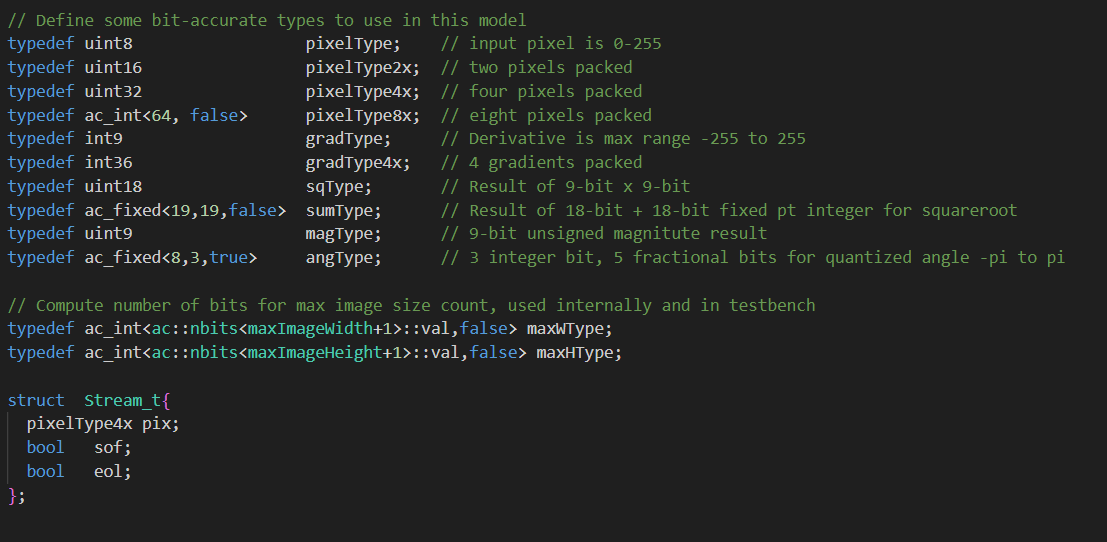
Edge detect Report

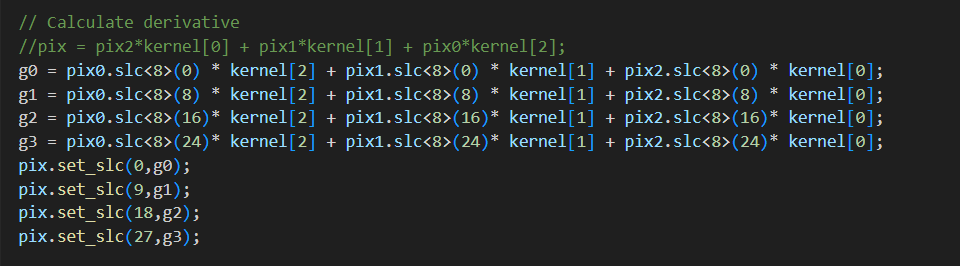
1. **5 modifications**
2. Process four pixels per clock cycle

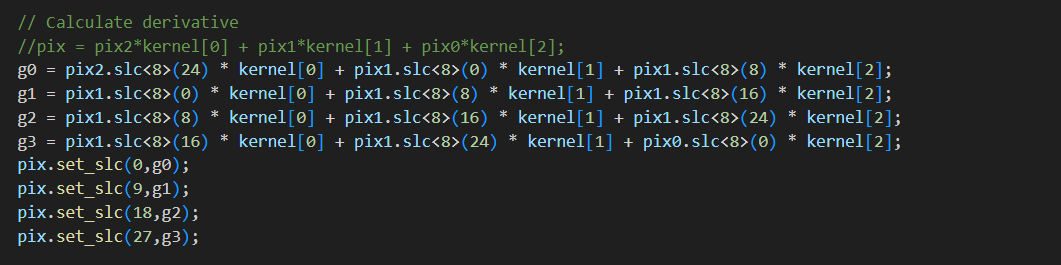
為了能夠一次傳送四筆pixel與gradient，我們定義了新的datatype



其中stream\_t包含4個pixel與tb會測試的sof與eol。

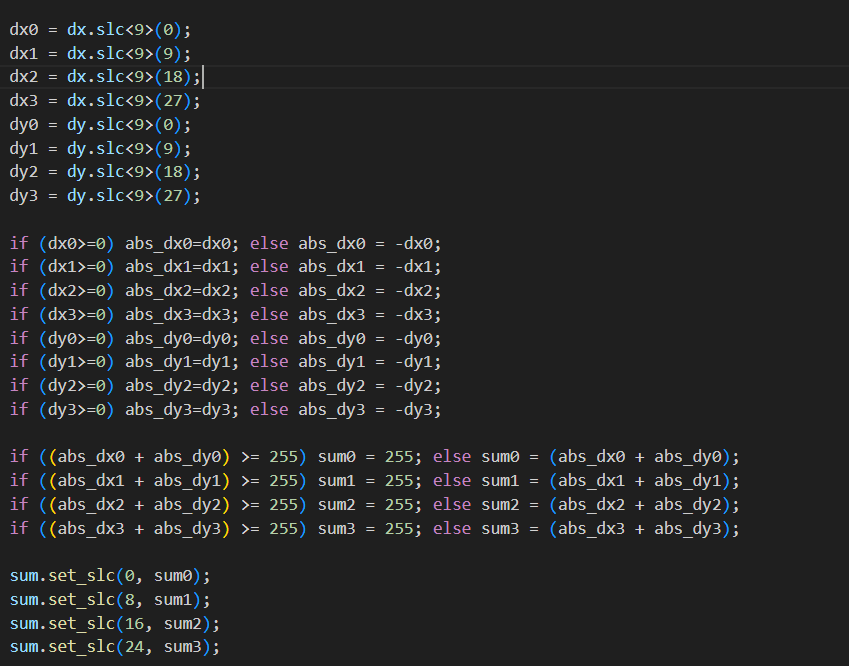
此外，在verder及horder中，會需要同時計算四筆pix經過filter得到的結果，我們採用手動unroll的方式，並利用.slc與.set\_slc來獲取對應的pix位置。





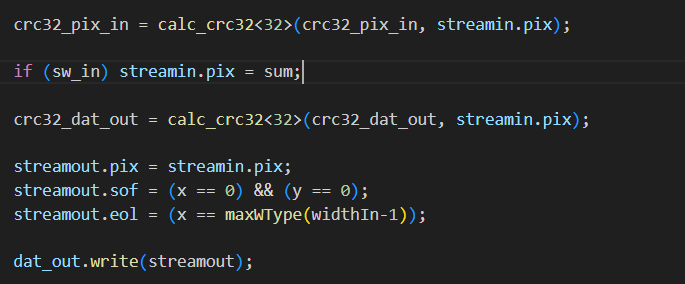
1. Use SAD

分別將四筆dx與dy取出後，透過手動的方式得到abs value，再做相加，並考慮overflow



1. Use crc32

如圖，使用calc\_crc32這個function對streamin.pix做一次，透過sw選擇streamout資料後，再對streamout.pix做一次。



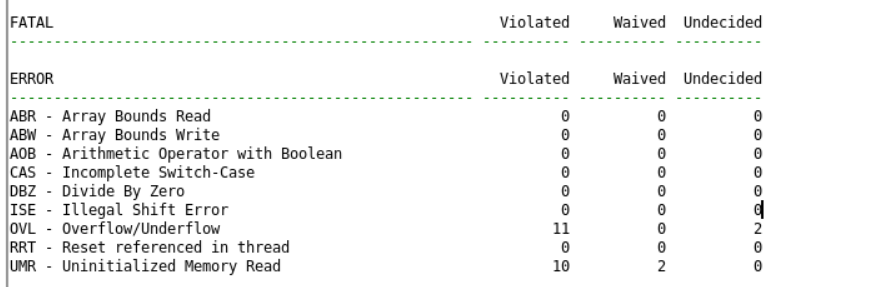
1. Select output source

如上圖，使用sw選擇streamout會是streamin或sum

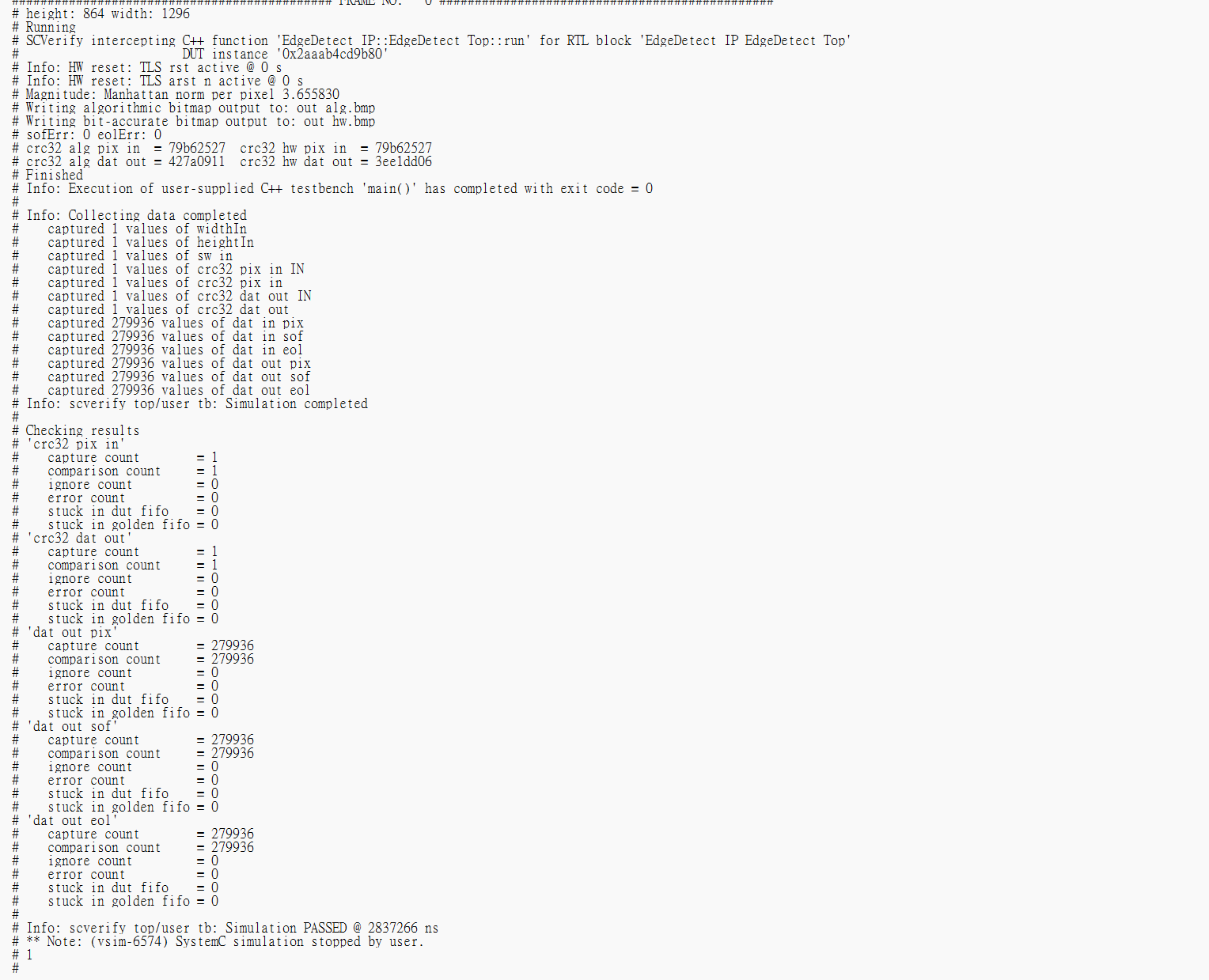
1. Remove angle

將angle caculation相關部分刪除。

1. catapult test result



檢查OVL部分都沒有甚麼問題，而UMR在講義中也出現，便先無視。



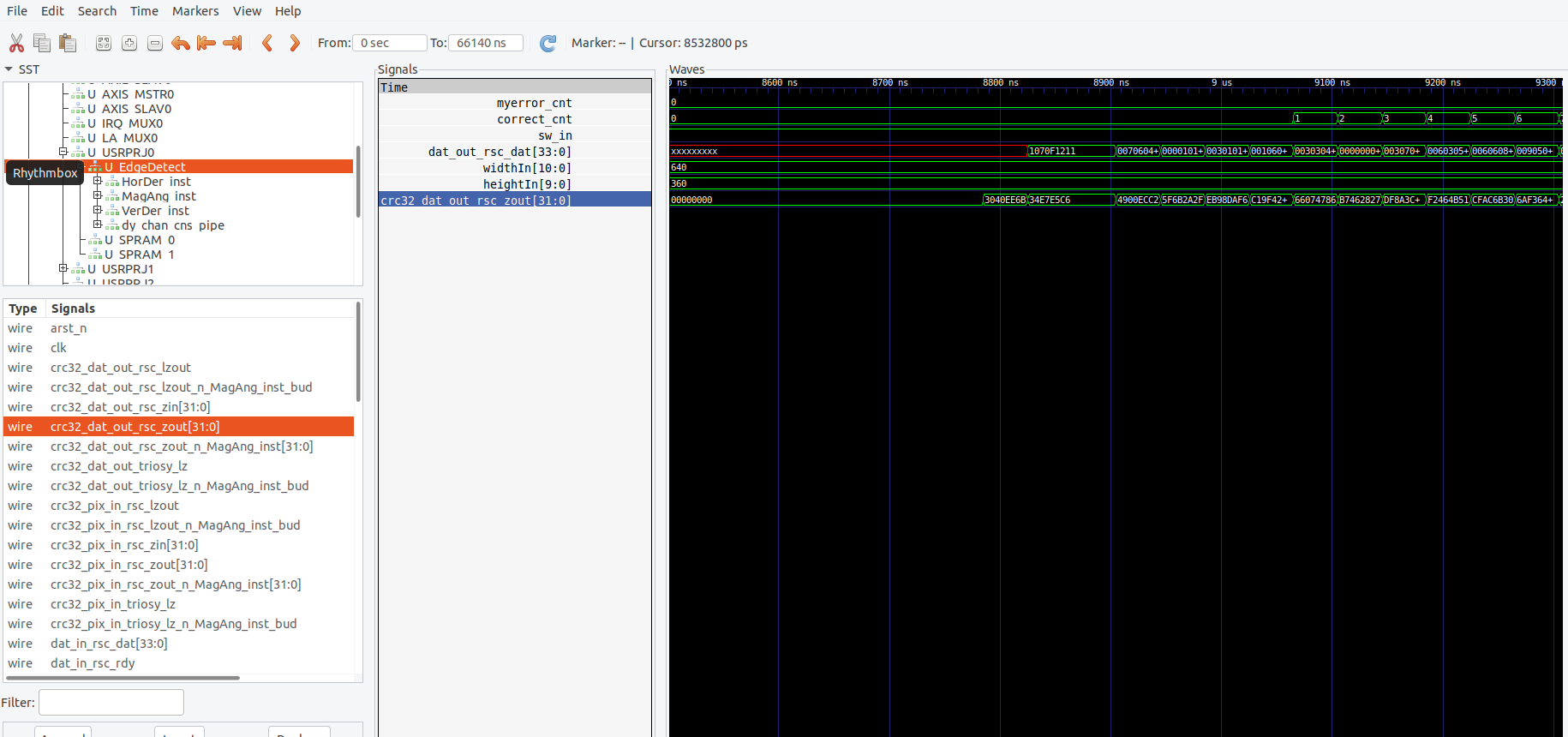
Questa sim結果與預期也都相同。

1. integrate to FSIC

使用lab fsic-sim的檔案架構，更改tb。在處理完reset之後，使用soc\_cfg\_write將圖片長寬與sw餵進去，並使用fpga\_axis\_req stream in data，在is\_as\_tvalid時比對data。

User\_prj0也需要修改，將edgedetect介面port都改成與HLS產生的verilog一致，並注意bit數。

1. simulation result of FSIC



比對golden data與user\_prj0產生的data，可發現都一致。

