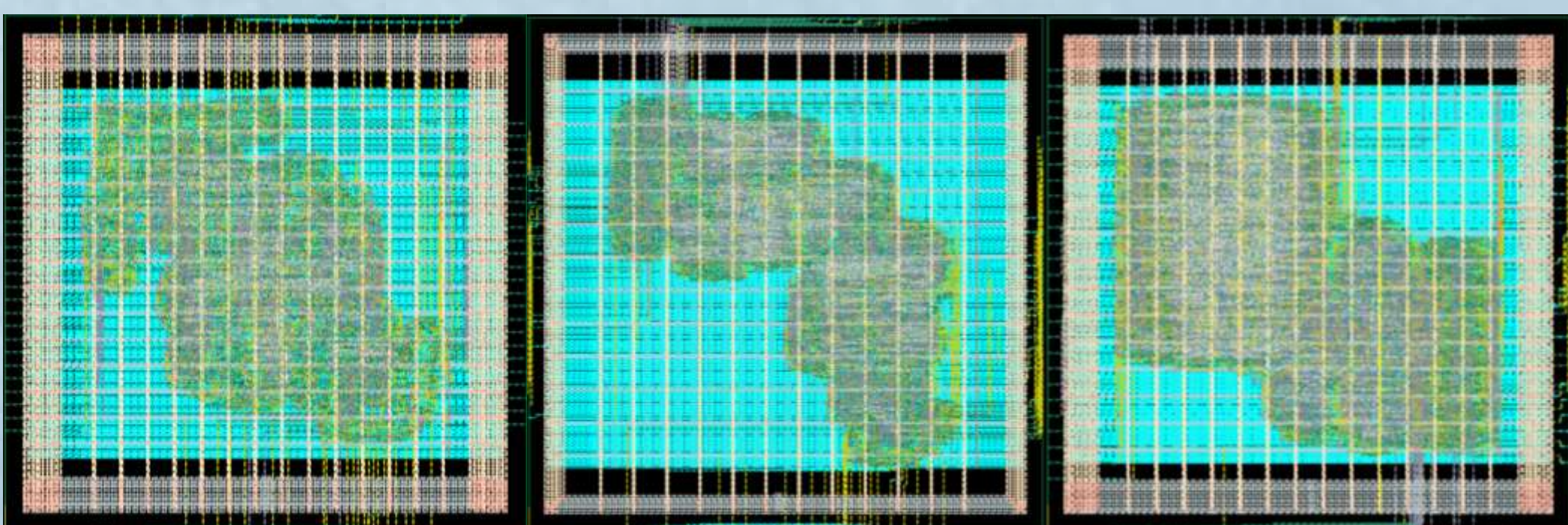
FFT的運算過程中旋轉因子(Twiddle factor)和蝴蝶運算(Butterfly operation)為相當重要的部分，本專題在16-point的基礎下整理出三種演算法(Radix-2、Radix 2^2 、Radix-4)，內部硬體的差別如下：

16-point / (N-point)	Stages	Butterfly 結構	旋轉因子 Complexity
Radix-2	$4 / (\log_2 N)$	簡單	$2 / (\log_2 N - 2)$
Radix-2 ²	$4 / (\log_2 N)$	簡單	$1 / (\log_4 N - 1)$
Radix-4	$2 / (\log_4 N)$	複雜	$1 / (\log_4 N - 1)$

Layout 結果比較

TSMC U18					
	Clock Rate	Clock Cycles / Total Time	Area (μm^2)	Gate Count	Power (mW)
Radix-2 ²	50M Hz (Max 66M Hz)	33 / 495ns	243268	25950	15.42
Radix-2	50M Hz (Max 52.6M Hz)	32 / 608ns	331876	35402	35
Radix-4	50M Hz	35 / 700ns	389737	41574	14.9

Layout



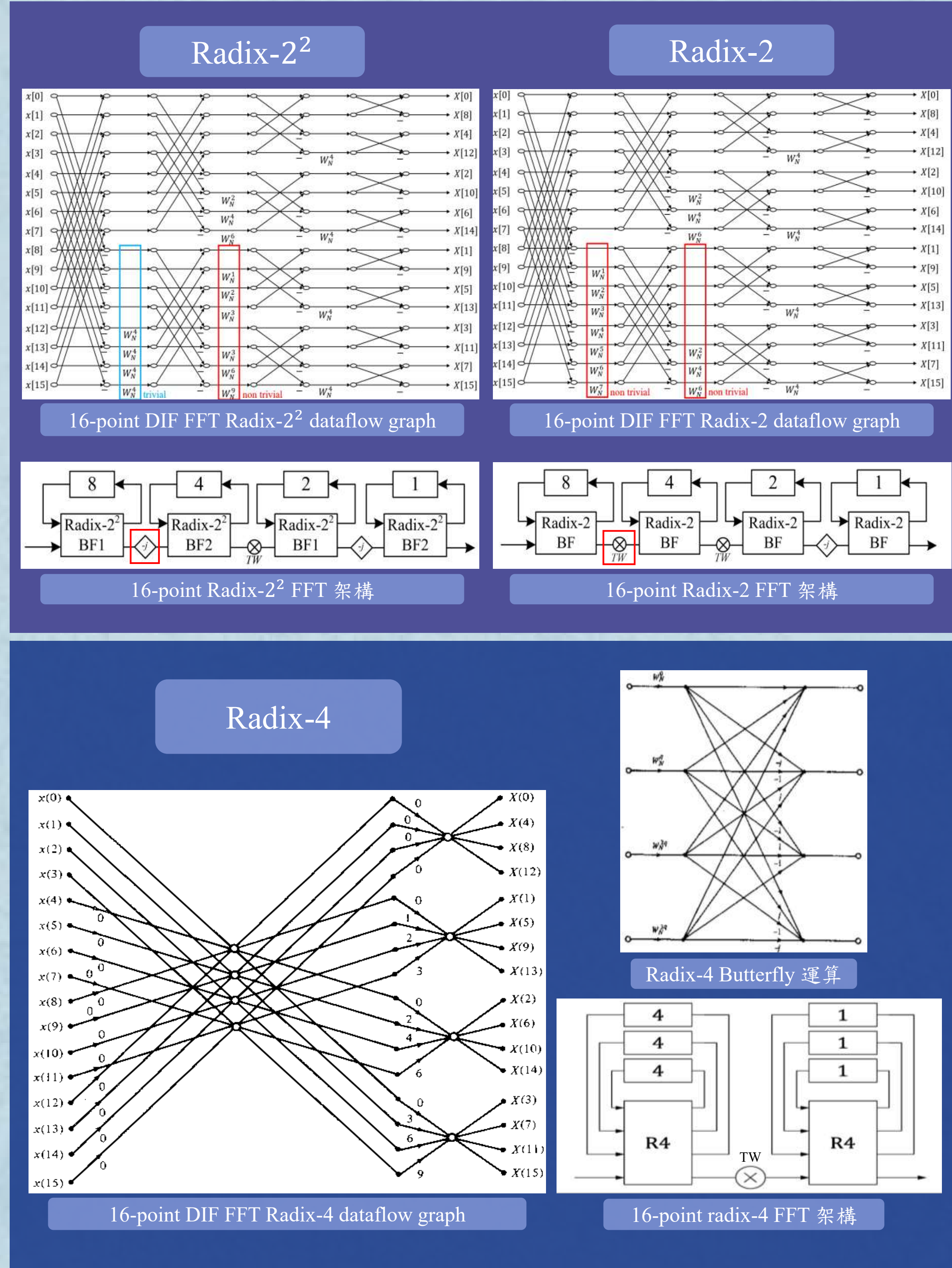
Radix-2²

Radix-2

Radix-4

系統架構

我們選用Single-path Delay Feedback(SDF)的方式來實現



結論

準確度：Radix-2², Radix-4 > Radix-2, 其中Radix-2²產生誤差的資料個數比Radix-4少，但誤差值較Radix-4大。

Radix-2：最原始的演算法，未經改良相較之下缺點較多。

Radix-2²：相較Radix-2，少一級旋轉因子，誤差較小，面積較小，功率消耗較低。

Radix-4：在16-point架構下，硬體成本最高，其餘表現與Radix-2²差不多，硬體設計最簡單，在資料數較多時較有優勢。

資料數較少的情況下選擇Radix-2²

資料數較多的情況下選擇Radix-4