# **SOC LAB**

Final project Report

Group 13 2024/01/04

## Topic:

- 1. 實現 fir,matmal,qsort 硬體加速器
- 2. 以 srdram 取代 bram
- 3. 在 uart 添加 reciver FIFO 與 transmitter FIFO

#### **UART**

## • Improvement:

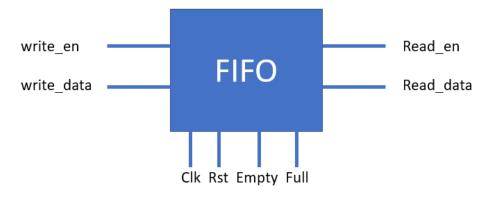
添加了 reciver FIFO 與 transmitter FIFO。

### ● 目的:

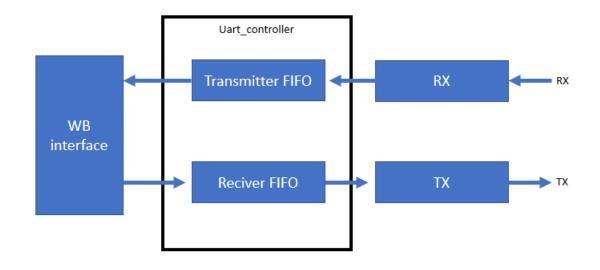
在尚未添加 fifo 時,當 rx 完成接收 data 時,會發出中斷訊號且 rx 會進入 wait 的狀態,並直到 data 通過 wishbone 離開時,rx 才會繼續接收下一筆 data。

加入 fifo 後,當 rx 完成接收 data 時,一樣會發出中斷訊號,但 data 直接存入 fifo,此時 rx 即可進行下一筆 data 的接收。

另一方面,在尚未添加 fifo 時,processor 須等待 tx 完成傳輸後才會接續工作。 但加入 fifo 後即可將 data 存入 fifo 中,而 processor 則可直接進行其餘工作。



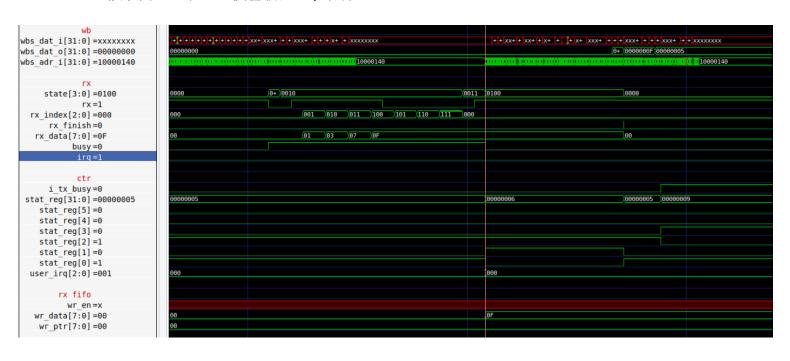
## ● 架構:



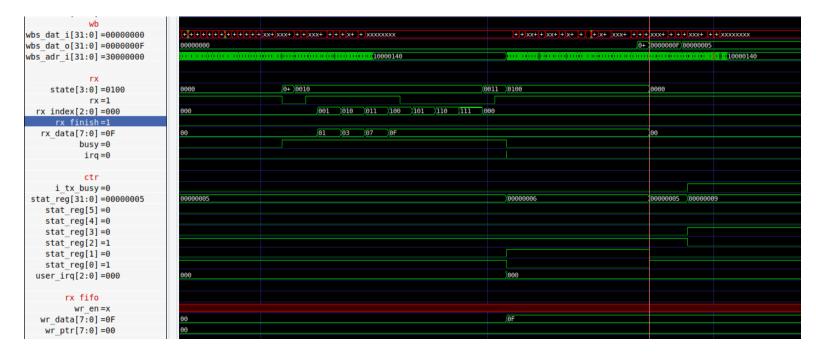
## ● 模擬結果:

#### 無添加 FIFO:

接收完 Rx 的 data 後會發送 irq 訊號

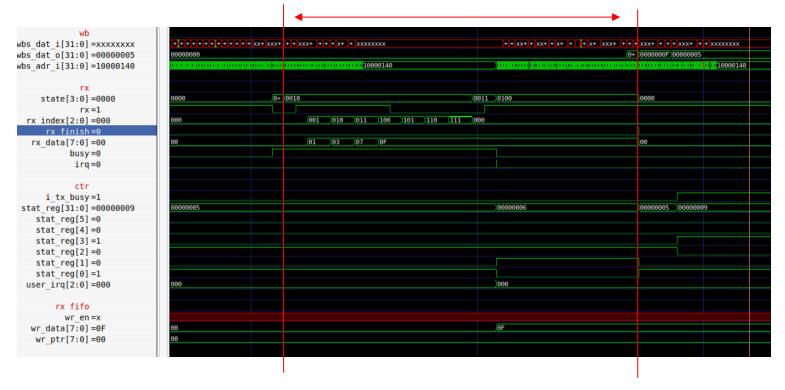


在 wb\_addr=3000 0000 時 rx\_data 會透過 wb 傳輸,完成後 finish 拉高。而 finish 拉高後 rx 可進行下一筆的 data 接收。

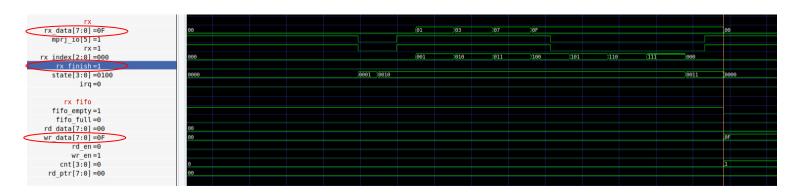


從 rx 完成 data 的接收後發出 Irq,到 controller 回傳 finish 時所經過的時間,約 630us。

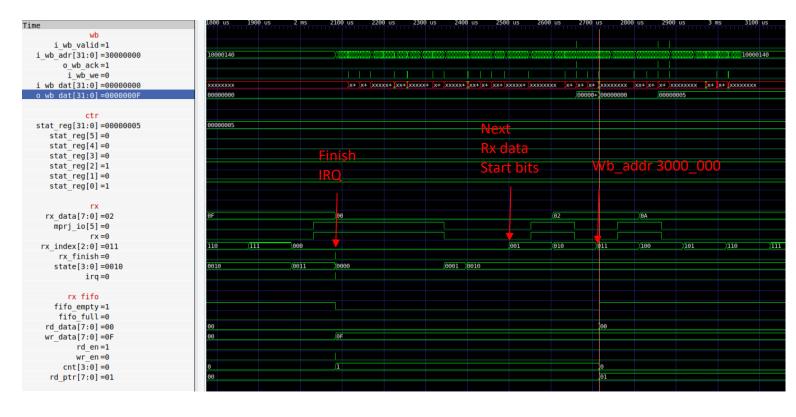
而 rx 完整一次傳輸,由 rx start\_bits 直到 data 送入 wishbone,約需要 2664us,(rx\_state 由 start\_bits 到 wait\_read 結束)



#### 添加 FIFO 後:



Rx\_data 接收完成後發送 irq 訊號且會存進 fifo,完成後 contoller 隨即發送 finish。 (在此 tb 預設在收到 finish 後 255us 發送下一筆 rx)



經過約 630us 後當 wb\_addr 為 3000\_0000 時,會觸發 FIFO 讀取 data 並透過 wb 送出 data。與無添加 FIFO 的波型圖相比, 添加 FIFO 後的 rx 在 wb 尚未送出 data 前就已在進行第二筆 rx\_data 的傳輸,大幅提升工作效率。

#### Observe:

可以依照需求來決定要每接收到一筆 data 就發送一次 interupt,或是當 FIFO 內存滿 data 時才發送 data。後者由於發送 interupt 的次數較少因此具有較高的效率。但由於時間的關係,本次實作並未將其實做出來。