

# SimMips/MieruPC

～システムシミュレータから  
計算機システムへ～

藤枝 直輝 (東京工業大学)

2010/08/05

SWoPP2010 BOFセッション

「シンプルハードウェアがもたらす  
計算機システム研究／教育の新展開」



# 発表者について

- ◆ 東京工業大学 吉瀬研究室 D1
- ◆ 2008年4月～ 「MieruPCプロジェクト: 中身がみえる計算機システムを構築する研究・教育プロジェクト」の中心メンバーとして開発に参加
- ◆ 2009年2月～ MieruPC株式会社 代表取締役

MieruPCプロジェクト: 中身が見える計算機システムを構築する研究・教育プロジェクト

ニュース

- Embedded Technology 2009にてMieruPC-2008のデモをおこないます。(2009-07-17)
- MieruPC-2008はMieruPC株式会社または 東京エレクトロデバイス株式会社から購入できます。(2009-07-17)
- ホームページを公開しました。(2009-07-17)

プロジェクトの目的

Computer Scienceを勉強しているのですが計算機システムを製作することはできますか？ できます！

- ハードとソフトのすべてが見えるスタンダードで動作する計算機システムを作る！
- FPGAあるいはVDECのチップなどを利用してCPUの内部も見える・修正できる計算機システムを構築する。
- 大学2,3年生が理解できる洗練された計算機システムの実現を目指す。
- 5万円以内のテキスト形式で洗練された計算機システムを実装する。
- 必要最小限のシンプルなハードウェアで実現する。低価格で購入できる計算機システムとする。
- シンプルで洗練されたソフトウェア群を提供する。
- テキスト版テトリスが動く計算機システムを実現する。
- エディタでテキストファイルの編集ができる計算機システムを実現する。
- 新規性がない方が好ましい。開発は基本的に既存のものを参考にしながらゼロから構築する。
- 成果はフリーとしてどんどん公開する。参加したい人は歓迎する。どんどん担当してもらおう。
- 貢献した人を讃える。論文をたくさん執筆する。ハードウェアを販売する。あまり厚くない本を出版する。
- 日本の情報工学を活性化する。日本発のMieruPCを世界に広める。

MieruPC-2008/MieruPC-2010 説明資料

- MieruPC-2008(サンプルキットを含む)の使い方 Ver.2009-02-24【PDF】
- MieruPC-2008)ハードウェアのセットアップと動作確認等【PDF】
- MIPS32のクロス開発環境の構築 ~ MieruPCのアプリケーション開発準備 ~【PDF】

<http://www.arch.cs.titech.ac.jp/mieru/>

MieruPC株式会社

製品紹介 利用方法 ダウンロード サポート 会社概要

MieruPC

What's New

- 2010/07/07 MieruPC2010に関する情報を追加しました。
- 2010/07/07 ITmedia +D PC USERで MieruPCが紹介されました。
- 2010/03/04 JTAGケーブルの価格改定を実行しました。
- 2010/02/16 JTAGケーブルの価格改定について追加しました。
- 2010/02/04 FPGA Card I/Oボードの技術情報を追加しました。

全ての履歴

このサイトは、教育・研究のためのシンプルで扱いやすい計算機システム MieruPC、およびMieruPCの製造・販売をおこなう MieruPC株式会社に関するサイトです。

Copyright (C) 2009-2010 MieruPC Inc. All Rights Reserved.

<http://www.mierupc.com/>

# MieruPC Project

◆ Computer Scienceを勉強しているのですが、  
 計算機システムを製作することは出来ますか？  
 → **できます！**

◆ 計算機システムを包括的に  
 学ぶためのプラットフォーム  
 として、中身がみえる計算機  
 システムMieruPCを開発中

2009年8月 SWEST11のポスター →

中身がみえる計算機システムMieruPC  
 ~ 2008 - 2009 ~  
 藤枝 直輝, 渡邊 伸平, 高前田 伸也, 佐野 正浩, 佐野 伸太郎, 吉瀬 謙二  
 東京工業大学 大学院情報理工学専攻

**みえる** - MieruPCプロジェクトの目的 -  
 Q. Computer Scienceを勉強しているのですが、計算機システムを製作することは出来ますか？  
 A. **できます！**

**つくる** - MieruPC-2008の開発 -  
 MieruPC-2008の全体像  
 Verilog HDLによるハードウェア記述

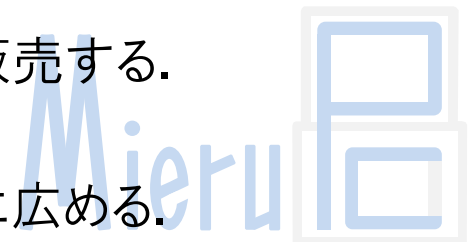
**そだてる** - MieruPC-2009にむけて -  
 MieruPC-2009の独自開発  
 MieruPC上で動作するOSの開発

**ひろめる** - 研究成果の公開・普及 -  
 ●大学・大学院の講義や高専生向けの講座での利用  
 ●ハードウェアの製造・販売のための会社設立

MieruPC Project  
<http://www.arch.ces.titech.ac.jp/mieru/>  
 2009/8/27-28 SWEST11 ポスター/デモセッション

# プロジェクトの目的

- ◆ ハードとソフトのすべてが見えるスタンドアロンで動作する計算機システムを作る！
- ◆ FPGA(あるいはVDECのチップなど)を活用してCPUの内部も見える・修正できる計算機システムを構築する.
- ◆ 大学2,3年生が理解できる洗練された計算機システムの実現を目指す.
- ◆ 5万行以内のテキスト形式で洗練された計算機システムを実装する.
- ◆ 必要最小限の**シンプル**なハードウェアで実現し, 安価な計算機システムとする.
- ◆ シンプルで洗練されたソフトウェア群を提供する.
- ◆ テキスト版テトリスが動く計算機システムを実現する.
- ◆ エディタでテキストファイルの編集ができる計算機システムを実現する.
- ◆ 新規性がない方が好ましい.  
開発は基本的に既存のものを参考にしながらゼロから構築する.
- ◆ 成果はフリーとしてどんどん公開する.  
参加したい人は歓迎する. どんどん担当してもらう.
- ◆ 貢献した人を讃える. 論文をたくさん執筆する. ハードウェアを販売する.  
あまり厚くない本を出版する.
- ◆ 日本の情報工学を活発化する. 日本発のMieruPCを世界に広める.



# それぞれご紹介

## ◆ SimMips

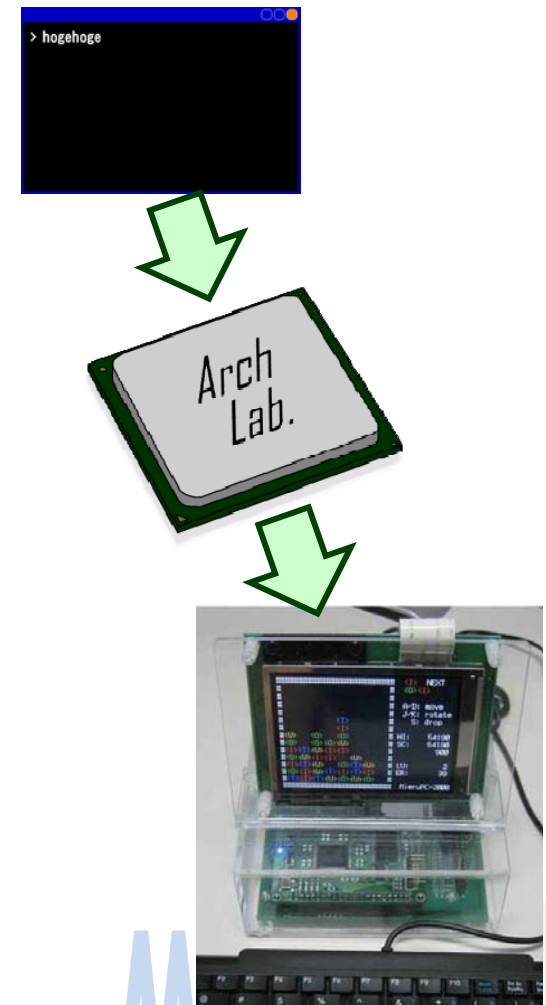
- MIPS システムシミュレータ

## ◆ MipsCore

- MIPSライク ソフトプロセッサ (Verilog-HDL)
- SimMipsがベース

## ◆ MieruPC

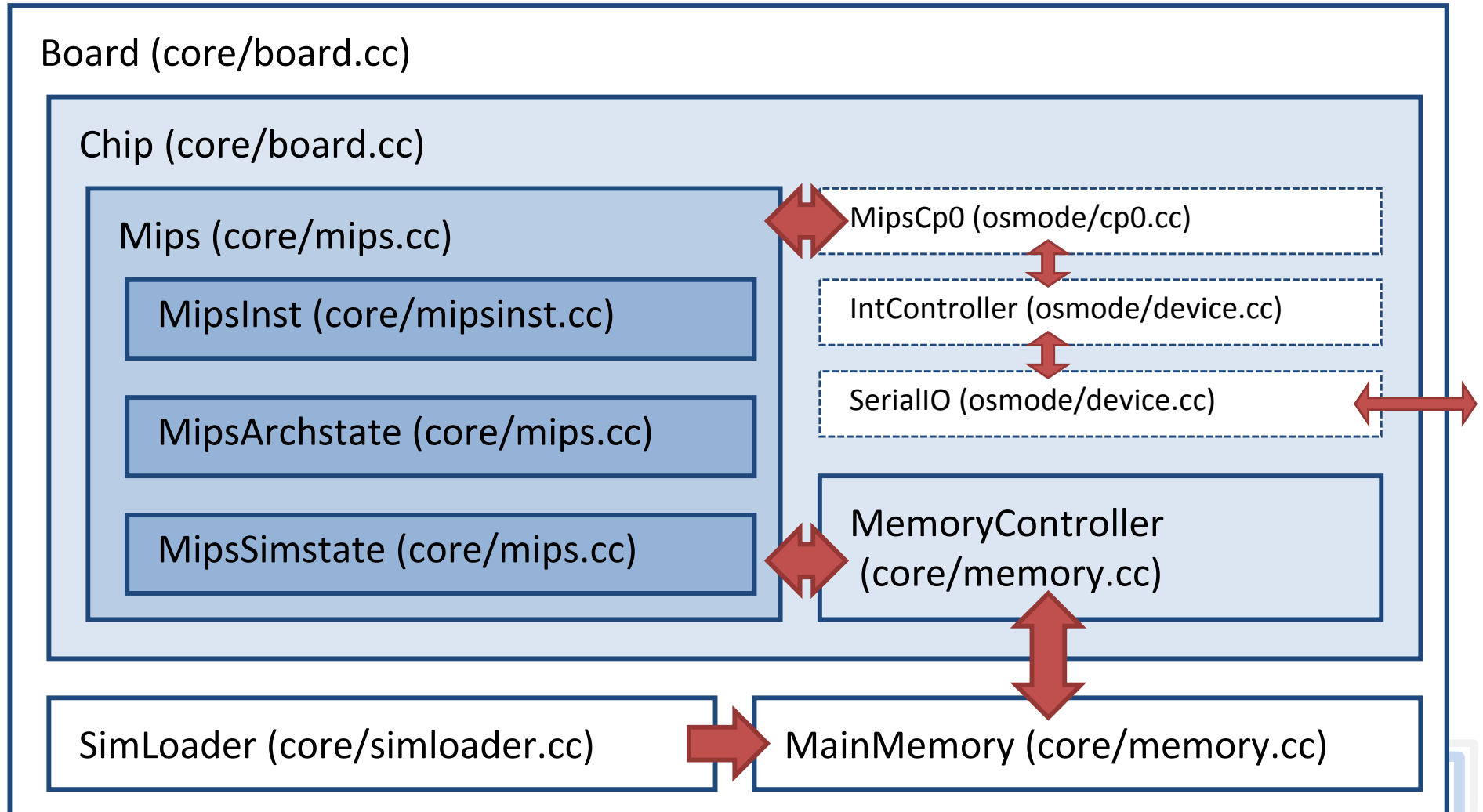
- MIPSライク 計算機システム
- MipsCoreをプロセッサコアとして使用



# SimMipsの概要

- ◆ 命令セット: MIPS32 R2のサブセット
  - 実装命令数は約100種  
(浮動小数点命令を除くほぼ全て)
- ◆ コード量: 7,519行(v0.6.6)
  - うち, コア機能が3,110行
  - コメント, 空行を含む
- ◆ 主要部分(~v0.4.0)の開発期間: 約2ヶ月半
  - 2008年4月~7月
  - 開発は学生2名

# SimMipsの構成

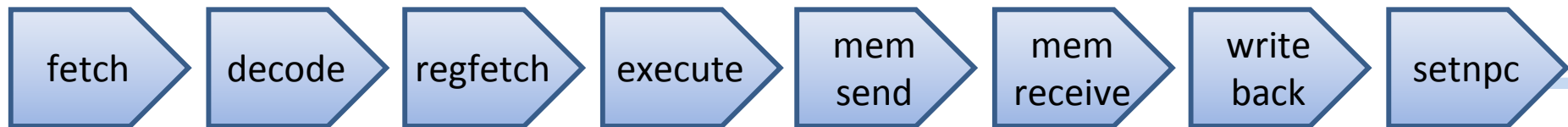


- ◆ 命令の実行を複数のステージに分ける
  - パイプライン構造のように
  - それぞれのステージは1つのメソッドで実行
  - 可読性向上

Typical 5-stage Pipeline



SimMips (8 stages)



# それぞれご紹介

## ◆ SimMips

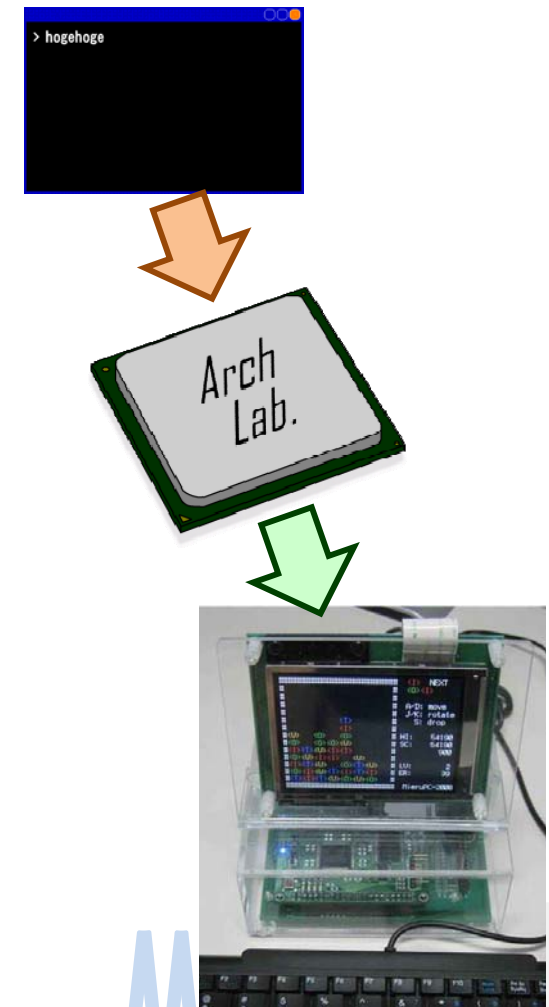
- MIPS システムシミュレータ

## ◆ MipsCore

- MIPSライク ソフトプロセッサ (Verilog-HDL)
- SimMipsがベース

## ◆ MieruPC

- MIPSライク 計算機システム
- MipsCoreをプロセッサコアとして使用

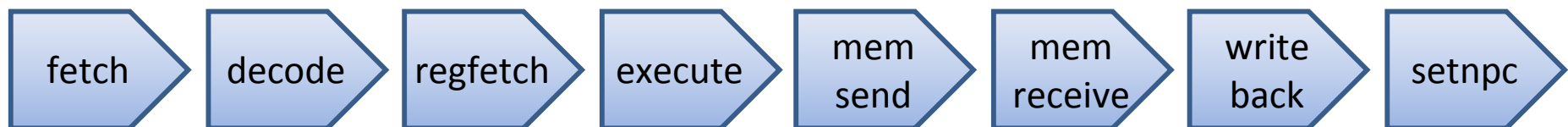


MieruPC

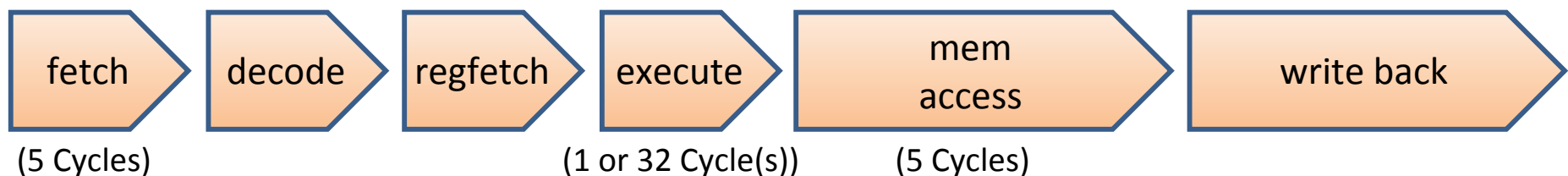
# MipsCoreの概要

- ◆ MIPSライク(互換ではない)のソフトプロセッサ
- ◆ SimMipsを部分的に移植
  - ハードウェアを意識した構成の利点
- ◆ マルチサイクルの実行モデル
  - 9~40 サイクル/命令

## SimMips



## MipsCore



# Implementation of MipsCore

## ◆ 命令実装部分の一例

- 半分以上のコードはこのように機械的に移植

### SimMips

```
1 void MipsInst::decode()
2 {
3     opcode = (ir >> 26) & 0x3f;
4     rs = (ir >> 21) & 0x1f;
5     ...
6     funct = ir & 0x3f;
7     ...
8
9     switch (opcode) {
10    case 0:
11        switch (funct) {
12            ...
13            case 33:
14                op = ADDU_____;
15                attr = READ_RS | READ_RT |
16                    WRITE_RD;
17            break;
18            ...
19        }
20    }
21 }
```

### MipsCore

```
1 /* MipsInst::decode() */
2 always@ ( DATA_IN ) begin
3     IDOPCODE = DATA_IN[31:26];
4     IDRS      = DATA_IN[25:21];
5     ...
6     IDFUNCT   = DATA_IN[ 5: 0];
7     ...
8
9     case ( IDOPCODE )
10    6'd0: begin
11        case ( IDFUNCT )
12            ...
13            6'd33: begin
14                IDOP = 'ADDU_____;
15                IDATTR = 'READ_RS | 'READ_RT |
16                    'WRITE_RD;
17            end
18        end
19    end
20 end
21 ...
```

# それぞれご紹介

## ◆ SimMips

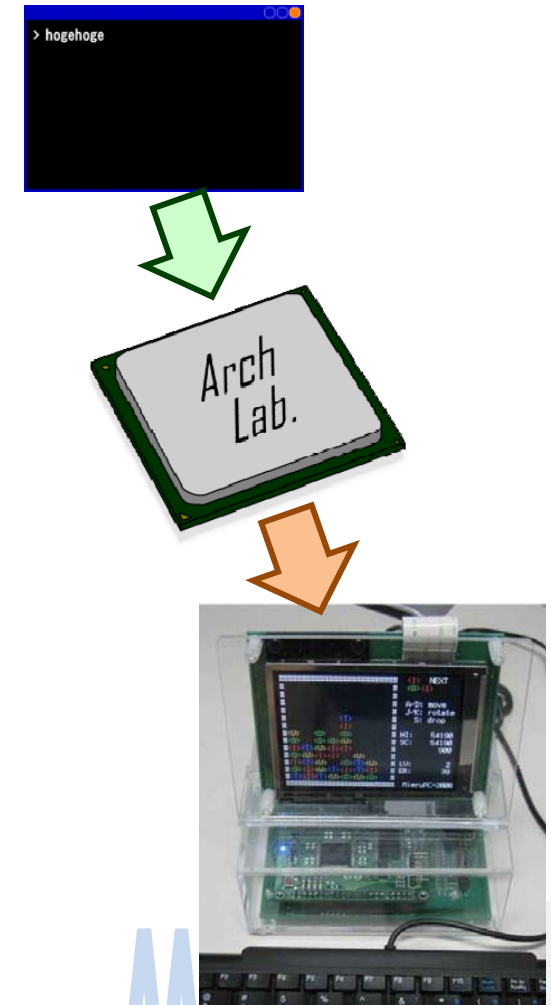
- MIPS システムシミュレータ

## ◆ MipsCore

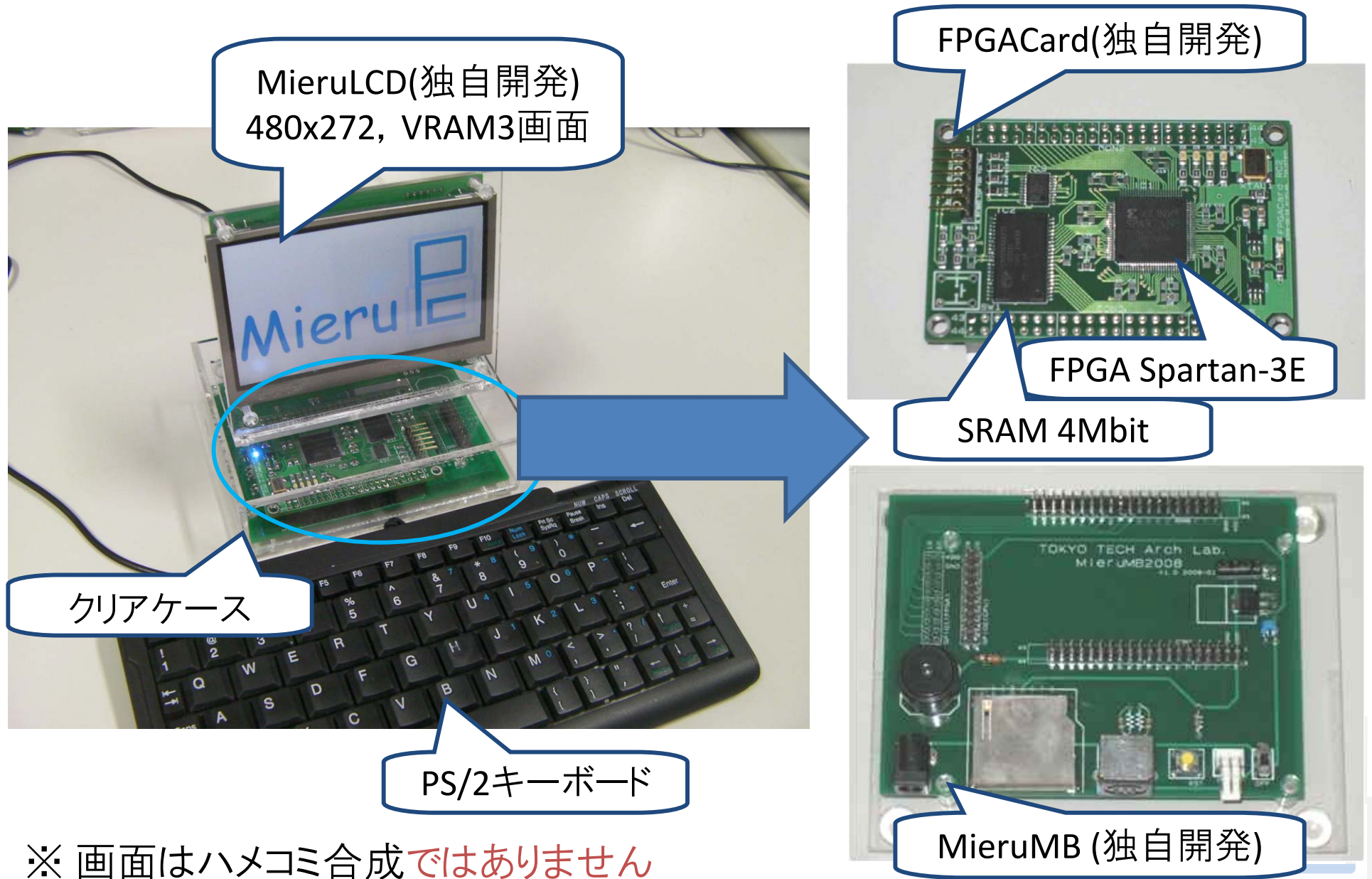
- MIPSライク ソフトプロセッサ (Verilog-HDL)
- SimMipsがベース

## ◆ MieruPC

- MIPSライク 計算機システム
- MipsCoreをプロセッサコアとして使用



# MieruPC-2010 ハードウェア



# MieruPC-2010 HDL

## ◆ FPGACardのFPGA

### ■ MipsCore

- キーボード, 液晶, マルチメディアカードのコントローラ
- 初期RAMイメージローダ

## ◆ MieruLCDのFPGA

- 液晶コントローラ
- キャラクターROM

## ◆ 全てVerilog HDL, フルスクラッチで記述



# MieruPC-2010 OS

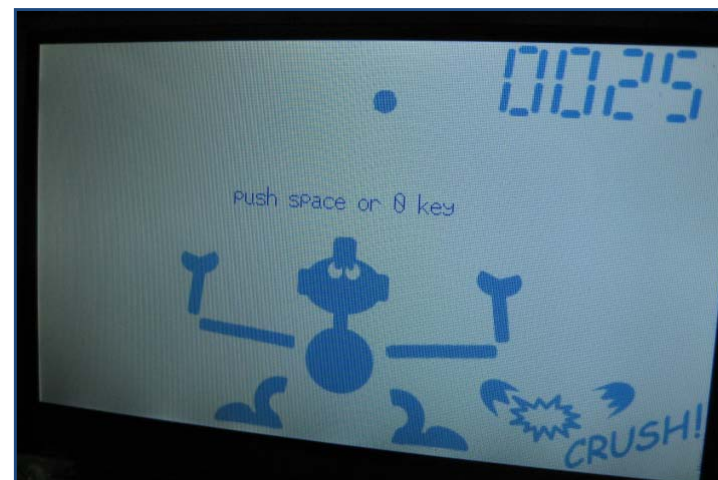
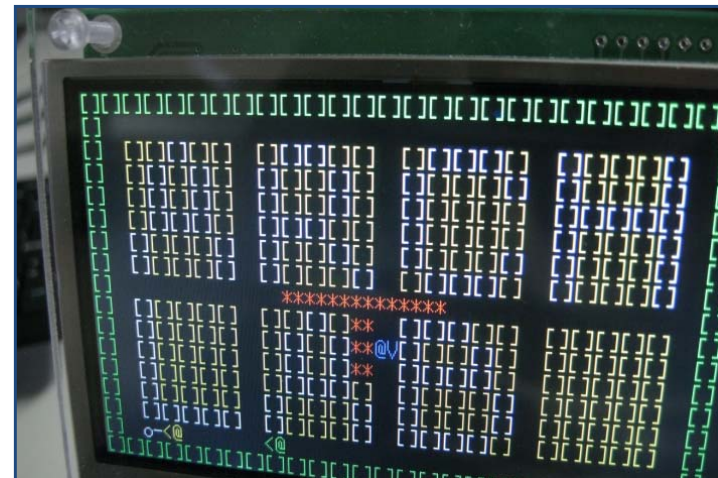
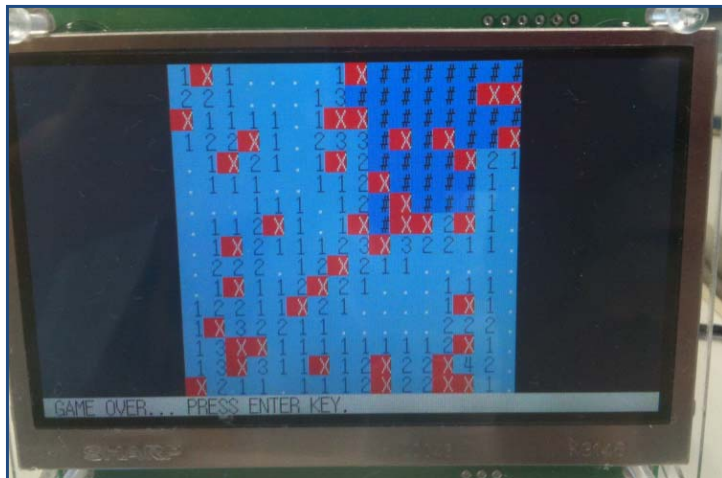
## ◆ 独自開発のOSが動作

- FATファイルシステムの操作
- ユーザプログラムのロード, 起動(シングルタスク)
- システムコール など...



# MieruPC-2010 ソフトウェア

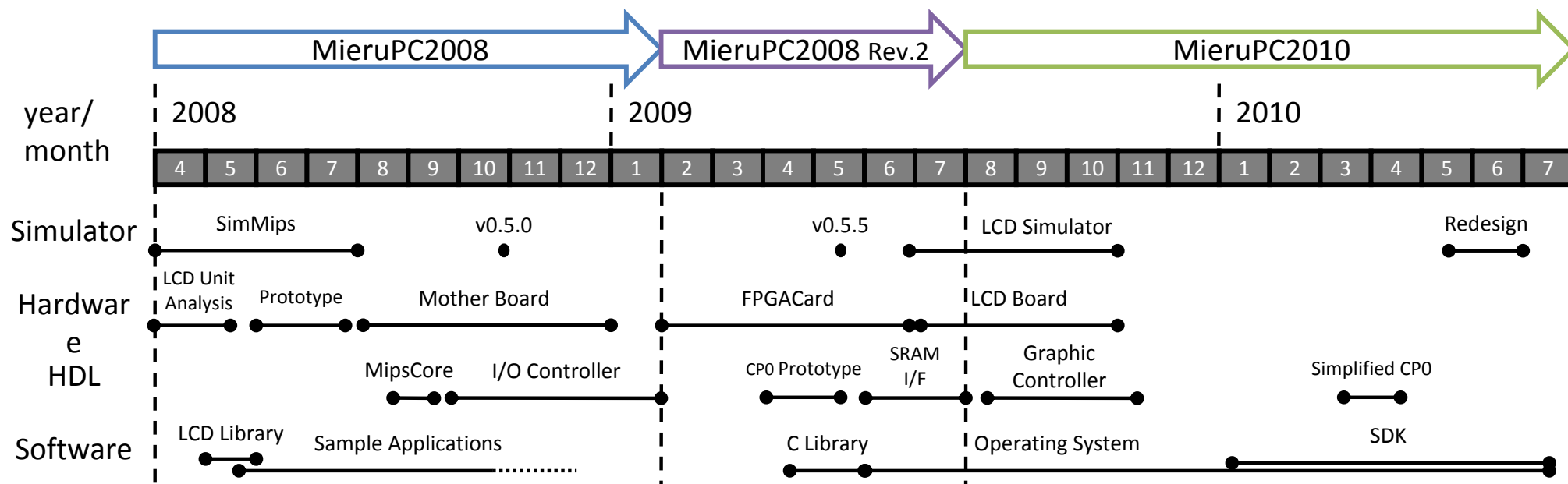
- ◆ いろいろ動いています
  - SDKで効率的に開発



## ◆ SimMipsとMieruPC

- SimMipsについては動画で
- MieruPCについては実機で

# つづいて時系列順に



- ◆ 利用するハードウェアの検討
  - FPGAにアットマークテクノ SUZAKU-Sを
  - 液晶にインテグラル電子 ITC-2432-035Hを
  
- ◆ 可能性検証
  - SUZAKU-Sに元々乗っているMicroBlaze+uCLinux上に
  - 液晶のシリアルを操作するデバイスドライバを書いて
  - テキストの時計アプリを動かす
  
- ◆ SimMipsの開発

- ◆ MipsCoreとPS/2・液晶のI/Oの開発
  - メインメモリはFPGA内のBRAMを使用
- ◆ マザーボードの試作品の制作
- ◆ 動作させるアプリケーションの内容は  
コンフィグレーションファイルに直書き

# この頃の全体図

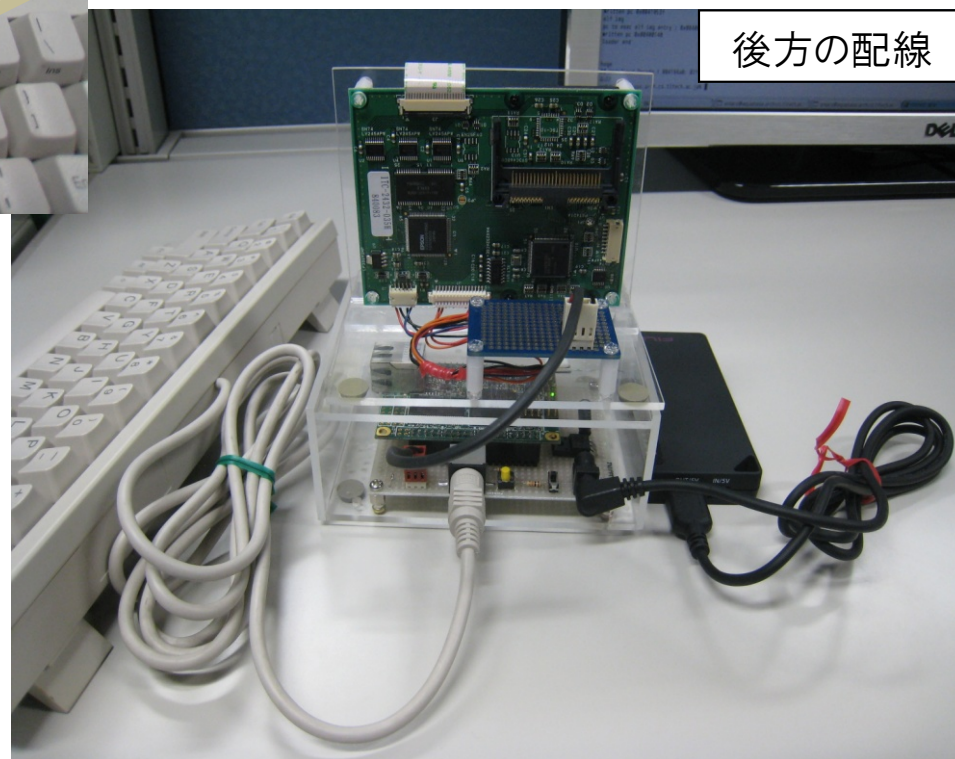
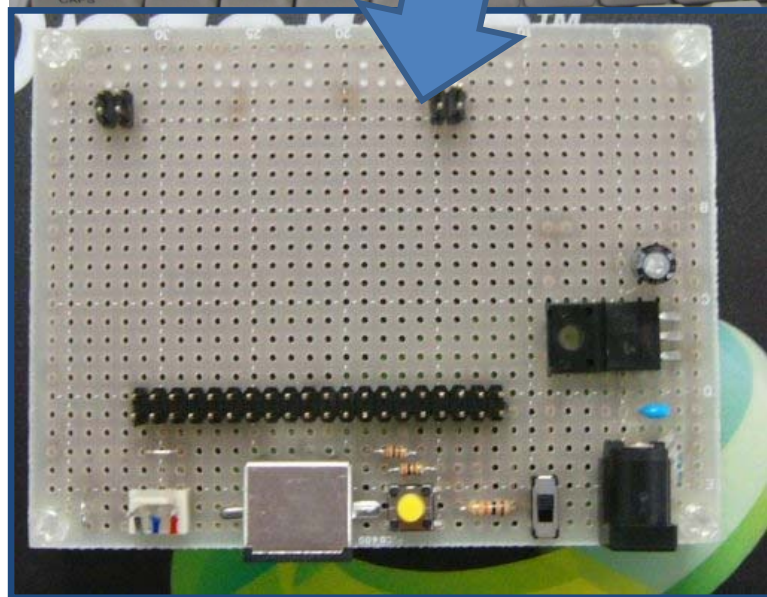
前から

コマンド  
インタプリタ液晶  
ITC-2432-035H

キーボード (PS2)  
Happy Hacking  
Keyboard Light 2 etc...

SUZAKU-S  
マザーボード

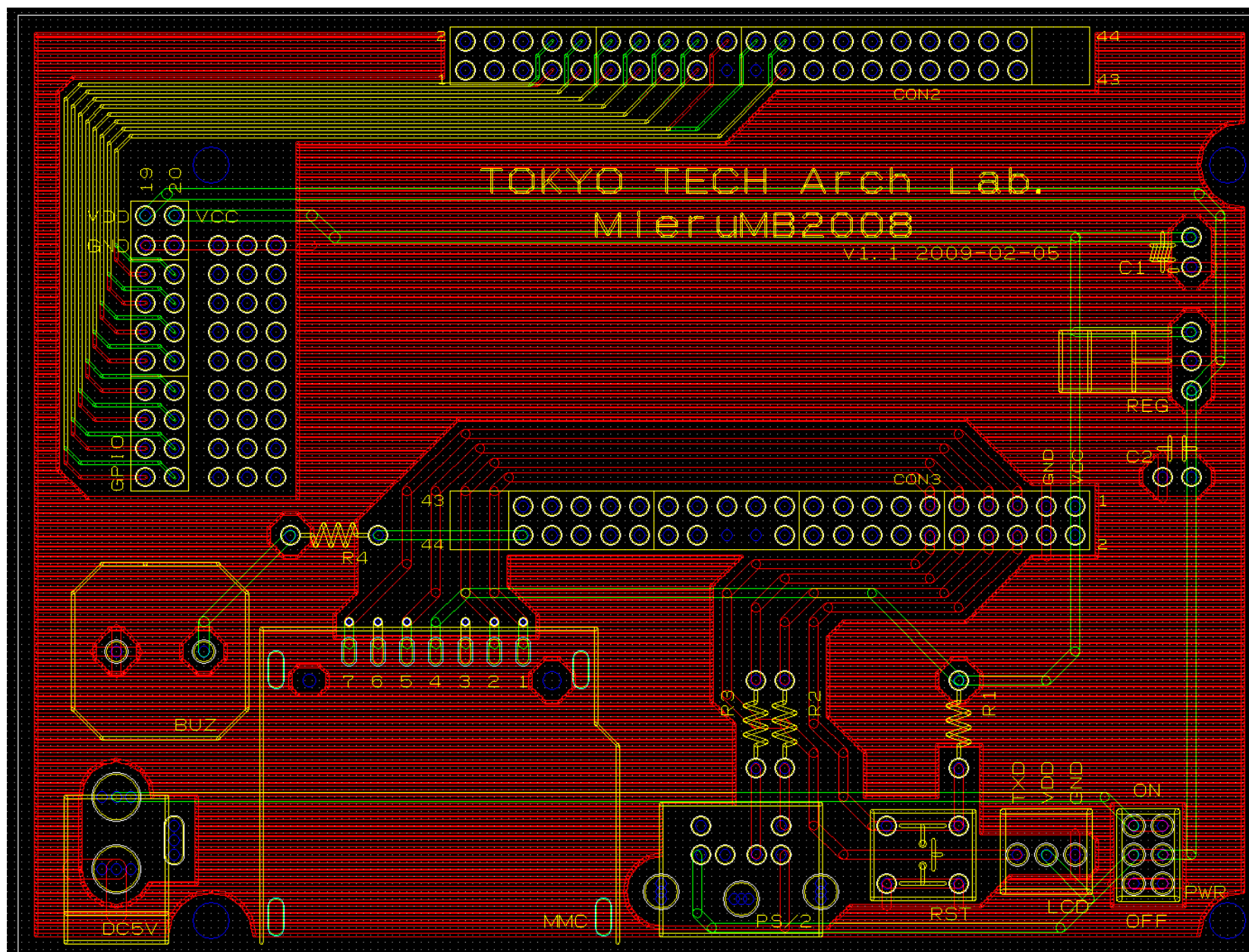
後方の配線



# MMC対応(08/11-09/1)

- ◆ マルチメディアカード(MMC)からのローダ開発
  - 動作させるアプリケーションはMMCからロード
- ◆ どこからロードするか？
  - FATをちゃんと読む？ 特定アドレス決め打ち？
  - 決め打ちにした
    - ◆ フォーマットして最初にファイルを置くと書き込まれるだろう位置
    - ◆ 環境依存
- ◆ マザーボードの制作
  - PCBEで作成, P板.comで製造
- ◆ ここまでがMieruPC-2008

# マザーボードの基板データ



# FPGACardの制作(09/2-09/7)

- ◆ マザーボードの制作ノウハウをもとに
- ◆ I/Oピンの本数はギリギリ
  - FPGAを小容量・実装しやすいパッケージに
  - SRAM接続だけで利用可能数の半分近い
- ◆ 完成品までに5回の試作
  - 1～4回も全て動作したが、軽微な問題あり
- ◆ ここまでがMieruPC-2008 Rev.2

## ◆ 液晶ユニットMieruLCD

- インテグラル電子との共同開発品がベース
- FPGAやSRAMといった構成はFPGACardと同じ
- 安価なゲーム機用の液晶を使用

## ◆ 独自OS(MieruOS)

- 東大・東工大のプロジェクト実習の成果がベース
- MipsCoreにもそれに合わせた機能追加
  - ◆ 制御コアの簡略版
  - ◆ MMCコントローラをロードだけでなく読み書き可能に



## 問題修正・まとめ(10/5-10/7)

- ◆使っていて見つけたバグを修正したり
- ◆SimMipsを再構成したり
- ◆各々リファクタリングしたり
- ◆開発環境をMieruSDKとしてまとめたり
  
- ◆ここまでがMieruPC-2010
  - ハードウェアは7/26より販売中
  - MieruSDKその他は近日公開予定



# まとめ

- ◆ SimMips, MipsCore, MieruPCの紹介
- ◆ シンプルなハードウェアによるシステム構築
- ◆ 低コスト・短期間開発のポイント
  - はじめは既製品ベース
  - ノウハウを貯えて独自構築
  - それまで作ってきたものの再利用
    - ◆ SimMips → MipsCore
    - ◆ FPGACard → MieruLCD