**MSoC Self-paced 3: PP4FPGA CORDIC**

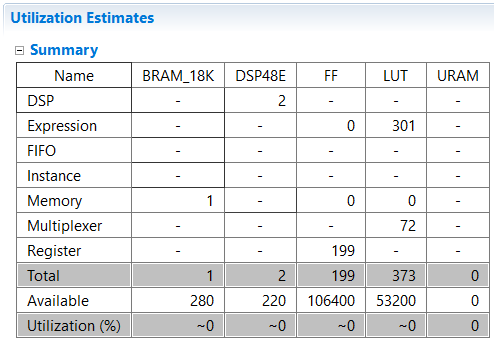
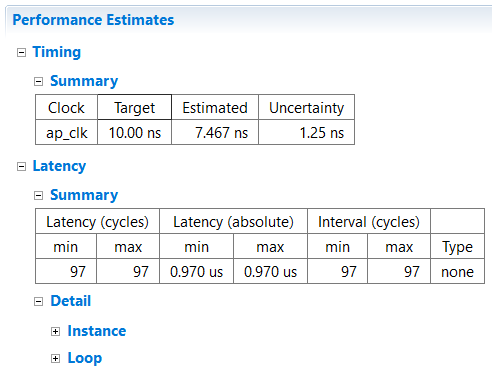
**R08943011 黃文璁**

* **[v] HLS C-sim/synthesis/cosim**
* **[ ] System bring-up (zedboard)**
* **[v] Improvement (latency, area)**
* **[v] Github:** [**https://github.com/b04901060/MSoC-Application-Acceleration-with-High-Level-Synthesis**](https://github.com/b04901060/MSoC-Application-Acceleration-with-High-Level-Synthesis)

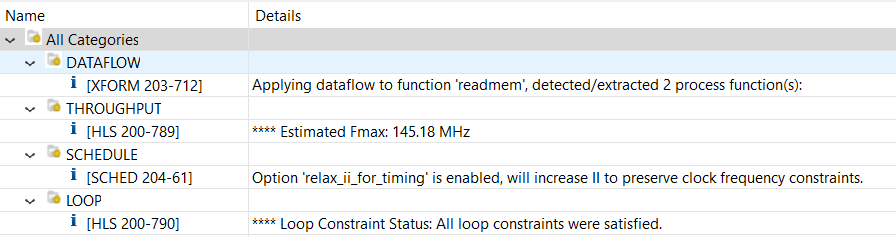
1. **Introduction**

本example實作CORDIC演算法，CORDIC透過shift運算來計算旋轉時需要的乘、除法，搭配預先計算的參數，如此一來可以用來近似cos, sin, cosh, sinh和除法等複雜函數。由於本實作中預設的fixed point精度較低，導致計算出來的error較大，我將精度由12bit增加至18bit進行模擬和合成。

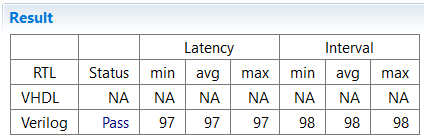
1. **C Synthesis and cosim**

****

合成結果沒有warning：



**Cosim:**

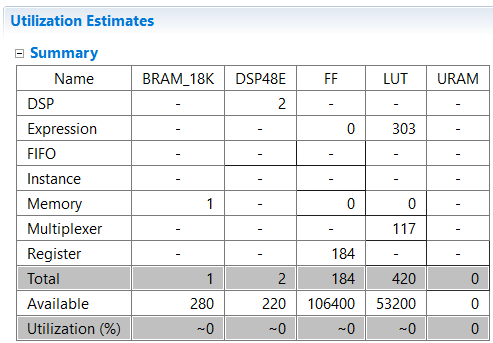
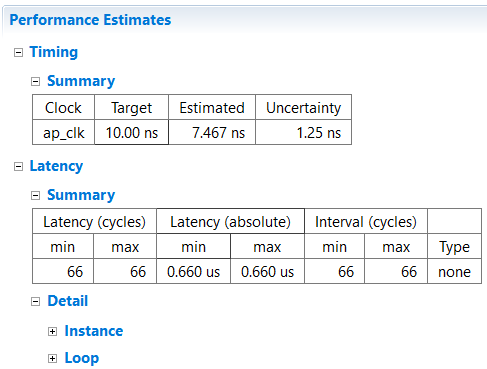


**-> Verilog pass**

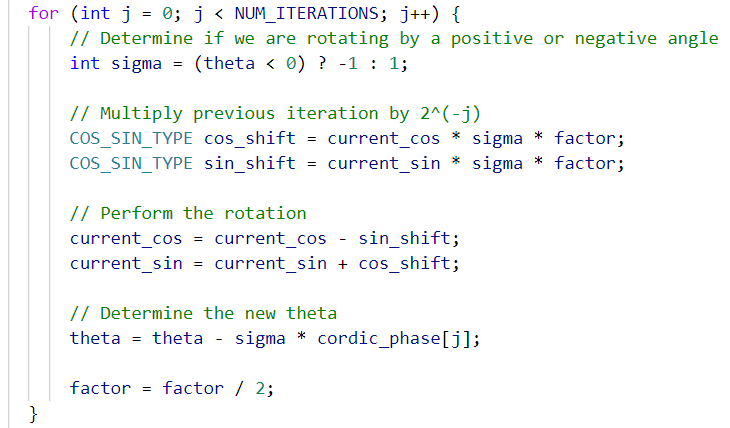
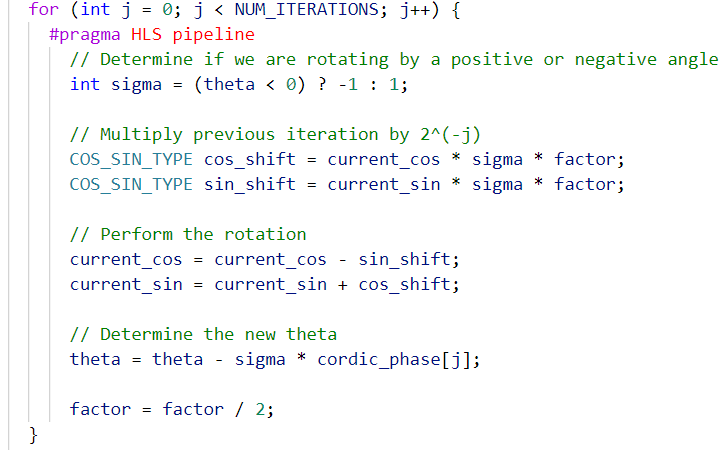
1. **Optimization**

由於CORDIC內部迴圈是常數iteration，可先採用unroll和pipeline等手段來加速，我嘗試在CORDIC的迴圈中加上pragma HLS pipeline指令來減少latency。

* + **Latency optimized synthesis result**

****

* + **Latency**: 97 -> 66 (**-32%**)
  + **FF**: 199 -> 184 (**+133%**)
  + **LUT**: 373 -> 420 (**+152%**)

**Before After**