

ミニマルEDA：設計ツール

株式会社ロジック・リサーチ

1. 総括

目的:フリーツールが使い物になるか、を判断をする

物理単位:ミニマルFabは、0.5inchウェハー(=直径12.5mm)なので最大9mm角(?)を製造できる。
ライブラリ(プロセスルール)に関わらず、1/9サイズの**3mm角は一括処理**して欲しい。

論理単位:**CPUコア部は一括処理**して欲しい。

時間単位:無制約なら半日。

面積制約、速度制約あっても、**丸1日で処理**して欲しい。

調査結果:①qflow、allianace共に、**32bitのシングルtask処理**

②は上記の3つの要求に、何とか合致 ←vivadoに比較すると貧弱だが

qflow_qrouter

<http://opencircuitdesign.com/qrouter/reference.html>のQAで、100Kゲートが限界の記が有る。

alliance

<http://www.cc.toin.ac.jp/sc/palacios/openbook/vlsie.pdf>のNEROの説明(p49にある1.20)では
最大4kゲート、**実用的には500ゲート**、で使うように記載が有る。

2. qflow課題 + 未だ、できていない事

課題①:TAT短縮に向け、マルチコア処理できないのか？

特に、qrouter。

⇒シングルコア & シングルtaskであれば、処理時間はCPUの動作周波数に依存。

課題②:3つの係数(density/rowSep/aspect)の関係性が不明。

⇒最適係数を見つけるのが、これまた大変。

課題③:qrouter_tkcon表示は、Nets_remaining=0だが

ツールの中で、無視されたネットは本当に存在しないのか？

課題④:qrouterの途中結果の保存と再開。

⇒write_def／read_defあるが、巧く使えない。

qrouter.sh起動の中身を変えると可能なのか？

課題⑤:DMA_64bit評価時に、STA結果が正常でないのに、なぜ次工程へ進めたのか？

課題⑥:速度制約を与えて、P&R収束できるのか？

課題⑦:graywolfによるビルディングブロック配置。 ←調査不足

課題⑧:ブロック間接続と遅延のバックアノテーション。 ←調査不足

各ツールの実行比較＆分析

- vivado
- qflow
- alliance

ただし、P&Rまで

3. 比較

ARM_M0コア部のみ	実行	CPU 処理	制約条件			P&R 実行時間	操作性	備考
			ライブラリ	速度	Pad			
Xilinx_vivado	64bit	マルチ	16nm	フリー	フリー	1分28秒	○ GUI	
Qflow:配置配線	※1 32bit	シングル	0.35um	フリー	フリー	102分 最適係数判明 後の最短時間	△ コマンド (最新は GUI?)	Qflow v1.1.112 Qrouter 1.3.33
Alliance ARM_M0をvbeに 変換できず 小さなmulti16で代用	※2 32bit	シングル	？？	フリー	フリー	直接の 比較データ 無	△ コマンド	V5.0-2012 0516-6 Debian向け
その他								

※1:make環境は64bitだが、router実行時のメモリ使用量2GB台より32bitプログラムと推定

※2:make環境は64bitだが、router実行時のメモリ使用量2GB以下より32bitプログラムと推定

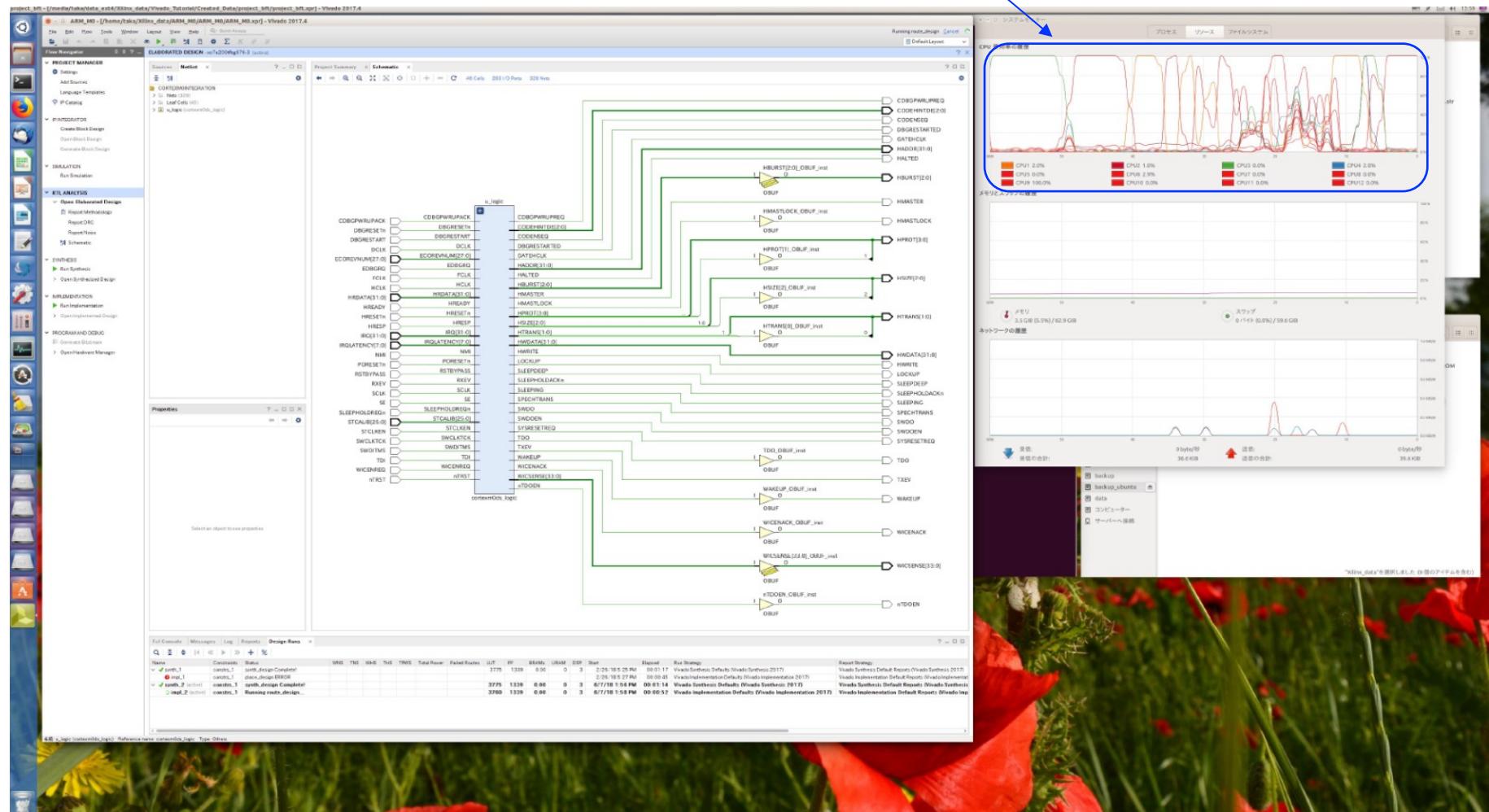
実行PC CPU:i7-7800K(6コア) * 3.7GHz
メモリ:DDR4-2666(64GByte)
Ubuntu16.04LT

※32bitプログラムだと、扱えるメモリが4GByteに制約され

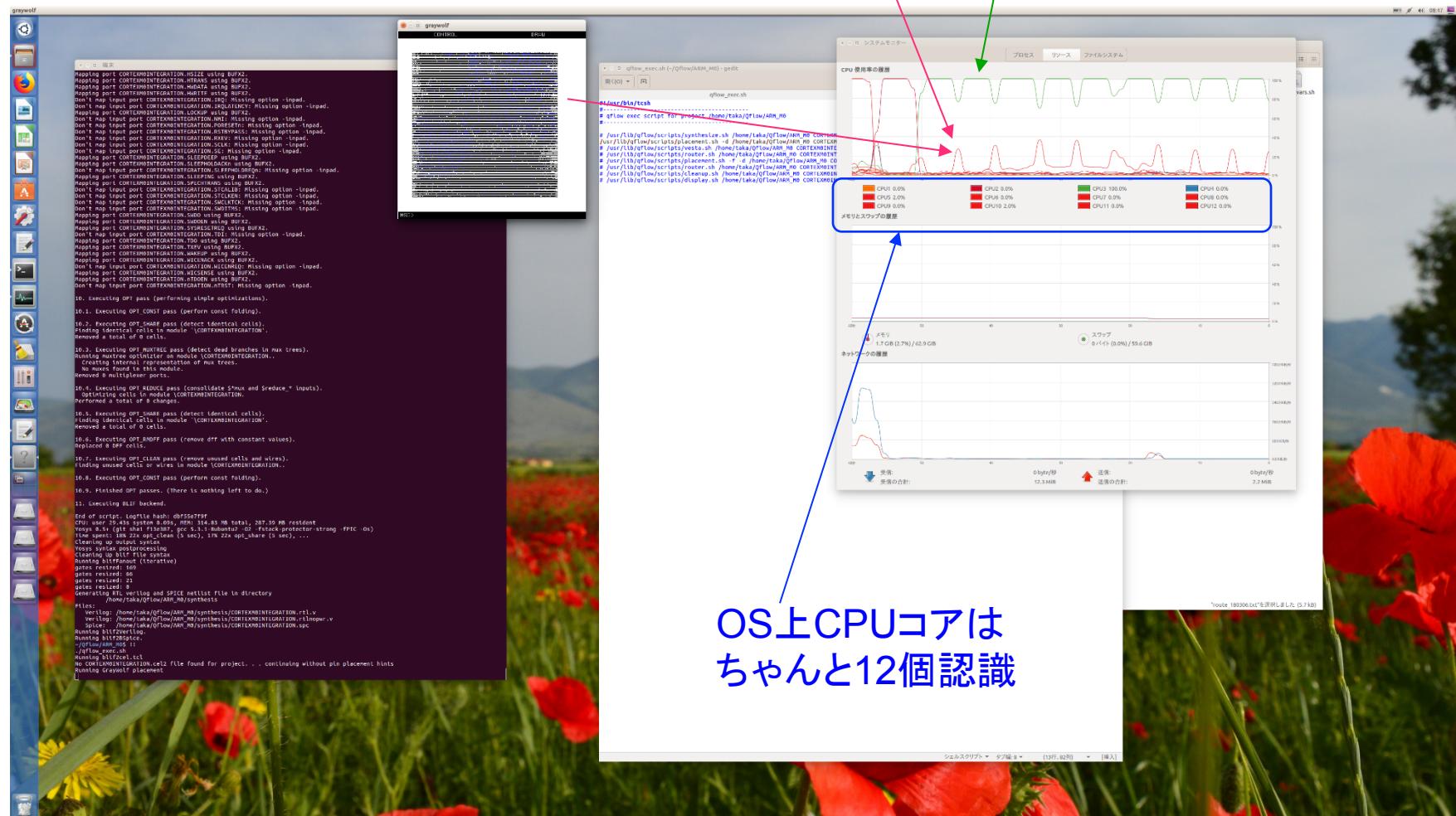
その中で、演算(配置配線+仮想遅延)を実施する。

4. Xilinx_vivado_impl実行

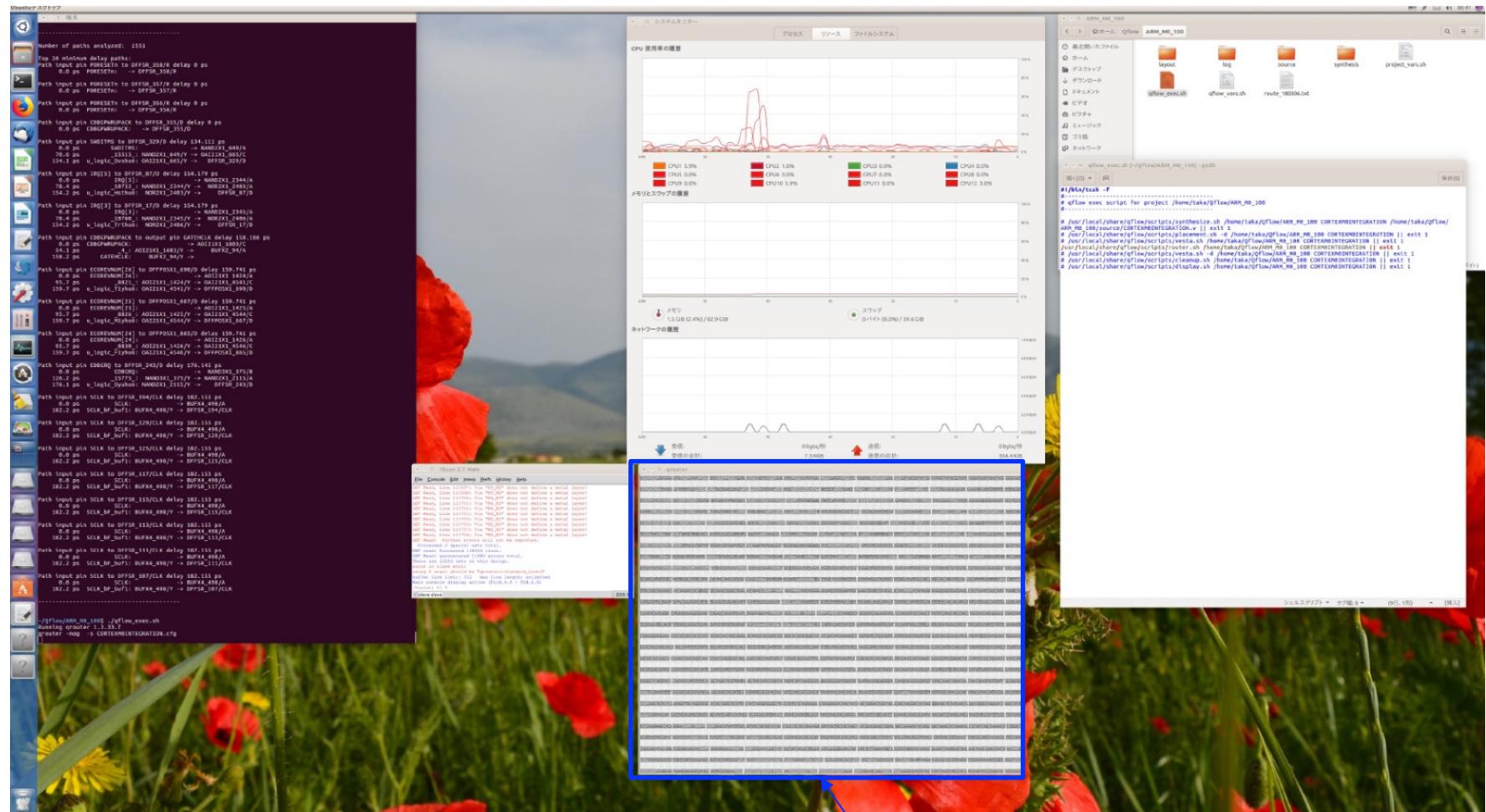
複数コアで実行



5 a. Qflow_placement : graywolf



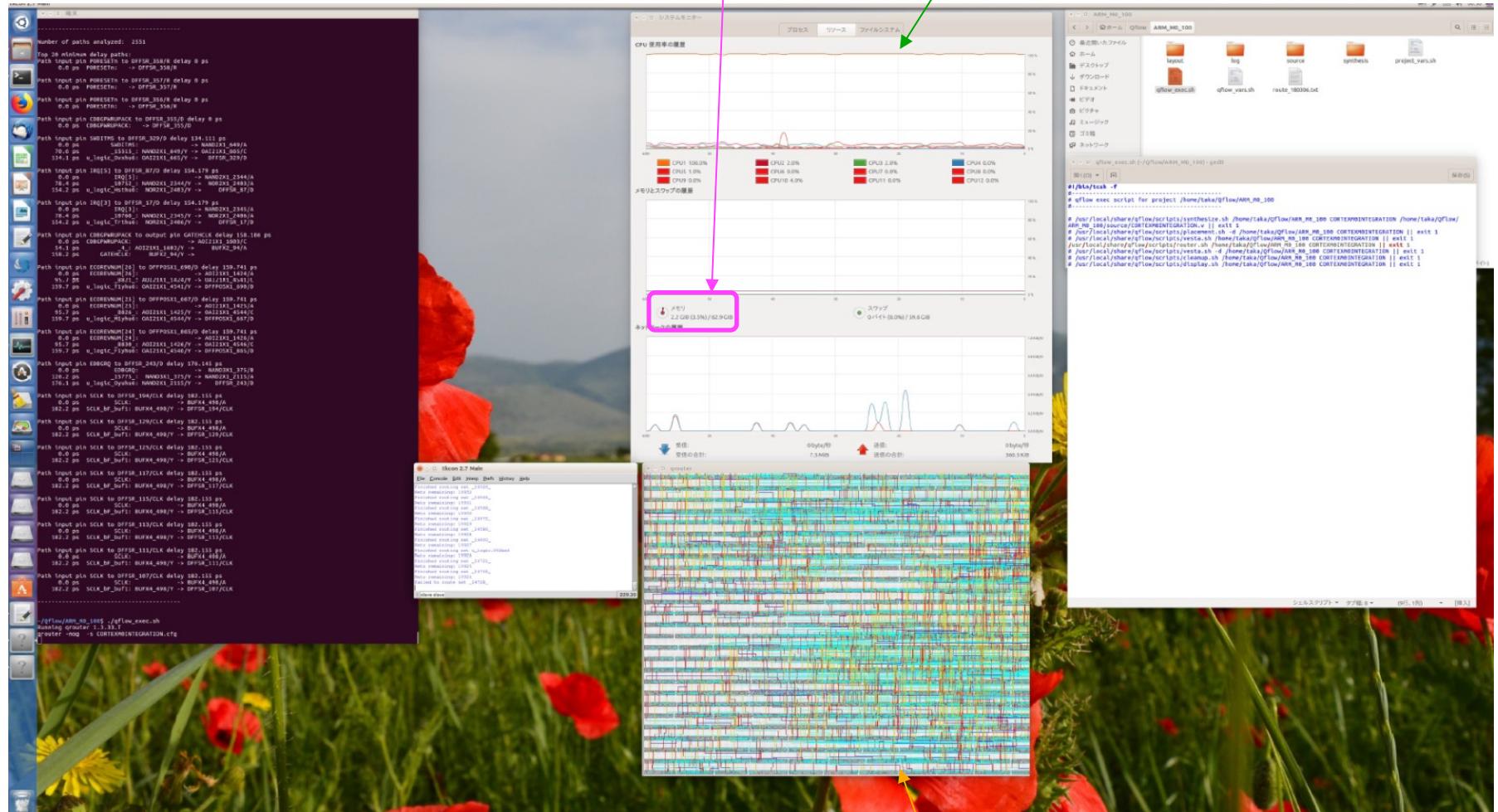
5 b. Qflow router : Qrouter



Placement終了時 未配線22000オーバー

Stage1 mask auto 実行中

使用メモリ2GB台 ⇒ 32bit空間 配置 = 1CPU実行

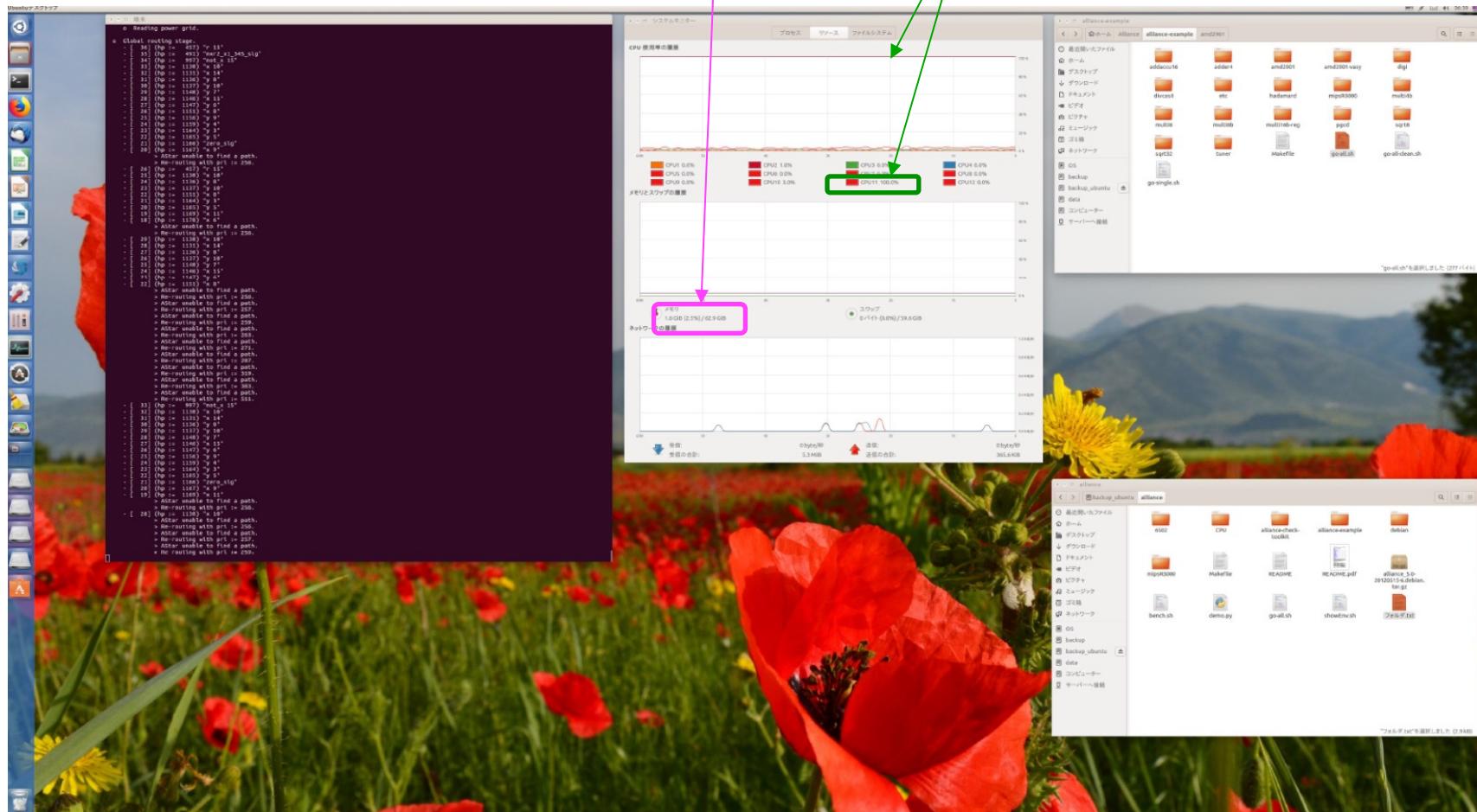


\ 縱配線：赤、黃綠
横配線：青、水色

6a. Alliance placement : ocp

使用メモリ1.6GB⇒32bit空間

配置=1CPU実行

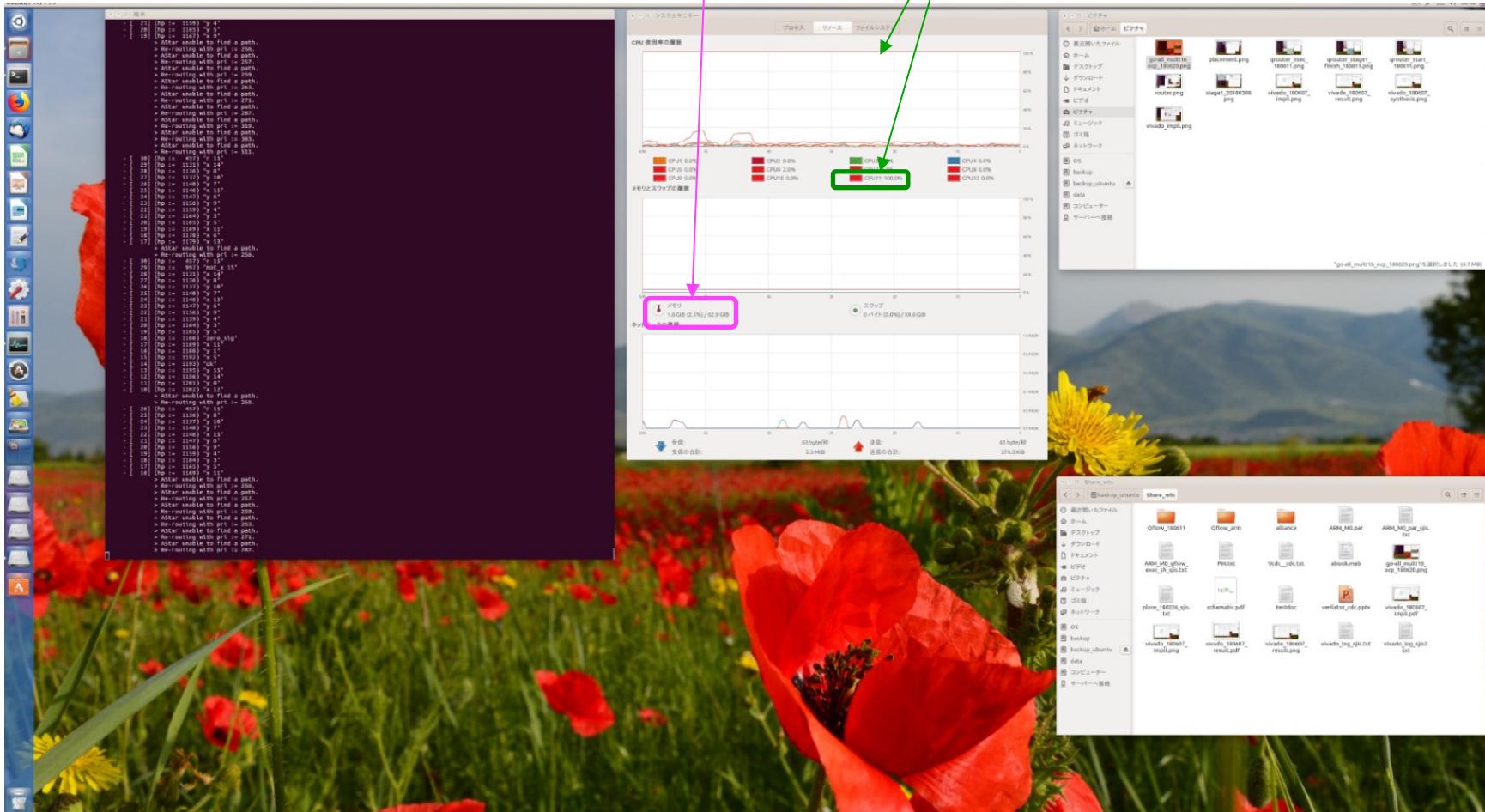


examples/go-all.sh 実行中のmulti16のocp(=実行に時間が掛かっている)でスクリーンコピー

6 b. Alliance_router : nero

使用メモリ1.6GB⇒32bit空間

配置=1CPU実行



examples/go-all.sh実行中のmulti16のnero(=実行に時間が掛かっている)でスクリーンコピー

Qflowベンチマーク

7 a. qflow明細・・・検討対象一覧

ARM-M0以外 OpenCoreにある素材	CPU コア				周辺				備考 取り得る値+機能
	openMSP430	ARM-M0 コア部	OR1200 コア部	DMAコントローラ	暗号 (AES)	DRAMコントローラ	メモリコントローラ	Ether	
公称回路規模		8kゲート	16kゲート	250kゲート	64bit版	38kゲート			
Xilinx vivado	topモジュール名			or2000_top	dma_ax64	aes_cipher_top			①階層把握・・・topモジュール名チェック
WebPack	implement LUT	106	3760	3347	2282	488	145	1108	+②回路規模
無償版	FF	192	1339	2231	1464	402	108	992	+③V/O 数チェック
	DSP	0	3	4	0	0	0	0	+④RTL文法チェック
	BRAM	0	0	SRAM 記述有 = 別建てになつて いる。FFはFF で構成	0	5 SRAM 定義無 →巨大case分岐 をvivadoが自動で 割付	0	0	4 SRAM 定義有
	実行時間	52秒	1分 28秒	1分 14秒	1分 14秒	1分 23秒	1分 15秒	1分 3秒	55秒

openMSP430とARM-M0以外に

追加調査として

視点 : 対象回路

大規模 : OR1200

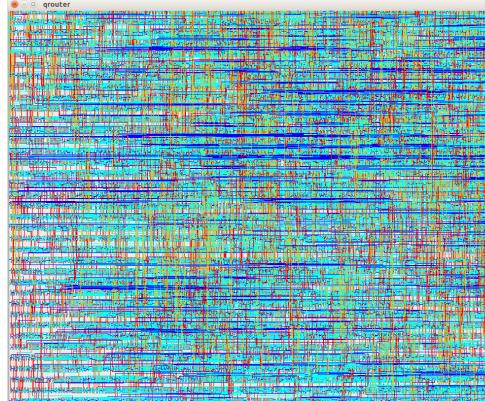
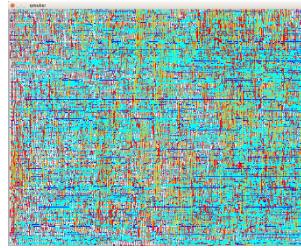
配線の束 : DMA-64bit

配線密度 : AES

を選択。

7 b. qflow実行結果

		CPUコア							備考 取り得る値+機能	
		openMSP430		ARM-CortexM0						
公称回路規模		8kゲート		16kゲート						
Xilinx_vivado webPack無償版 impliment	LUT	106			3760				実行目的 ①階層把握+②回路規模 ③I/O数+④RTL文法チェック	
	FF	192			1339					
	DSP	0			3					
実行時間		52秒		1分28秒						
Qflow	修正方向		縦横配線 スカスカ で開始	残1対策 縦配線増 大化	縦配線減 横長の方 に	さらに 縦配線 少なく	横配線 減らして 正方側に	縦配線増 横配線減 横長		
	Project_var.sh /density	0.7	0.55	0.52	0.6	0.65	0.6	0.55	0(緩い)～1(密着、デフォルト)	
	layout/xxx.apr /rowSep	0.3	1.2	1.2	1.2	1.2	1.1	1.1	0(なし、デフォルト)～1.5未満(緩緩)	
	layout/xxx.apr /aspect	0.8	0.75	0.75	0.7	0.7	0.75	0.65	0.75(デフォルト)	
	P&R合計時間	11分		212分	102分					
0.35umライブリ	dieサイズ	2.2x1.6	4.2x3.0	4.6x3.3	3.9x2.6			4.4x2.7	layout/xxx.defに出たサイズ 単位mm	



使用qrouterは1.3.33

7 c. qflow明細① · · · 16/32bit-CPUコア

ARM-M0以外 OpenCoreにある素材		CPUコア							備考 取り得る値+機能
公称回路規模		openMP430	ARM-CortexM0コア部						
Xilinx_vivado	topモジュール名	8kゲート	16kゲート						
WebPack	implement LUT	106	3760						①階層把握・・topモジュール名チェック +②回路規模
無償版	FF	192	1339						+③I/O数チェック +④RTL文法チェック
	DSP	0	3						
	BRAM	0	0						
実行時間		52秒	1分28秒						
Qflow	修正方向		縦横配線 スカスカで開始	残 1対策 縦配線増 巨大化	縦配線減 横長の方 に	さらに 縦配線 少なく	横配線 減らして 正方側に	縦配線増 横配線減 横長	
	Project var.sh/density	0.7	0.55	0.52	0.6	0.65	0.6	0.55	0(緩い)~1(密着、デフォルト)
	layout/xxx.apr/rowSep	0.3	1.2	1.2	1.2	1.2	1.1	1.1	0(なし、デフォルト)~1.5未満(緩緩)
	layout/xxx.apr/aspect	0.8	0.75	0.75	0.7	0.7	0.75	0.65	0.5~0.75(デフォルト)~1.5
	placement実行時間	3分	17分	16分	16分	18分	17分	17分	
router	配線本数	10701本	22264本・32bitCPUで32本の束が複数						
	stage1	mask auto	mask 8	mask 8	mask 8	mask 8	mask 8	mask 8	auto=1、接続するのに曲げる許容数
		残306本	残272本	残386本	残414本	残351本	残280本	残455本	隙間だけで接続
		3分	56分	82分	50分	118分	56分	63分	配線状況をGU表示させながら
	stage2	mask 32	mask 64	mask 64	mask 64	mask 32	mask 32	mask 64	auto=4、最大200?、255はNG
		残16本	残18本	残51本	残36本	残100本	残87本	残59本	邪魔な配線を差し替えて接続
		3分	39分	62分	41分	99分	68分	48分	
	stage2	mask 64	mask 128	mask 128	mask 128	mask 128	mask 128	mask 128	
		残3本	残6本	残15本	残2本	同じ場所	残23本	残15本	
		1分	7分	26分	7分	で配線	33分	9分	
	stage2	mask 128	mask 200	mask bbox	mask 150	入替	mask 200	mask 200	bboxは配線アルゴリズムが別版
		残0本	残2本	残0本	残1本		残17本	残2本	
		1分	2分	26分	2分		19分	2分	
	stage2		mask bbox		mask 200		mask NG	mask bbox	
			残1本		残0本		同じ場所	残2本	
					2分		で入替		
P&R合計時間		11分		212分	102分				
0.35um ライブリ dieサイズ		2.2x1.6mm	4.2x3.0mm	4.6x3.3mm	3.9x2.6mm			4.4x2.7mm	layout/xxx.defに出たサイズ
		3.5mm^2			10mm^2				

7 d. qflow明細②・・・大規模対応

ARM-M0以外OpenCoreにある素材		CPU					備考 取り得る値+機能
公称回路規模		0R1200 コア部 250kゲート					
Xilinx_vivado	topモジュール名	or2000_top					①階層把握・・・topモジュール名チェック
WebPack	implement LUT	3347					+②回路規模
無償版	FF	2231					+③I/O数チェック
	DSP	4					+④RTL文法チェック
	BRAM	0					
		SRAM記述有=別建てになっている、FFはFFで構成					
	実行時間	1分 14秒					
Q flow	修正方向				縦長	横長	
Project var.sh/ density	0.6	0.4	0.5	0.45	0.4	0.2	0(緩い)～1(密着、テフルト)
layout/xxx.apr/ rowSen	1.2	1.2	0.8	1.2	1.4	1.4	0(なし、テフルト)～1.5(未満、緩慢)
layout/xxx.apr/ aspect	0.7	0.7	0.6	0.85	1.2	0.6	0.5～0.75(テフルト)～1.5
placement 実行時間	途中で落ちる	42分	52分	55分	46分	62分	
router 配線本数		37816本					
	stage1	mask 8	mask 8	mask 8	mask 8	mask 8	auto=1、接続するのに曲げる許容数
		残643本	残810本	残807本	残779本	5時間	隙間だけで接続
		255分	169分	216分	154分	終了せず	配線状況をGUI表示させながら
	stage2	mask 64	mask 64	mask 64	mask 64		auto=4、最大200?、255はNG
		残76本	7時間	5時間	残158本		邪魔な配線を差し替えて接続
		194分	終了せず	終了せず	182分		
	stage2	mask 128			mask 128		
		残32本			残54本		
		46分			177分		
	stage2	mask bbox			mask bbox		bboxは配線アルゴリズムが別版
		残2本			残0本		
		40分			147分		
	stage2	mask 200					
		残2本					
		3分					
	P&R合計時間				660分		
0.35um ライブリ dieサイズ		7.1x4.8mm	5.4x4.1mm	5.8x4.7mm	5.7x6.5mm	11.2x6.5mm	layout/xxx.defに出たサイズ
					37mm^2		

7 e. qflow明細③・・・配線束と配線集中

ARM-M0以外Opencoreにある素材		周辺				備考
公称回路規模		DMAコントローラ 64bit版				
手修正	vivado カーボンプロファイルを与える+yosys_各ファイルにinclude追加 define.hに yosys向け	1)ファイル名変更 .64/.32のアンダーハード削除 2)モジュール dm_a_ax_64.vの中で、下位出力とショートを修正				yosysファイルスキャンがおかしい assignの3行をコメント
Xilinx_vivado	topモジュール名 WebPack	dm_a_ax_64 2282				①階層把握 ... topモジュール名チェック +②回路規模 +③I/O 数チェック +④RTL文法チェック
無償版	FF	1464				
	DSP	0				
	BRAM	0				
	実行時間	1分14秒				
Q flow	修正方向	縦長	横長	横長で全体縮小→横縮ま す		横長
	Project var.sh/densitv layout/xxx.apr/route layout/xxx.apr/aspect	0.6 1 1.2	0.4 1 0.8	0.5 0.9 0.65	0.45 0.4 0.8	0. 繁い)～1 密着(フォルト) 0 なし(フォルト)～1.5未満(緩緩) 0.5～0.75 (フォルト)～1.5
placement	実行時間	18分	18分	18分	18分	STAできず *1 STAできず STAできず STAできず STAできず
router	配線本数	22220本 ... 64bitバス				
	stage1	mask auto	mask auto	mask auto	mask auto	auto=1、接続するのに曲げる許容数 隙間だけで接続
		残350本	残485本	残599本	残512本	配線状況をGU表示させながら
		24分	36分	26分	30分	
	stage2	mask 8	mask 8	mask 8	mask 8	auto=4、最大200?、255はNG 邪魔な配線を差し替えて接続
		残133本	残182本	残141本	残256本	
		16分	25分	19分	27分	
	stage2	mask 128	mask 128	mask 128	mask 128	
		残14本	残17本	残14本	残22本	
		39分	67分	31分	72分	
	stage2	mask bbox	mask bbox	mask 200	mask bbox	bboxは配線アルゴリズムが別版
		残3本	残1本	残6本	残5本	
		25分	46分	18分	37分	
	stage2			mask bbox	mask 200	
				残1本	残1本	
				11分	5分	
	P&R合計時間				77分	
0.35umライド dieサイズ		3.3x3.8mm	4.8x3.7mm	4.8x3.0mm	4.6x3.6mm	5.4x3.3mm layout/xxx.defに出了したサイズ 17.8mm ²
結果考察		64bitを4層配線でやるのは難しい?				
		※ 1.syhth.bgの結果で、出力負荷が 0A121X1=7.4倍、DFFSR=5.2倍 付いている。 制約がデフォルトなので 1つのネットをバッファ分割できないようだ。 buffer amountで調整トライするも巧く行かず。				

ARM-M0以外 OpenCoreにある素材		周辺					備考 取り得る値+機能
		暗号(AES)			38kゲート		
公称回路規模							
手修正	vivado グローバルプロパティを与える+yosys 各ファイルにinclude追加 yosys向け						yosysファイルスキャンがおかしい assignの3行をコメント
Xilinx_vivado	topモジュール名	aes_cipher_top					①階層把握・・topモジュール名チェック
WebPack	implement LUT	488					+②回路規模
無償版	FF	402					+③I/O数チェック
	DSP	0					+④RTL文法チェック
	BRAM	5 SRAM 定義無 →巨大case分岐をvivadoが自動で割付 ※2					
	実行時間	1分 23秒					
Q flow	修正方向						
	Project var.sh/ density	0.6	0.5	0.6	0.5	0.4	0 緩い)~1 密着、(デフォルト)
	layout/xxx.apr/ rowSew	1	1.2	1.4	1.4	1	0 なし、(デフォルト)~1.5未満(緩慢)
	layout/xxx.apr/ aspect	0.7	0.65	0.8	0.75	0.6	0.5~0.75 (デフォルト)~1.5
	placement実行時間	22分	27分	24分	26分	23分	
router	配線本数	25717本 ..回路規模に対して配線数大 ※3					
	stage1	mask auto	mask auto	mask auto	mask auto	mask auto	auto=1、接続するのに曲げる許容数
		残1082本	残1130本	残1306本	残1250本	残1052本	隙間だけで接続
		38分	43分	35分	49分	42分	配線状況をGU表示させながら
	stage2	mask auto	mask auto	mask 8	mask 8	mask 8	auto=4、最大200?、255はNG
		収束NG	収束NG	収束NG	4時間	残530本	邪魔な配線を差し替えて接続
		同じ場所	同じ場所	同じ場所	終了せず	225分	
	stage2	で配線 入替	で配線 入替	で配線 入替		mask bbox	
						残13本	
						180分	
	stage2				mask 200	bboxは配線アルゴリズムが別版	
						残4本	
						4分	
	stage2				mask bbox	残4本	
						2分	
	P&R合計時間						
0.35um ライブドリ	dieサイズ	4.2x2.8mm	5.2x3.2mm	4.0x3.1mm	5.0x3.6mm	5.6x3.3mm	layout/xxx.defnに出たサイズ
結果考察	配線密度が高く、セルへ接続できない?						
		※2 暗号の"Sbox" = 6入力 4出力が多数。 vivado=メモリに割当 yosys=純粋に論理展開→配線爆発					※3 試しに、SboxのRTL記述を 入力8bitを、そのまま8bitで返す、に修正。 理論上で6bit入力の4bit出力だが) 配線本数は 5613本 (ク2%)に激減。

Qflowノウハウ

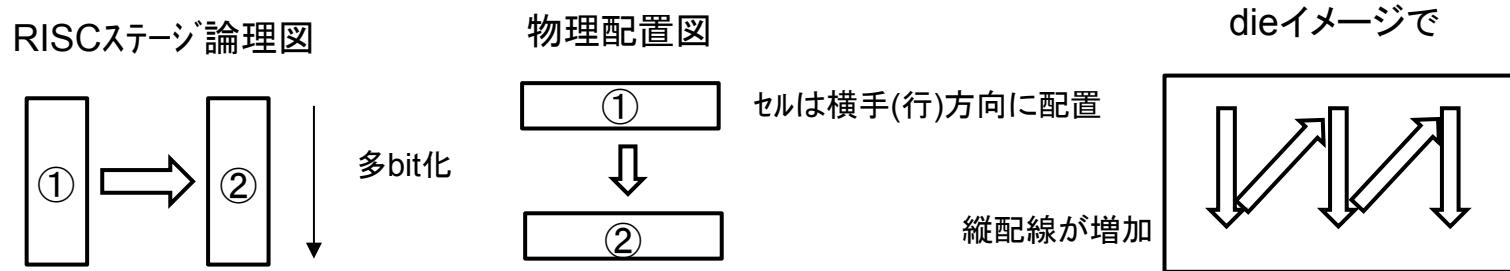
8. qflowノウハウ・・・独自解釈

(ア)placement

- ・縦配線の確保=home/project_vars.shのset initial_density:0(緩い)～1(密着、デフォルト)
- ・横配線の確保=home/layout/openMSP430.parの*rowSep:0(なし、デフォルト)～1.5未満(緩緩)
- ・アスペクト比=home/layout/openMSP430.parのTWMC*chip.aspect_ratio:0.75(デフォルト)±0.25
⇒数値組合せが悪いとqrouting起動時にtap_pointエラーが発生する ←ルール不明

示唆1: routerの配線図を見て行間に隙間の空白が多い=横配線がゆるゆるなら
アスペクト比は小さく(=横長に)するのが良い。

示唆2: CPUのbit数が大きい場合、bit数に応じた束が、次段へ接続されるので
縦配線を十分に取る必要がある →アスペクトは横長が良い。デフォルトは横長の0.75



アルゴ推測: densityで指定された密度で、セルを一度、横一列に並べて
rowSepで指定された高さを加えた後
chip.aspect_ratioに合うように並べ変える。

(イ)qrouting

stage1/2で与えるmask_nnの数字nnは、何回まげて接続して良いかの数字。
stage2のmax数は200くらい(?)、255は駄目だった。

(ウ) 収束NGの兆候

Nets remaining: 14
Nets remaining: 13
Nets remaining: 12
Best route of _30575_ collides with nets: _30354_<17>
Ripping up blocking net _30354_<17>
Nets remaining: 12
Best route of _30507_ collides with nets: _10_<23>
Ripping up blocking net _10_<23>
Nets remaining: 12
Best route of clk_i_bF\$buf125 collides with nets: _12344_
Ripping up blocking net _12344_
Nets remaining: 12
Best route of _30719_ collides with nets: _30718_ _26883_bF\$buf98 _30418_
Ripping up blocking net _30718_
Ripping up blocking net _26883_bF\$buf98
Ripping up blocking net _30418_
Nets remaining: 14
Nets remaining: 13
Nets remaining: 12
Best route of _30509_ collides with nets: _30355_<30>
Ripping up blocking net _30355_<30>
Nets remaining: 12
Nets remaining: 11
Best route of _30653_ collides with nets: _30707_ or1200_tt.ttcr<23> _30509_ _439_ _30719_
Ripping up blocking net _30707_
Ripping up blocking net or1200_tt.ttcr<23>
Ripping up blocking net _30509_
Ripping up blocking net _439_
Ripping up blocking net _30719_

Nets remaining: 15
Best route of _30413_ collides with nets: _30355_<31>
Ripping up blocking net _30355_<31>
Nets remaining: 15
Nets remaining: 14
Nets remaining: 13
Nets remaining: 12
Nets remaining: 11
Best route of _12344_ collides with nets: _12340_ _226_<9>
Ripping up blocking net _12340_
Ripping up blocking net _226_<9>

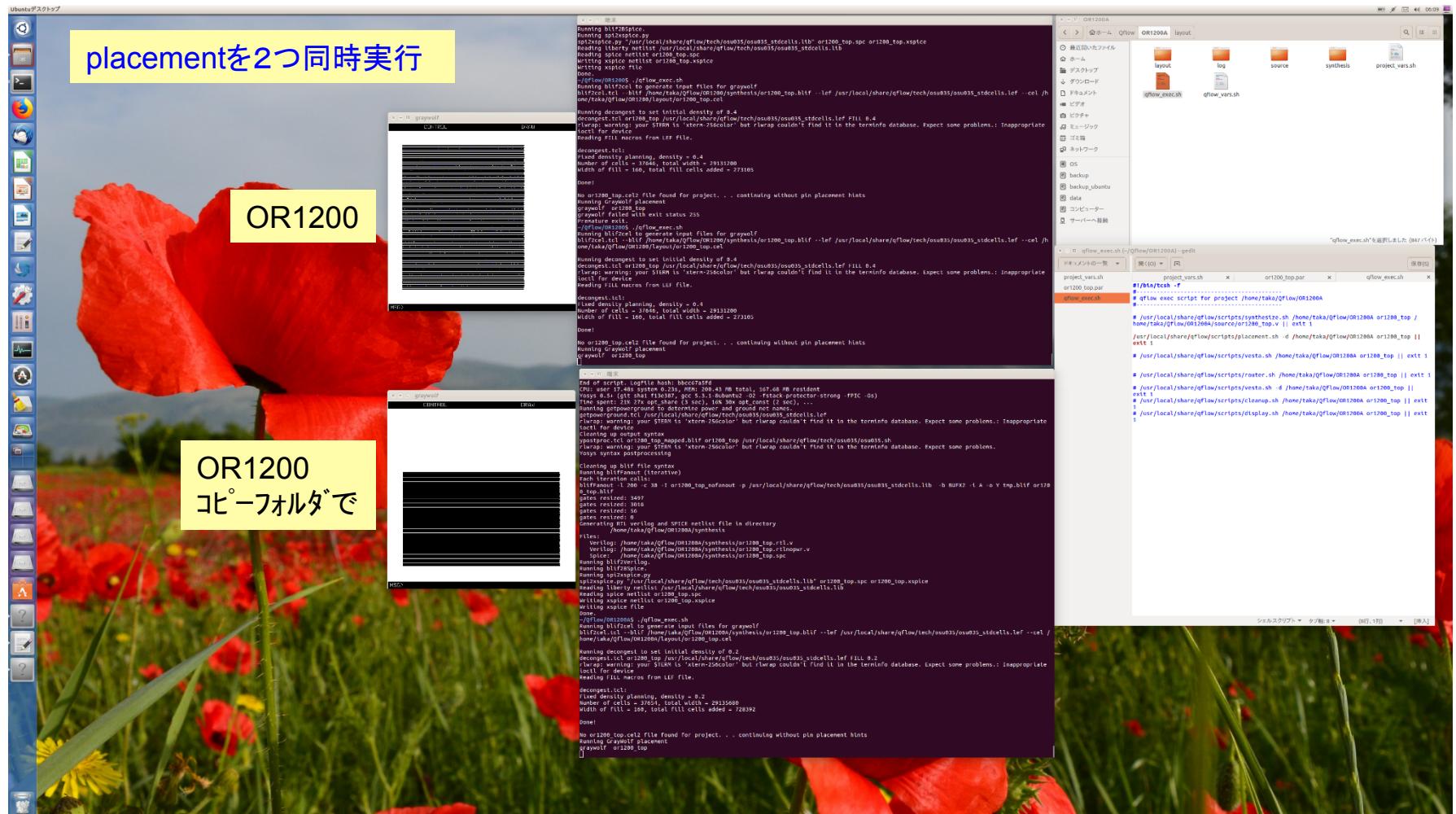
○ 置いたネットを
直ぐに剥ぎ取っている

○ 剥ぎ取ったネットで
残ネット数が、元に戻る=増加

(工)係数探索の効率化

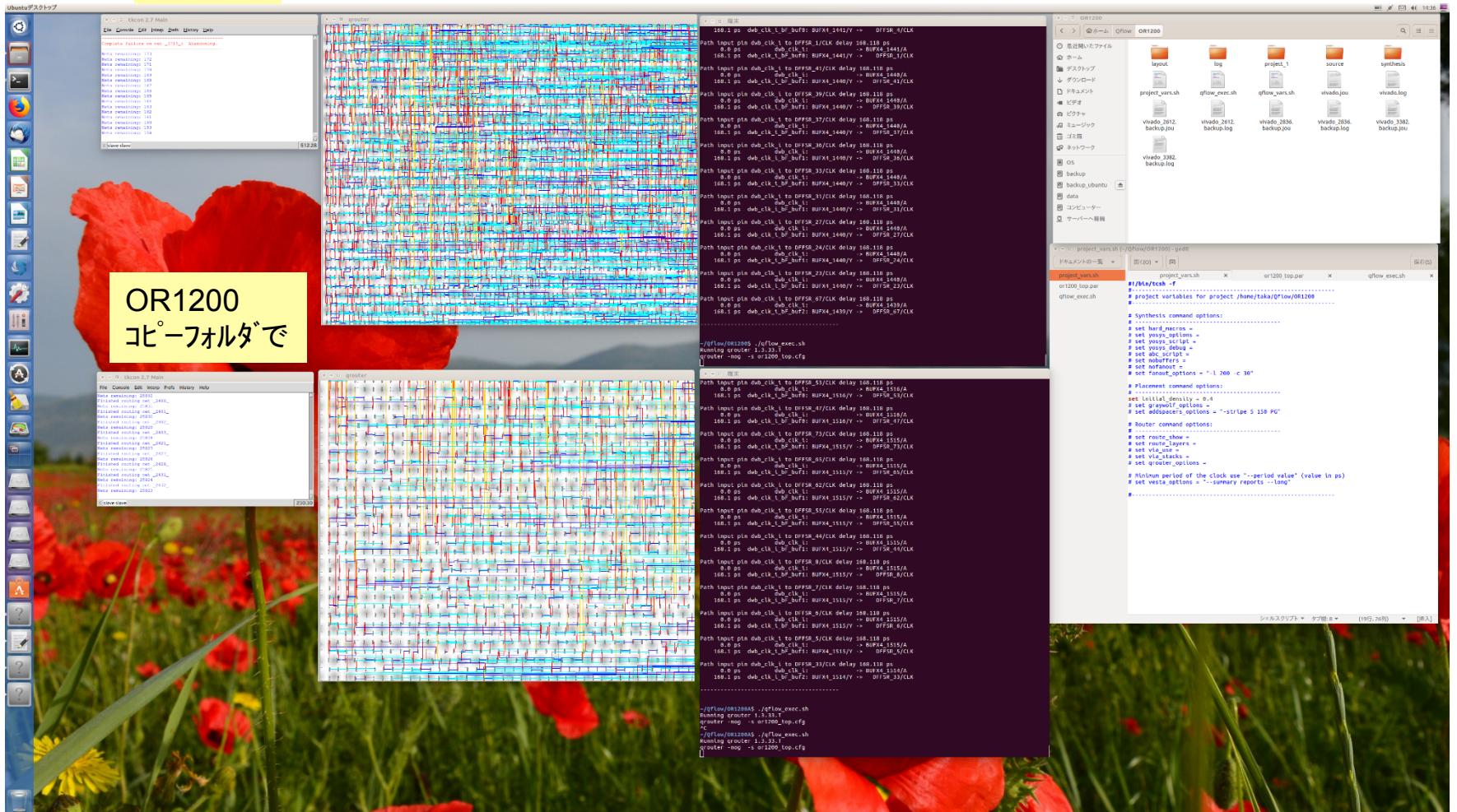
qflow処理は、シングルコアでシングルtaskなので、[マルチ実行](#)が可能なはず、と推測。

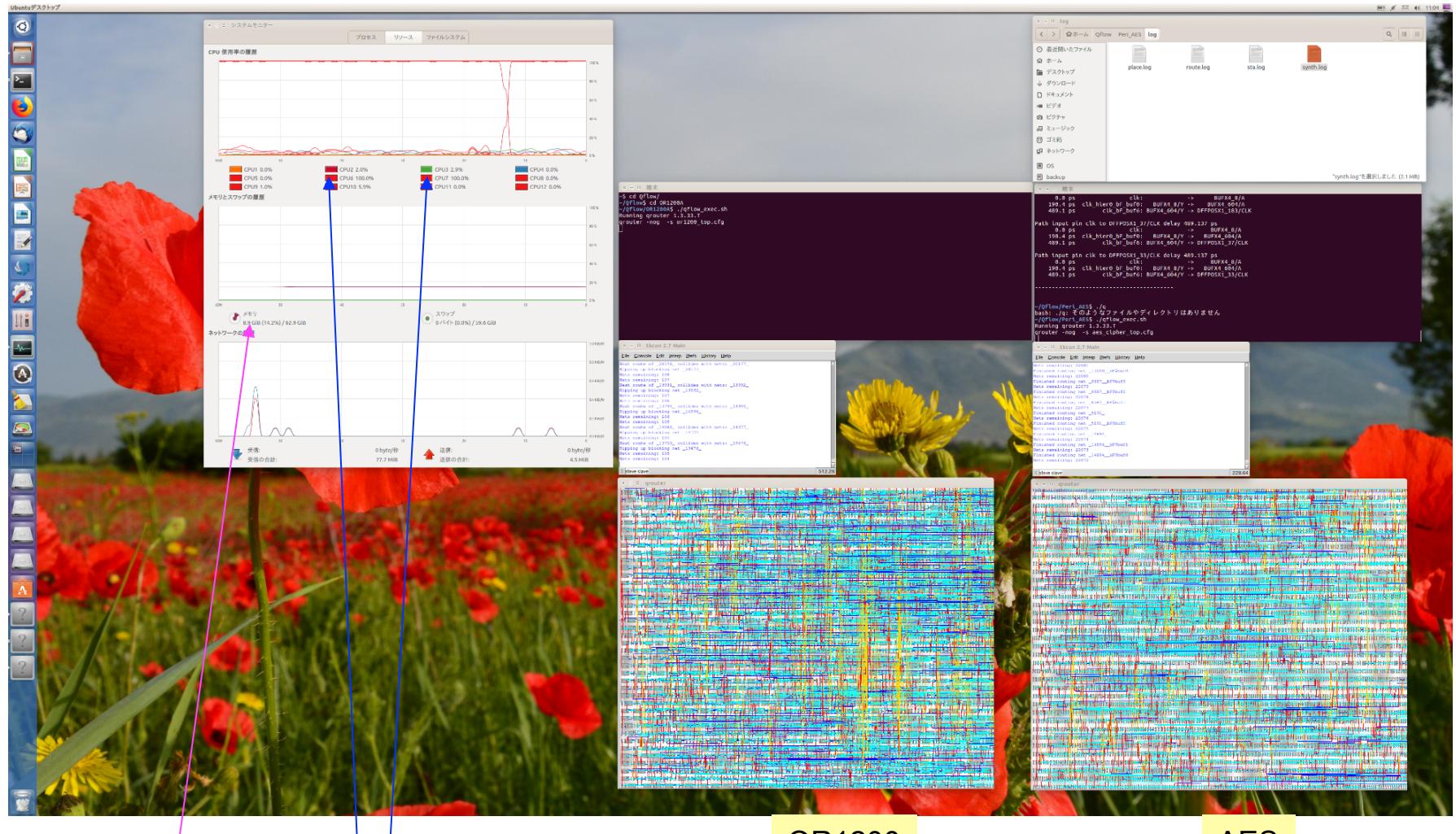
- ⇒ ・フォルダが違うと同時実行可能…placementとrouterで確認済
・同一RTLでも、フォルダを別名(RTL_topは同一名称)にしてコピーすると同時実行可能



qrouterを2つ同時実行

OR1200





使用メモリ
約9GB

2つのコアが
CPU負荷100%

OR1200

AES

qrouterを2つ同時実行