



# Linuxが動作するRISC-Vソフトプロ セッサRV-PCのFiC FPGAボードへの 移植 ← OS屋がFPGA RISC-Vに手を出した話

並木 美太郎 (Mitaro Namiki) |  
東京農工大学大学院工学府 教授、  
三好 健文 (Takefumi Miyoshi) |  
わさらぼ合同会社、  
天野 英晴 (Hideharu Amano) |  
慶応大学 教授

# 簡単な自己紹介:場違い?

- 専門はOSなどシステムソフトウェアを30年程。学生時代1980年代90年代前半までゼロからOS、Cコンパイラ作っていた(OS/omiconプロジェクトM68000-)。
- 1986年日立製作所基礎研究所入社で冬の時代の前のAI研究、数年勤めて大学助手へ
- 10年ほど前に省電力CRESTプロジェクトでpower gatingのプロセッサ研究(Geyser)に参加
- Power gating MIPSをFPGAに乗せて、Linuxを移植。その後、実チップに。同業者からは変人扱い
- 4年前に脳出血後回復、社会復帰で教授継続以前より実装能力落ちた
- 定年も近いし、最後のご奉公、RISC-V見ておこうか



# 1. はじめに.: 目的1

---

- OSなどのシステムソフトウェアの研究では、新しいハードウェアに対する仮想化、資源管理の方式を考察したい。また既存のCPUの拡張を行って方式の有効性と性能向上を考察したい
- OSの研究者なので、比較的容易にCPUを修正して検証できる環境を入手したい。



## 目的2

---

- MEC (Multi-access Edge Computing)用のFPGAクラスタFiC (Flow-in-Cloud)は、各ボードにRaspberry-Pi3ドーターボードを搭載している
- Raspberry-Pi3とFPGA上のロジックはGPIOで接続  
→ 転送性能が貧弱
- FPGA内にソフトプロセッサを搭載すれば、内部のロジックと密に結合できる

# MEC(Multi-access Edge Computing)

## JST CREST Society 5.0を支える革新的コンピューティング技術「MEC用マルチノード統合システムの開発」

5G : low latency (< 0.5msec) and high bandwidth  
→New applications treating timing-critical multiple jobs.  
Low energy and low cost



- .
- Sun, Y., Amano, H: FiC-RNN: A multi-FPGA acceleration framework for deep recurrent neural networks, IEICE Transactions on Information and Systems Volume E103D, Issue 12, 1 December 2020, pp.2457-2462, 2020.12
- 手塚健介・天野英晴: AlexNetのマルチFPGAシステムへの分割検討と実装, 信学技報, vol. 119, no. 147, CPSY2019-23, pp. 91-95, 2019.7.

# RISC-V on FiCボード

- RISC-Vを搭載したFPGAボードは多数存在する
- FPGAクラスタの各ボードでRISC-V搭載可能なシステムは存在しない
- RISC-Vを搭載するかどうかボードごとに選択可能

FiCボード



FiC FPGAクラスタ



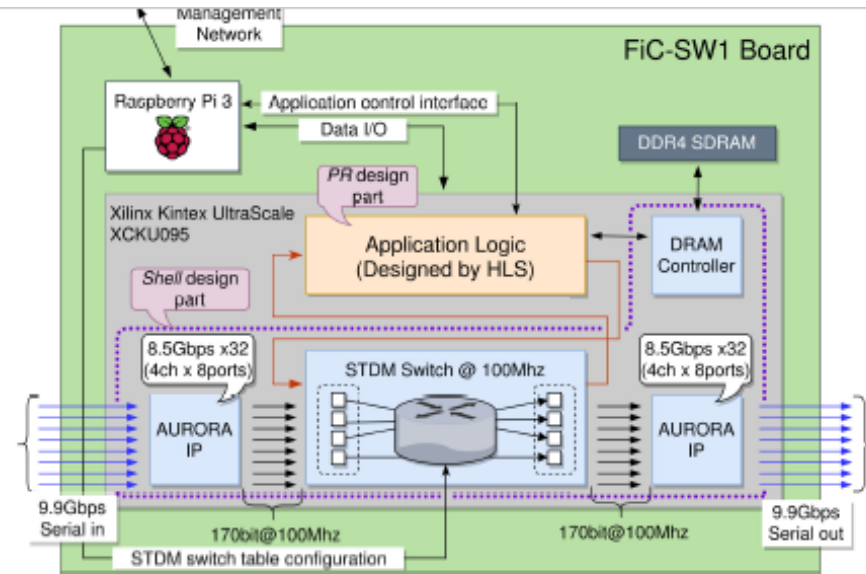
# 方針

---

- 実行性能追及でなく、理解が容易なできるだけ小規模のオープンソースであること
- Linuxを実行できるようにするためページングをサポートし、多重仮想アドレス空間を実現できること
- 主記憶やI/Oデバイスの接続が容易であること
- できるだけ、標準部品で構成できること
- FPGAの使用リソースは応用処理も実現できるくらいの規模であること
- RISC-Vの記述言語は、余計な学習、環境構築を避けるため、VerilogなどのHDLであること。

# FiC FPGAボード(本ボードに基づくM-KUBOS(Zynq)は、Paltek社より販売中)

- Xilinx社XCKU115、ボード接続用のスイッチ、インターコネクティブIP、DRAMコントローラを搭載
- ユーザがHLSまたはHDLで



- 設計したアクセラレータは、部分再構成を使うことで、スイッチ部を変更せずに実装できる。
- DRAMとしては、DDR-4 SDRAM 4GBを2組、8.5Gbpsのシリアルリンクを32組。制御用のRaspberry Pi3はドーターボードの形で搭載され、GPIOを用いてFPGA内のロジックとデータ交換





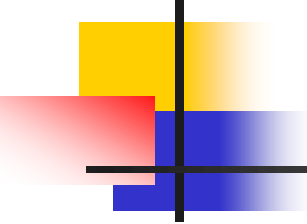
### 3.RV-PC: RISC-Vdays 2021 Springで発表。 「FPGA入門用ボードで手軽に実現するLinux対応 のRISC-Vパーソナルコンピュータ」

- 先の条件を満たすオープンソースのRISC-Vのコードを検討した。
- Linuxを実行できるオープンのRISC-Vの記述言語はVerilog

#### 2) 東工大吉瀬研究室のRV-PC10700行程

度、 Junya Miura, Hiromu Miyazaki, Kenji Kise: A portable and Linux capable RISC-V computer system in Verilog HDL arXiv:2002.03576 [cs.AR] 2020-02.

- Hiromu MIYAZAKI, Takuto KANAMORI, Md Ashraful ISLAM, Kenji KISE, RVCoreP: An Optimized RISC-V Soft Processor of Five-Stage Pipelining, IEICE Transactions on Information and Systems, 2020, Volume E103.D, Issue 12, Pages 2494-2503, 2020.12
- [6] 吉瀬謙二: FPGA入門用ボードで手軽に実現するLinux対応のRISC-Vパーソナルコンピュータ, RISC-V Days Spring, RISC-V協会,



# 3.RV-PC: RISC-Vdays 2021 Springで発表。 「FPGA入門用ボードで手軽に実現するLinux対応 のRISC-Vパーソナルコンピュータ」

- RV-PC: Digilent社Nexsys4 DDR、Arty A7-35Tで稼働
- Linuxカーネルが動作する。
- Nexsys4DDRは、VGA, イーサネット、SDカードコネクタを有し、USBマウスと pmod PS/2を用いてPS2キーボードを接続し、Xwindowを実行できる。Artyについては、SDカードとGPIOイーサネットPHYカードを接続し、シリアルコンソールでLinuxとXwindowを実行できる。
- 遂次版はVerilogで6300行程度、
- パイプライン版は10700行程度
- 1)CVA6: 19000行程度
- このRISC-VをFICボードに移植する。

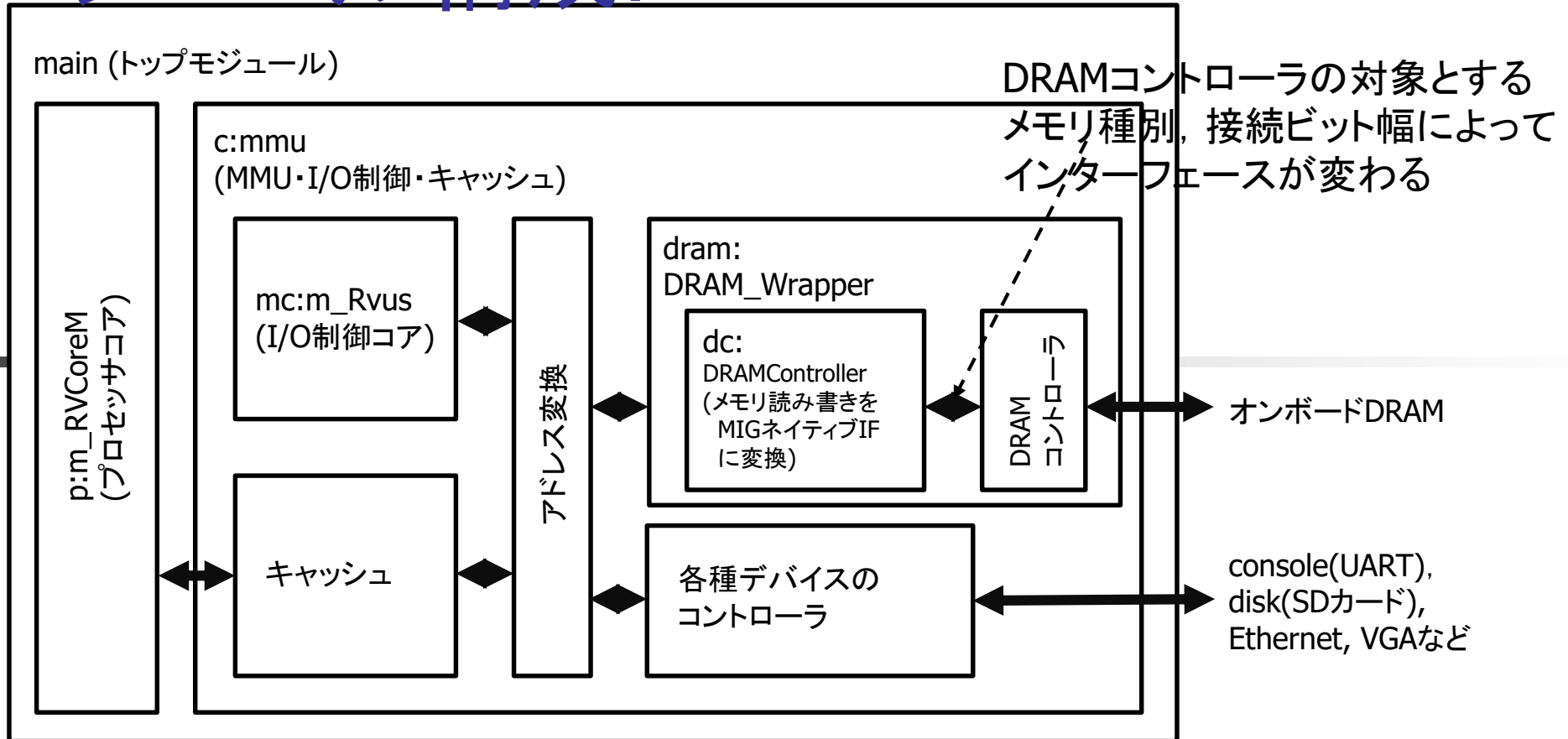


## 4. RV-PCのFiCへの移植

---

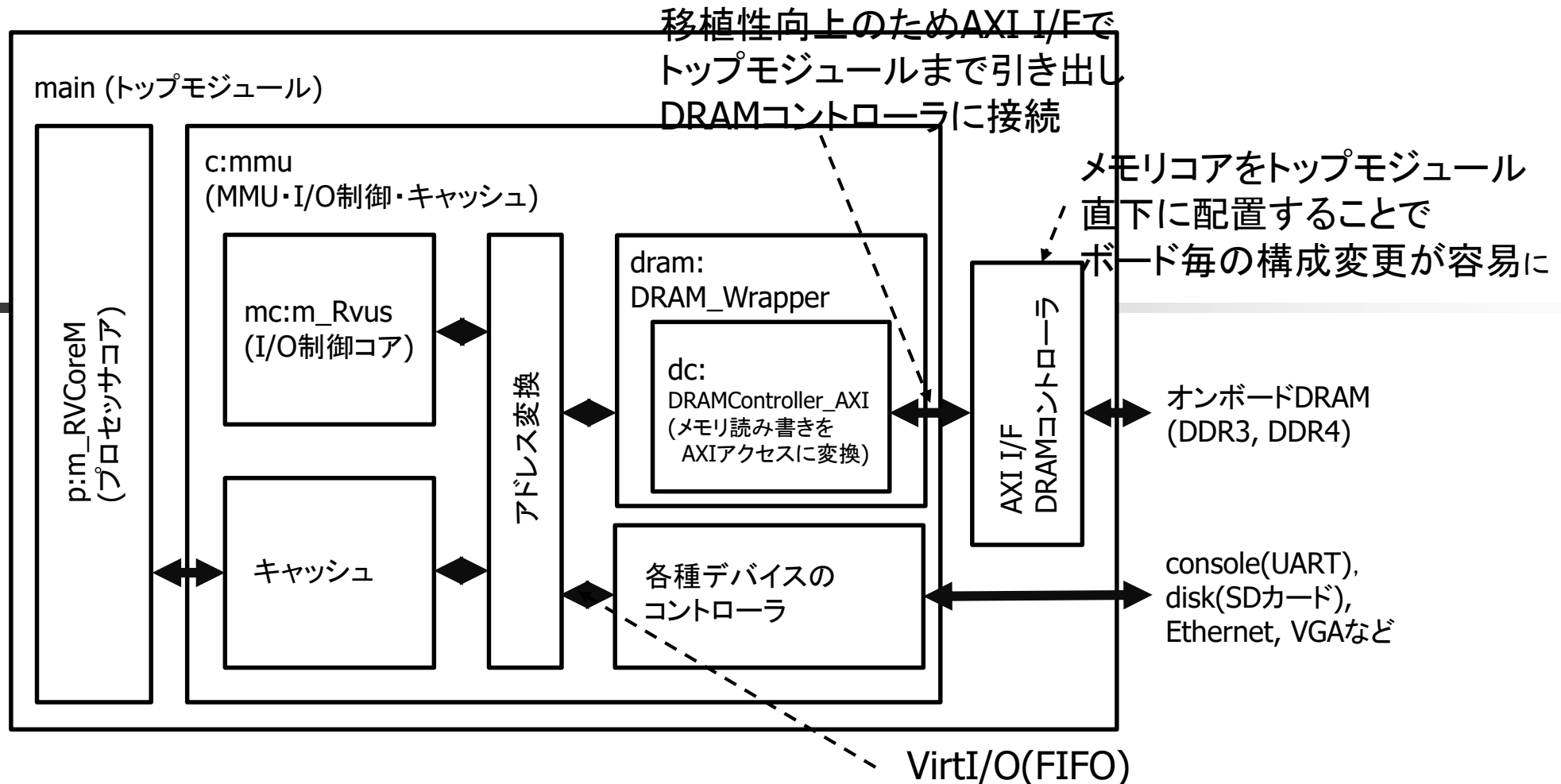
- RV-PCのRISC-VをFiCボードに移植する。
- 移植の際の主な作業は、
  - 1)主記憶のDRAM
  - 2)I/O
- をFiCで動くようにどうするか

# 3. 1オリジナルのRV-PCのモジュール構成.



DRAMコントローラでMIGのネイティブI/Fを採用⇒DRAMの変更  
(ターゲットボードの変更)時の移植コストが大きい  
⇒AXIに修正する

# 3. 2 移植性向上のため、メモリインターフェースをAXI化したRV-PCのモジュール構成



ターゲットメモリによらずAXIでDRAMコントローラと接続できるため  
様々なボードへの移植が容易



# I/O

- I/Oは VirtI/O(FIFO)でアクセス
- I/Oは最初はそのまま使用
- 安定したらいずれAXIに

デバイス	VirtI/O アドレス
console	0x40000000
disk	0x41000000
ether	0x42000000
keyboard	0x43000000
mouse	0x44000000
frame buffer	0x45000000
OPLIC	0x50000000
CLINT	0x60000000
DRAM	0x80000000

OS屋的には、FPGAでVirtI/Oで単純なデバイスモデルに仮想化されたようなもの→良い基本設計だった



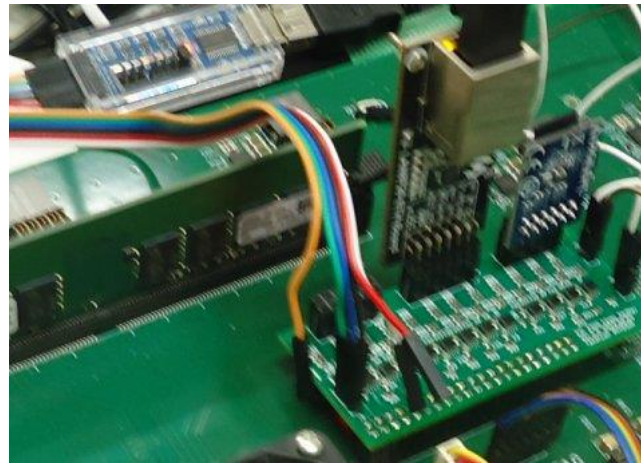
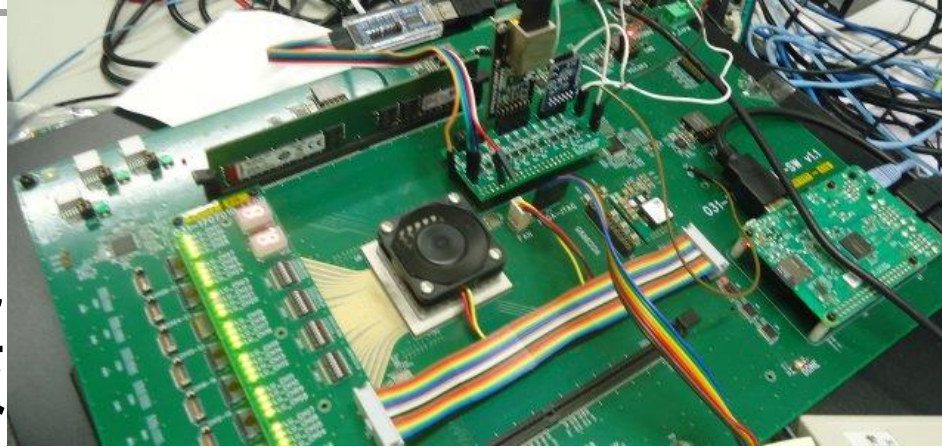
# 移植手順

---

- いきなりFiCはデバッグが大変かも
- まず、ArtyA7-100T、Genesys2でデバッグ
- それからFiC
- 修正量減らしたい、できるだけ、同じに
- Pmodで
- UART、SD、カード、
- Ether PHY

# 修正したFiC

- 確認後
- FiCの外部I/Oに電圧レベル変換基板作成して
- Pmod $\mu$ SD、
- Pmod Ether PHY
- 電圧合致のUSB-UART



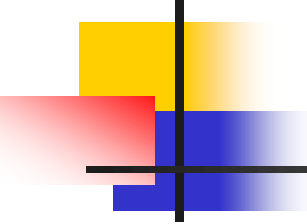


# 次のFPGAボードで動作している

項番	board		device	DRAM	I/O	
1	FiC		xcku095ffvb2104-1	DDR4 4GB	ether,uart,sd	pmod,gpio外付け
2	ExstickGE	eTrees	xc7a200tsbg484-2	DDR3 256MB	uart,sd	pmod,gpio外付け
3	ArtyA7-100T	Digilent	xc7a100tcsg324-1	DDR3 256MB	ether,uart,sd	pmod,gpio外付け
5	Nexsys4DDR	Digilent	xc7a100tcsg324-1	DDR2 128MB	ether,uart,sd, vga,kb,mouse	
6	genesys2	Digilent	xc7k325tffg900-2	DDR3 256MB	ether,uart,sd	pmod,gpio外付け

- PCIe接続のKCU1500はデバッグ中



- 
- なお、AXI化したコードなどは、それぞれ次のgitubから利用できる。
  - 1)起動イメージをUARTからアップロードする版(表2,3の項番4のRISC-V、RVSoC):
    - [https://github.com/miyo/rvsoc\\_axi](https://github.com/miyo/rvsoc_axi)
  - 2)SDカードからLinuxブート, Ethernetが利用できるLinux版(表2, 3の項番3のRISC-V、RV-PC):
    - <https://github.com/miyo/rv-pc-axi>



## 5. 性能評価

---

- 性能チューニング、不要部の削除などは未了
- 実行性能をDhrystoneとリソース使用量を検証

# Dhrystoneによる実行性能評価

- Sifive、Maixbit、Zynq(PynqZ1)の値も参考まで
- CVA6は論文[1]から
- FPGA部の制御には十分使える

項番	board		core clock(MHz)	device	DRAM	I/O	DMIPS/MHz	DMIPS	Dhrystone/s
1	FiC		75	xc7a095ffvb2104-1	DDR4 4GB	ether,uart,sd	0.28	21.32	37453
2	ExstickGE		75	xc7a200tsbg484-2	DDR3 256MB	uart,sd	0.28	21.32	37453
3	ArtyA7-100T (5段パイプライン)		75	xc7a100tcsg324-1	DDR3 256MB	ether,uart,sd, vga,kb,mouse	0.29	21.56	37878
4	Arty(逐次)		100	xc7a100tcsg324-1	DDR3 256MB	uart	0.04	4.33	7616
5	Nexsys4DDR		75	xc7a100tcsg324-1	DDR2 128MB	ether,uart,sd, vga,kb,mouse			
6	genesys2		75	xc7k325tffg900-2	DDR3 256MB	ether,uart,sd	0.28	20.85	36630
7	CVA6[9]			VCU118			1.21		
8	Sifive	ASIC	1200				1.84	2205.17	3874484
9	MaixBit/k210	ASIC	400				0.70	279.64	491328
10	PynqZ1	ASIC:hardcore	650				1.12	725.03	1273885

- .
- [1] .Dorflinger , M.Albers , B.Kleinbeck , Y.Guan , H.Michalik
- A comparative survey of opensource application-class RISC-V
- processor implementations, CF '21: Proceedings of the 18th ACM International Conference on Computing Frontiers, pp.12, 20 2021, 2

# リソース使用量

- チューニングはこれから
- 応用処理のリソースはある

項番	board	device	DRAM	I/O	CLB LUTs	CLB Regs	BRAM	DSP
1	FiC	xcku095ffvb2104-1	DDR4 4GB	ether,uart,sd	6.10(32160/537600)	2.82(30342/107520)	3.72(62.5/1680)	2.34(18/768)
					Slice LUTs	Slice Regs		
2	ExstickGE	xc7a200tsbg484-2	DDR3 256MB	uart,sd	10.96(14752/134600)	3.49(9406/269200)	11.51(42/365)	2.03(15/740)
3	ArtyA7-100T (5段パイプライン)	xc7a100tcsg324-1	DDR3 256MB	ether,uart,sd, vga,kb,mouse	24.07(15262/63400)	7.75(9825/126800)	97.78(132/135)	6.67(16/240)
4	Arty(逐次)	xc7a100tcsg324-1	DDR3 256MB	uart	11.43(7249/63400)	2.79(3535/126800)	28.15(38/135)	6.25(15/240)
5	Nexsys4DDR	xc7a100tcsg324-1	DDR2 128MB	ether,uart,sd, vga,kb,mouse	18.60(11790/63400)	4.87(6179/12800)	98.52(133/135)	6.67(16/240)
6	genesys2	xc7k325tffg900-2	DDR3 256MB	ether,uart,sd	8.92(18179/203800)	3.12(12707/407600)	9.44(42/445)	1.79(15/840)



## 6.結論

---

- RV-PCの移植性向上を行い、FiCボードに移植し、FiCボードでLinuxが稼働、
- マルチFPGA、OSの研究基盤を構築できた。

## 7. 今後の課題

現在、次の内容を検討している。

- CRESTプロジェクトでのRISC-Vの実チップ化
- PCIeに刺さるボードへの移植  
←手持ちのKCU1500でデバッグ中
- MECの高速リンクの導入
- USB,HDMIなどI/Oの拡充
- ブロックデザイン化 頑張っても論文にならん...
- HypervisorModeの導入←RISC-V用VMの研究





# 今後

---

- チップ化の検討
- PCIeのFPGAボードに
- 手ごろな組込み用RISC-Vを
- Xilinxに乗せて
- Toppersか
- No MMU Linux乗せてもいいかも
- ←これなら能力イマイチの学生でもできるかも
- Picorv32をBasys3に乗せようとして悪戦デバッグ中
- Vivadoなど開発環境に慣れるの大変