

# EPICS内蔵小型デバイスサーバの開発

帯名 崇 / KEK

路川 徹也 / (株)東日本技術研究所

# 本日の内容

1. 開発の動機
2. ハードウェア候補と選定
3. ハードウェア設計・製作
4. ソフトウェア
5. デバイスサーバとしての利用実績
6. その他
7. まとめと情報公開について

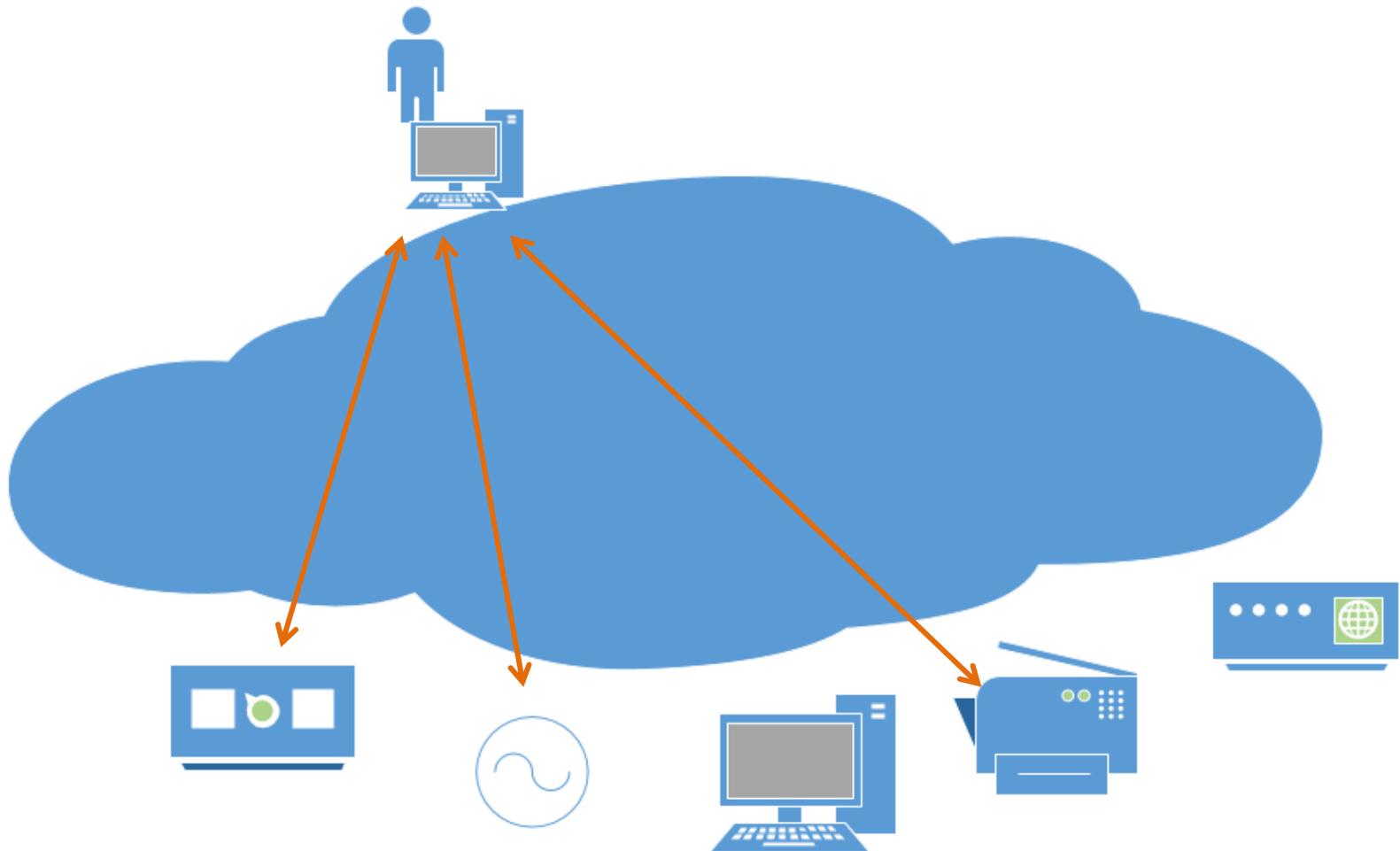
# 1. 動機

- 加速器制御の一般論として
  - 遠隔地から多数の機器を制御することが必要：ネットワーク経由



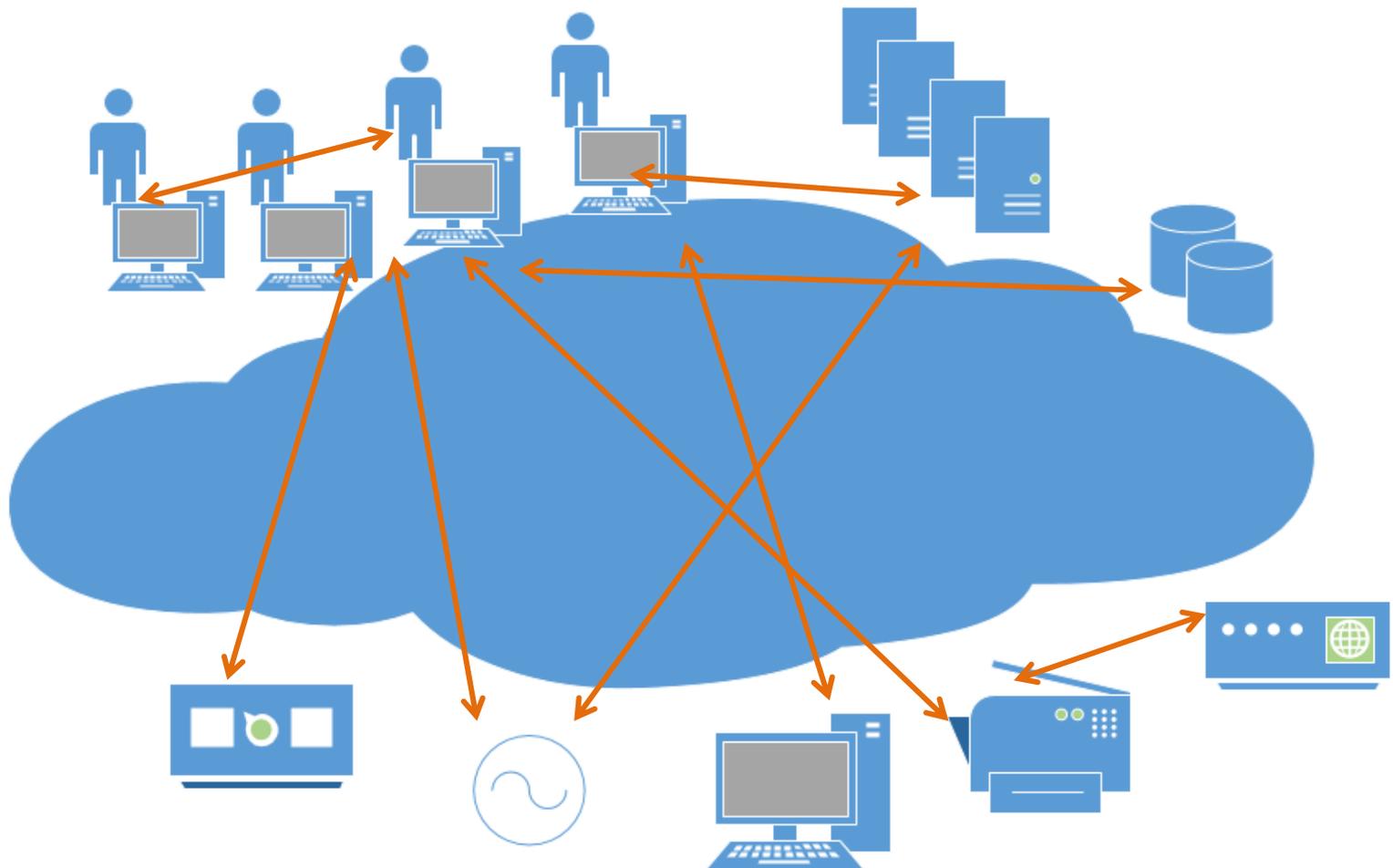
# 制御プログラム

開発者が少ない or 対象機器が少ないならば、単純な通信と制御のプログラムで十分。しかし加速器では**事実上不可能**



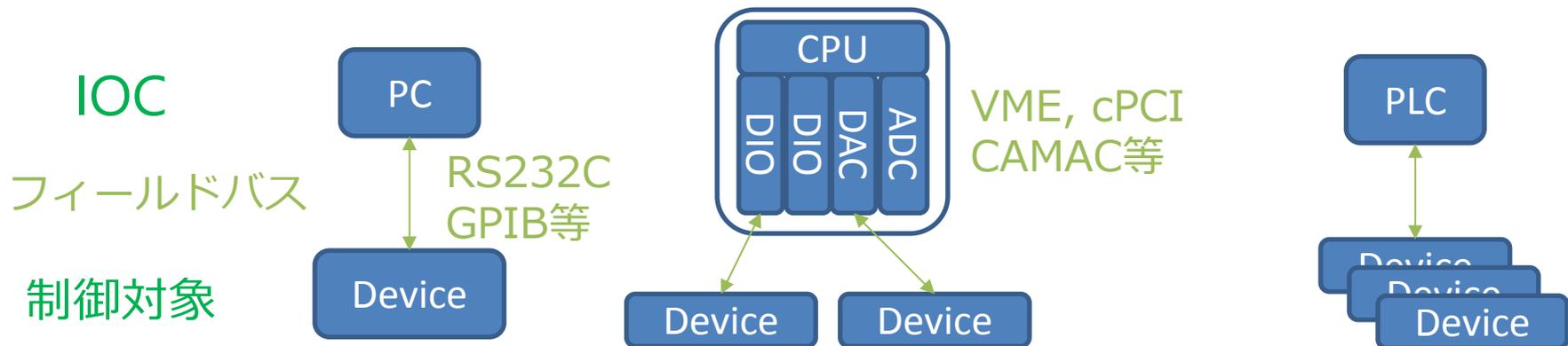
# 制御フレームワーク

複数の開発者・プログラム・装置が有機的に連携するには  
「制御フレームワーク」が必須: EPICS, MADOCA, TANGO等



# EPICSの場合

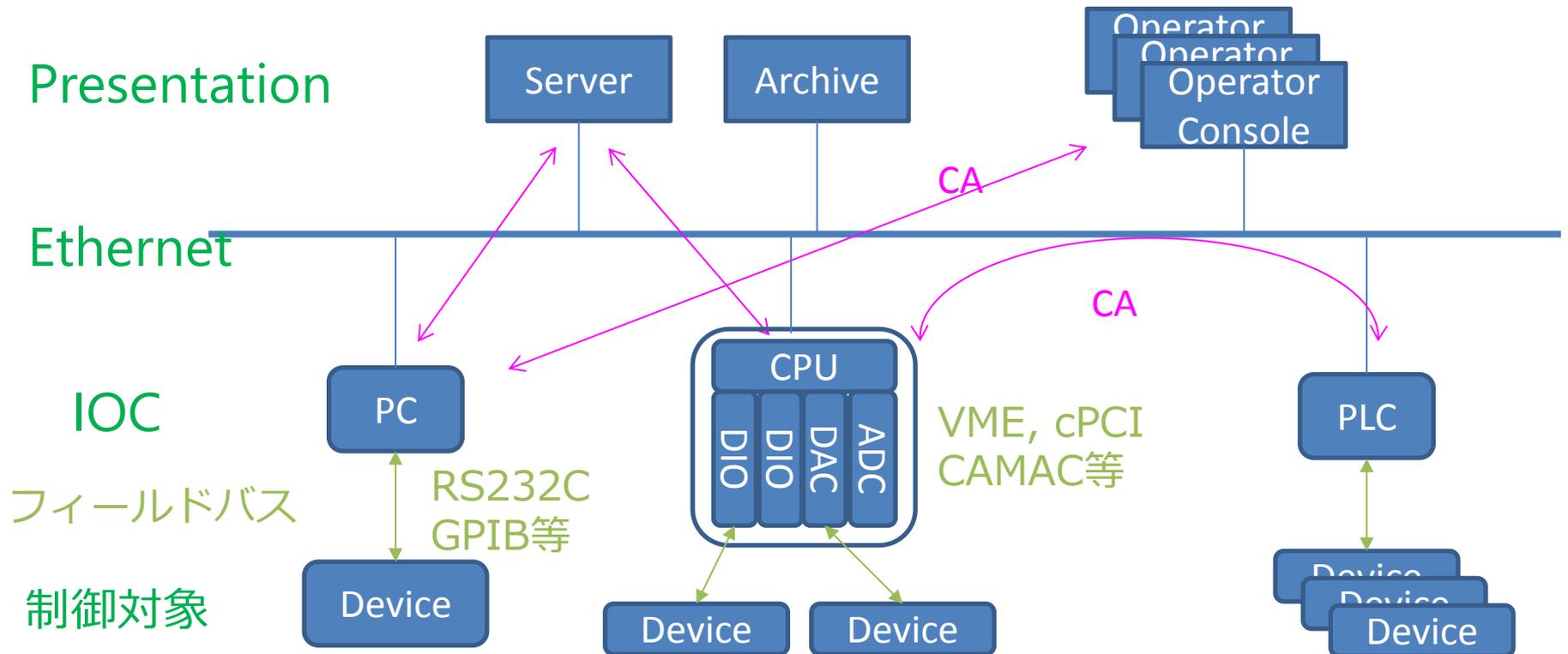
- 対象となるデバイスを入出力コントローラ(IOC, Input-Output Controller)が制御
- IOCはPCだったり、VMEだったり、PLCだったり、色々なパターンがある
- 制御対象デバイス依存部分



# EPICS : 通信プロトコル

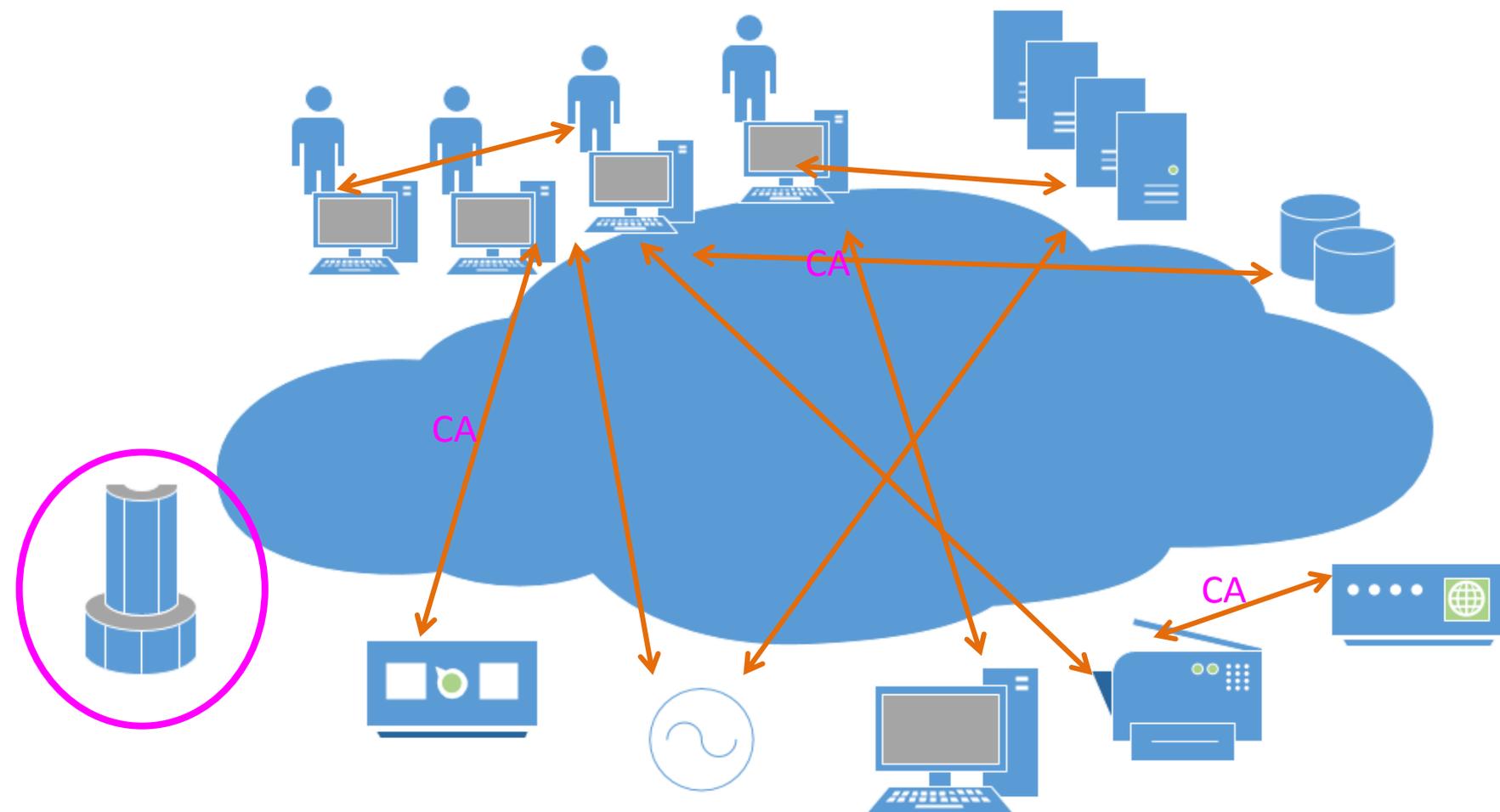
チャンネルアクセス(CA)プロトコルを通して  
状態変数(PV, Process Variable)のやりとりをする

- 上位層との通信、IOC間の通信ともにCAを使用
- デバイス非依存、ネットワーク透過、分散型



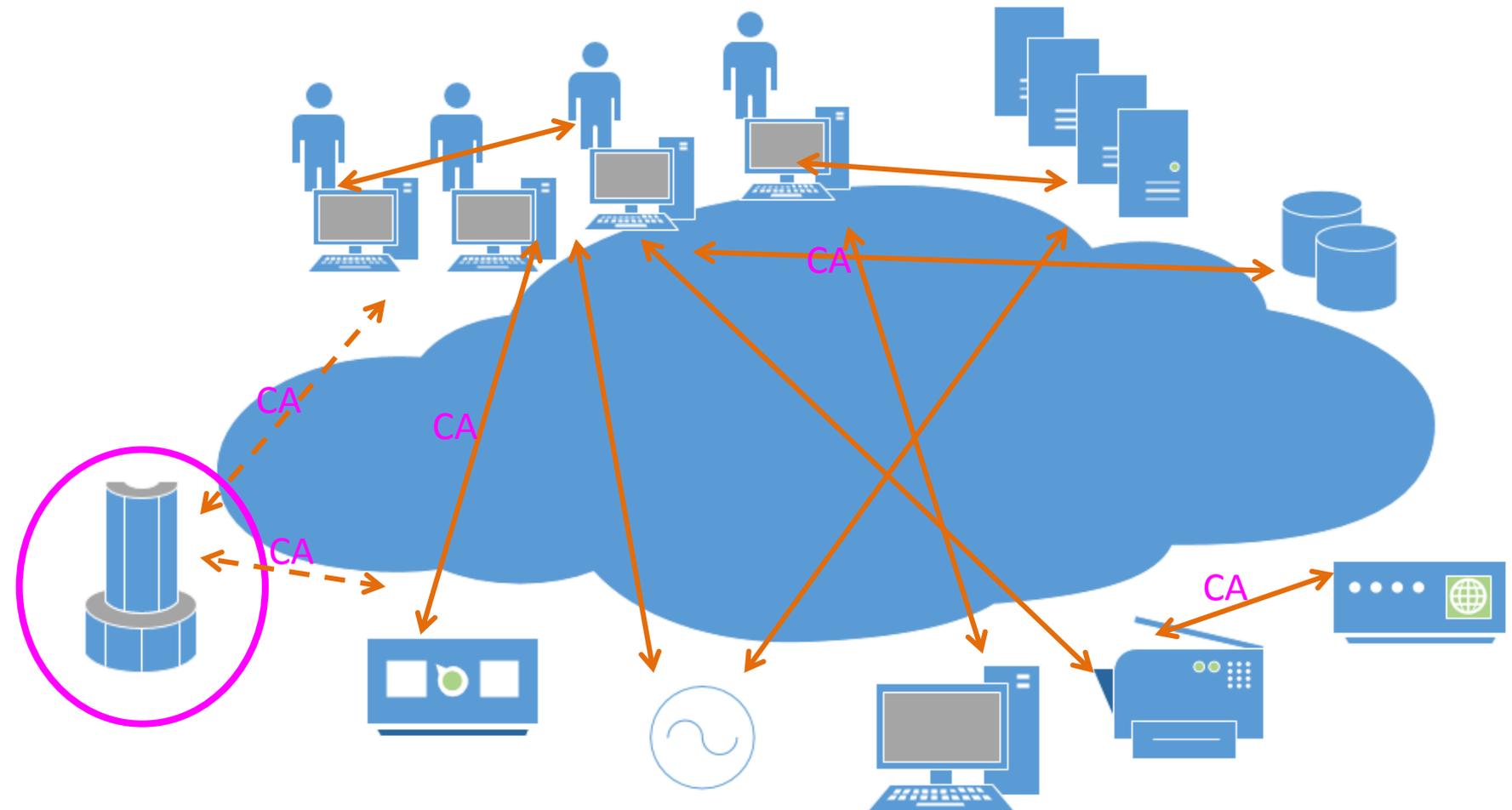
# 新しい装置（市販品）を買ってきたとき

既存の制御フレームワークに合わせる必要あり



# しかし、現実には

最初から加速器制御フレームワークに準拠している製品など、市販品にはほとんど存在しない：何らかの形で対処が必要



# EPICSの場合：1つの理想形

デバイスにEPICS IOC を内蔵している



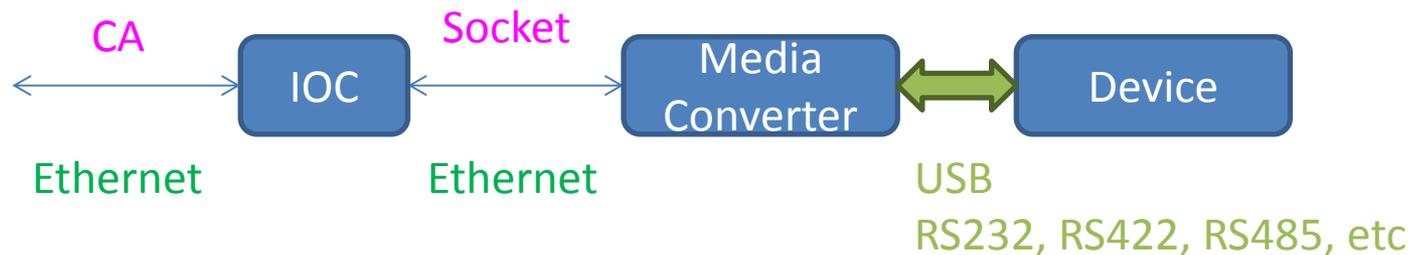
買ってきたものにEthernet付き  
EPICS IOC が動き、CAで通信できる  
電源いれてEthernetつないだらおしまい！

しかしそんな製品は(ほぼ)市販していない

## 現状：RS232C機器を例にして

Ethernet変換器（メディアコンバータ）を使用

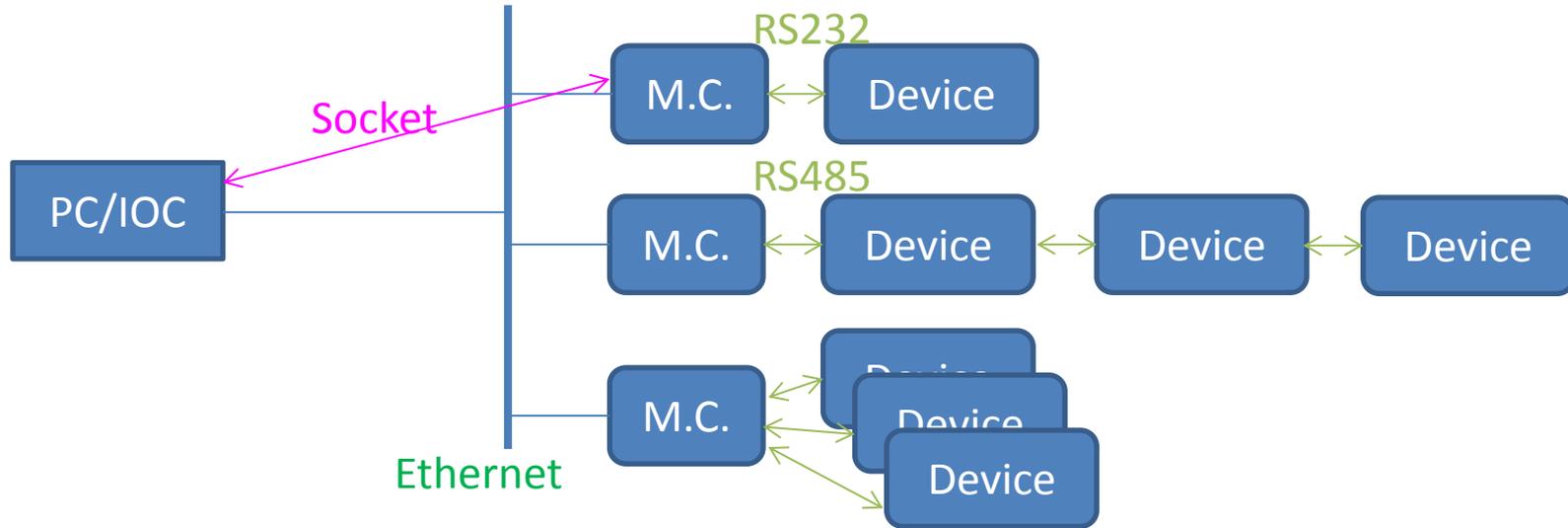
- IOCと対象デバイスとの距離が長くても良い
- 1つのIOCで複数デバイス制御も楽



PCにポートを増設して…とか、  
RS485の専用線を引っ張って…とか  
苦勞していた時代にくらべるとかなり楽になった

# 製品例

- RS232C をイーサネット変換する



## Serial-Ether メディアコンバータ

各社から出ている: 値段も機能も多種多様

1ポートのほか、多ポート(4, 8, 16 port)の製品もある



<http://www.lantronix.jp/products/xport.shtml>

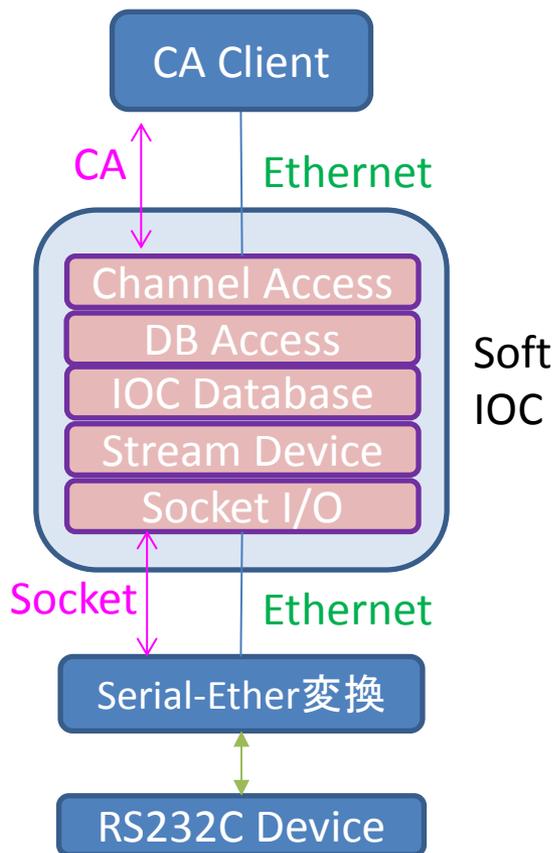


<http://japan.moxa.com/>

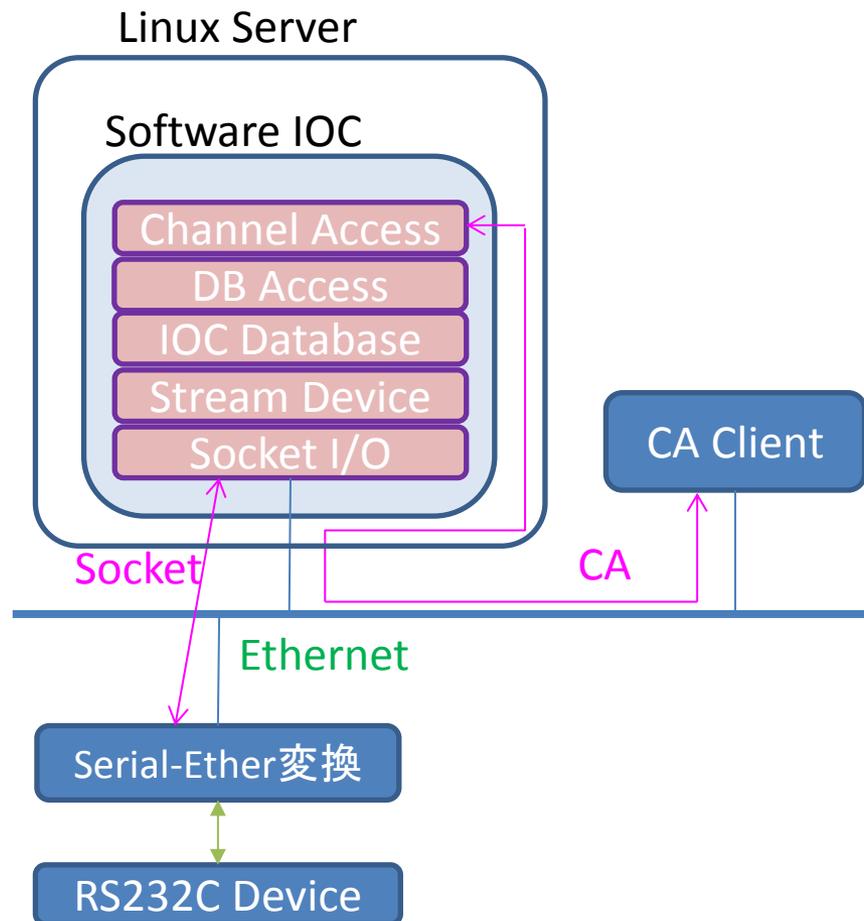
# EPICS + メディアコンバータで制御する例

- サーバ計算機で Soft IOC を動かして制御する

## 論理構成

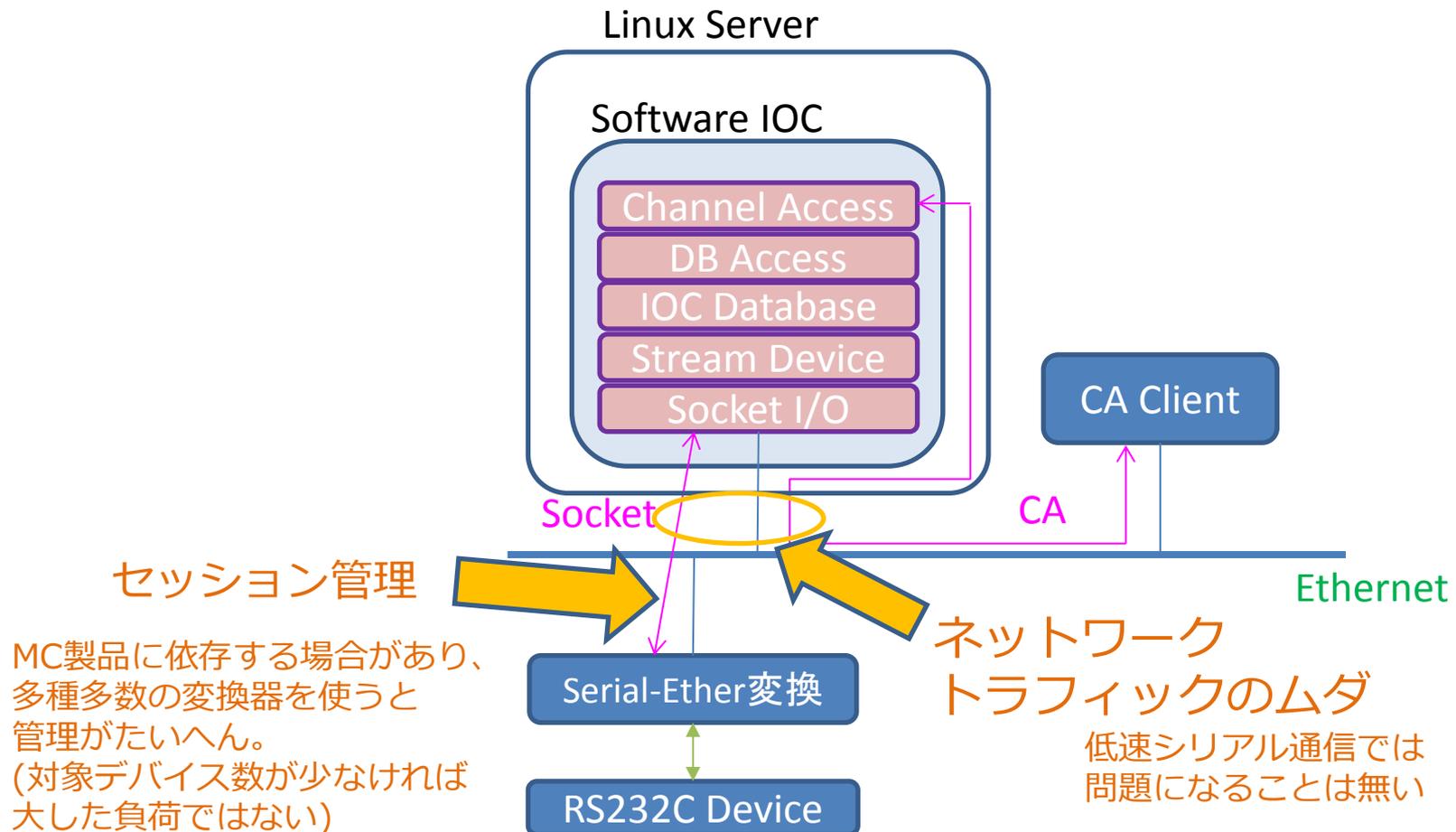


## 物理構成



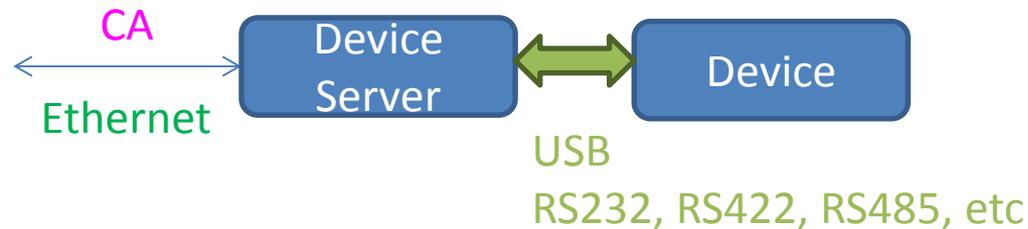
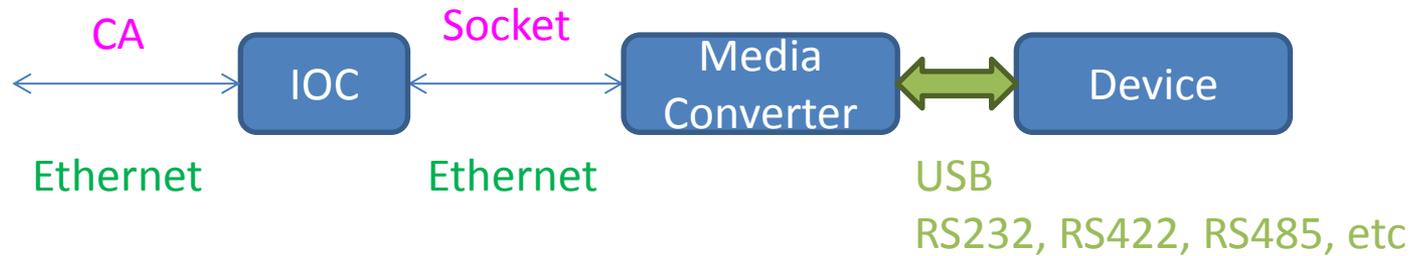
# Soft IOC + メディアコンバータ構成の欠点

- トラフィック増加につながる
- Socket通信の初期化や障害後の再接続など、セッション管理を自分でおこなう必要がある（CAでは規定済み）



# すこしでも理想に近づきたい

外付けの小型デバイスサーバがあれば良い

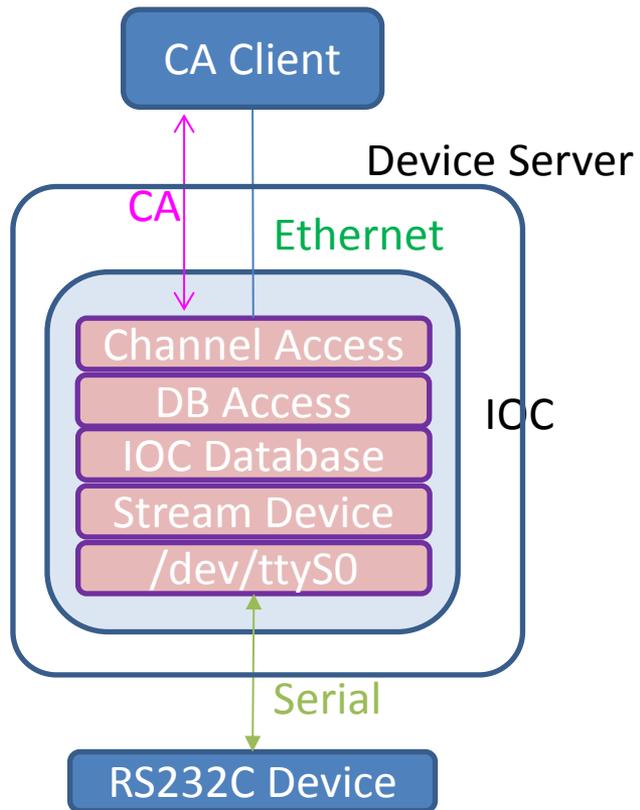


EPICS IOC 内蔵の小型デバイスサーバを用意する  
(PoEかUSB給電の方が理想)  
対象機器の近くに設置。そこから先はCAによって通信

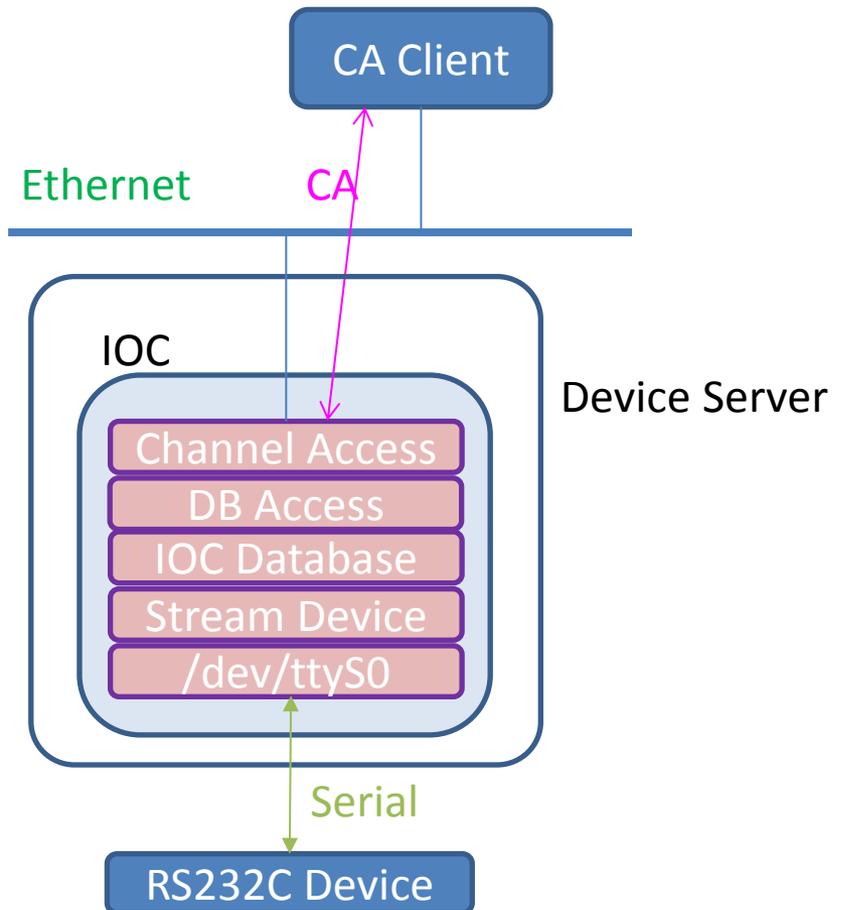
# EPICS内蔵デバイスサーバの場合

- 理想的な構成に少し近づく

## 論理構成

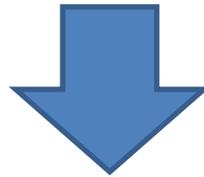


## 物理構成



## デバイスサーバの利点

- デバイス制御用の通信と、CAネットワークの分離
  - トラフィック軽減：必要なデータのみCAで通信
  - Socket通信部分のセッション管理不要：安定した制御実現
- 管理が容易
- 安価に製作可能
- 小型、低消費電力



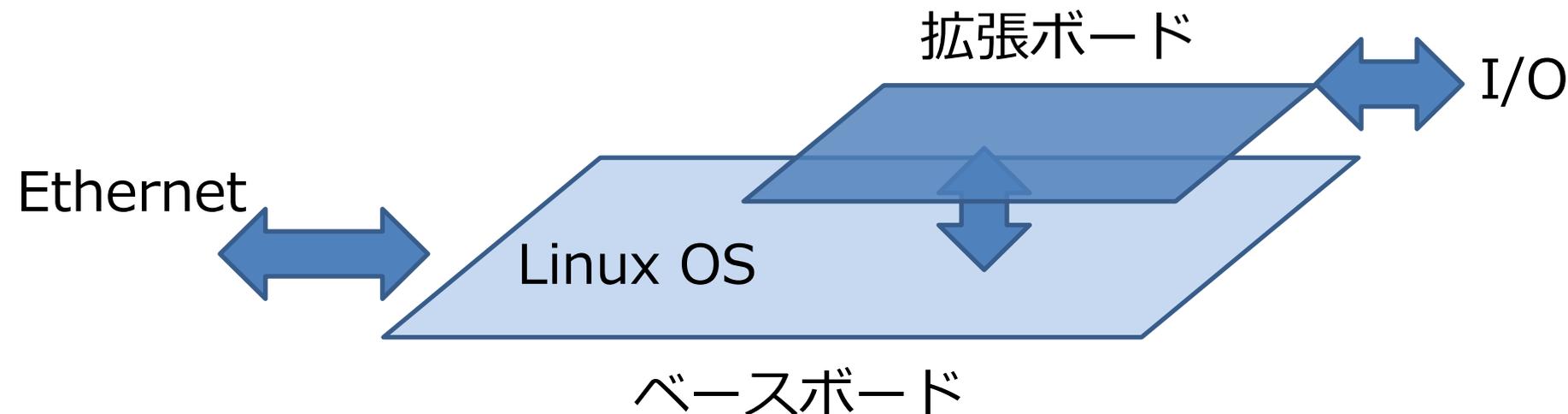
EPICS内蔵小型デバイスサーバを製作する

# 本日の内容

1. 開発の動機
2. ハードウェア候補と選定
3. ハードウェア設計・製作
4. ソフトウェア
5. デバイスサーバとしての利用実績
6. その他
7. まとめと情報公開について

# 小型デバイスサーバの設計方針

- イメージ概略



OS(EPICS IOC)はベースボード側にもたせる  
拡張ボードを交換することで目的に応じた  
I/Oに交換できるように  
(RS232C, USB, A/D, D/A, DIO など)

## 自作 or 市販品？

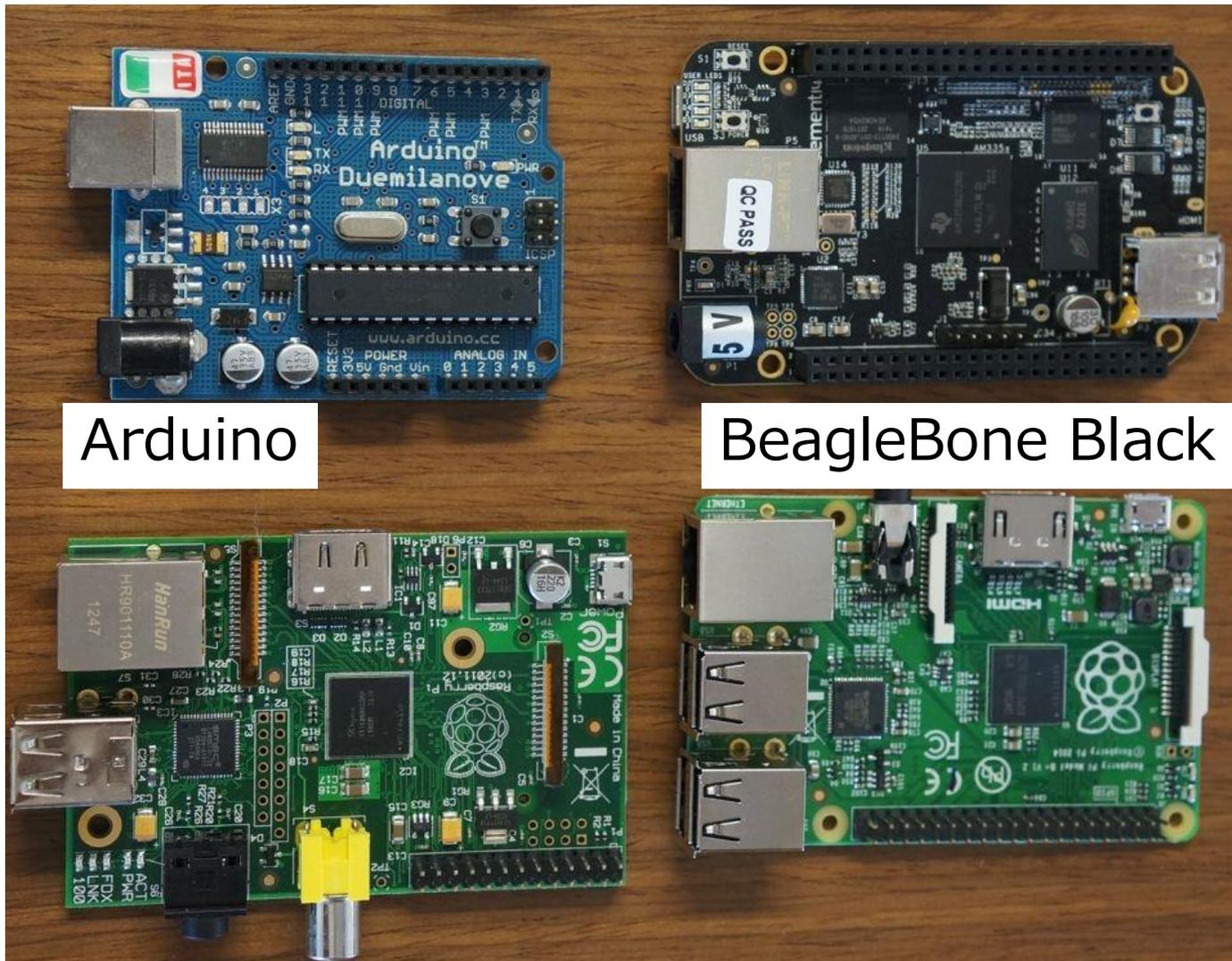
開発にかかるコスト（お金と時間の両方）を節約

- ベース部分は市販品を利用
- 必要に応じて、拡張部分のみ自作する
- ハード・ソフトともに多くの情報が得られる  
ベース製品を選定する

という方針を決定した

# ベースとなるハードウェア候補

名刺サイズ小型ボード



Arduino

BeagleBone Black

Raspberry Pi (model B)

Raspberry Pi (model B+)

# 候補、比較 1

	BeagleBone Black(Rev.C)	Raspberry Pi (ModelB)	Arduino Uno
CPU	TI AM3359 (ARM Cortex-A8)	Broadcom BCM2835 (ARM11)	Atmel ATmega 328
Speed	1 GHz	700 MHz	16 MHz
RAM	512MB DDR3L@400MHz	512MB SDRAM@400MHz	2 KB
Storage	Onboard eMMC 4GB microSD slot	SD slot	32 KB Flash
I/O(GPIO)	65	8	20
ADC	7	n/a	8
Ethernet	10/100 x1	10/100 x1	n/a
OS	Debian(default) Angstrom Linux Fedora etc	Raspbian(Debian) Pidora(Fedora) ARCH linux etc	n/a
Size [mm]	86.4 x 53.3	85.6 x 54	75 x 53.3
Price	¥ 6,500	¥ 3,940	¥ 3,024

## 候補、比較 2

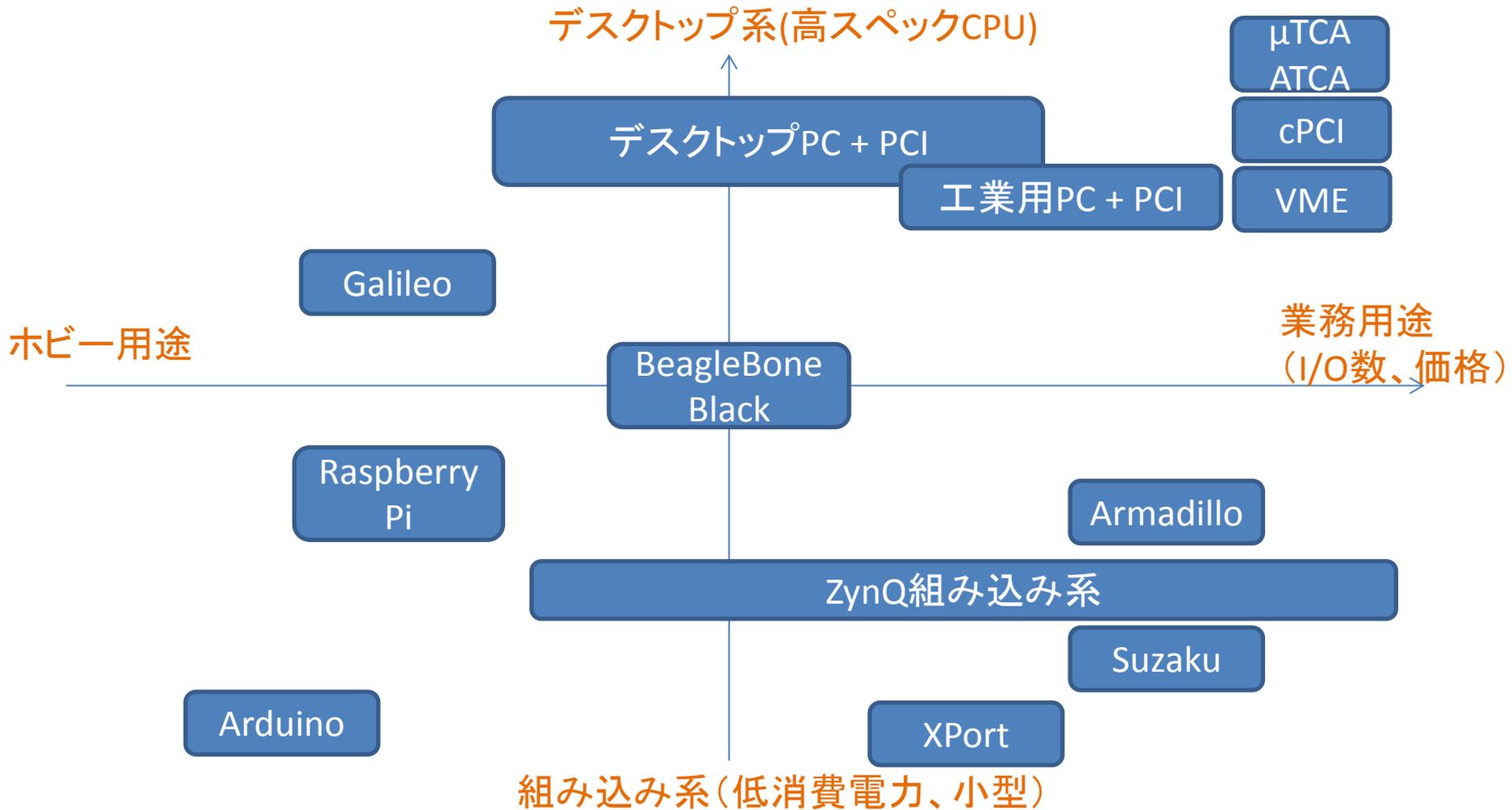
	Armadillo 840	Suzaku	Galileo
CPU	Renesas R-Mobile A1 (ARM Cortex-A9)	Xilinx Vertex-4 (PowerPC405)	Intel Quark Soc X1000
Speed	792 MHz	350 MHz	400 MHz
RAM	512MB DDR3@800MHz	64MB DDR2	256MB DDR3@800MHz
Storage	Onboard 128MB	Onboard 8MB	Onboard 8MB microSD Slot
I/O(GPIO)	112	86	20
ADC	n/a	n/a	8
Ethernet	10/100 x1	10/100 x1	10/100 x 1
OS	Debian(ATDE5)	Linux(ATDE5)	Linux
Size[mm]	98 x 60	72 x 47	106.8 x 72
Price	¥ 32,400	¥ 42,000	¥ 7,693

# (參考資料)CPU速度比較

CPU	Dhrystone MIPS值	DMIPS/MHz	
VAX-11/780	1MIPS @ 5MHz	0.2	
ARM2	4 MIPS @ 8 MHz	0.5	
Motorola 68030	11 MIPS @ 33 MHz	0.3	
Intel 386DX	11.4 MIPS @ 33 MHz	0.3	
Motorola 68060	88 MIPS @ 66 MHz	1.33	
ARM Cortex M3	125 MIPS @ 100MHz	1.25	
Intel Pentium	188 MIPS @ 100 MHz	1.88	
PowerPC 750	525 MIPS @ 233 MHz	2.3	
PowerPC 405	608 MIPS @ 400 MHz	1.52	Virtex 4
ARM11 v6KZ	875MIPS @ 700MHz	1.25	RasPi
ARM Cortex A8	2,000 MIPS @ 1.0 GHz	2.0	BBB
Intel Atom N270 (Single core)	3,846 MIPS @ 1.6 GHz	2.4	
ARM Cortex A15 (Quad core)	35,000 MIPS @ 2.5 GHz	14.0	
Intel Core i7 Extreme Edition 990x	159,000 MIPS @ 3.46 GHz	46.0	

# ベースとなるモジュール候補比較

- 用途と傾向（主観含む）



注: 厳密な性能比較図ではありません。大凡の傾向として見てください

# 最初のApplication : RS232C機器のデータ取得

cERLシールド壁 4 か所に設置したエリアモニタ12台の監視

→ 4-Port RS232C用 Device Server を 4 台 製作する

[詳細後述](#)

これに適したボードは何か？

	BeagleBone Black	Raspberry Pi	Arduino
RS232C 4port以上	○ HWで最大5.5	× HWで最大2	△ HWで1、SWで4
Ethernet	○	○	△ Shieldあり
OS & Epics	○ Linux	○ Linux	× n/a

今後の拡張性も考慮

# ハードウェア選定基準

- CPUパワー
- I/Oの数
- OS
  - EPICS環境構築が容易 (Linux)
- 価格
- 入手が容易
- RS232C ホスト機能必須
  - USB ホスト機能があればより良い

ベースとなる製品として  
BeagleBone Black (BBB) を選定した

## 疑問

これ、大丈夫なの？

信頼性とか、サポートとか、色々

## (当面の) 結論

# まず、やってみよう

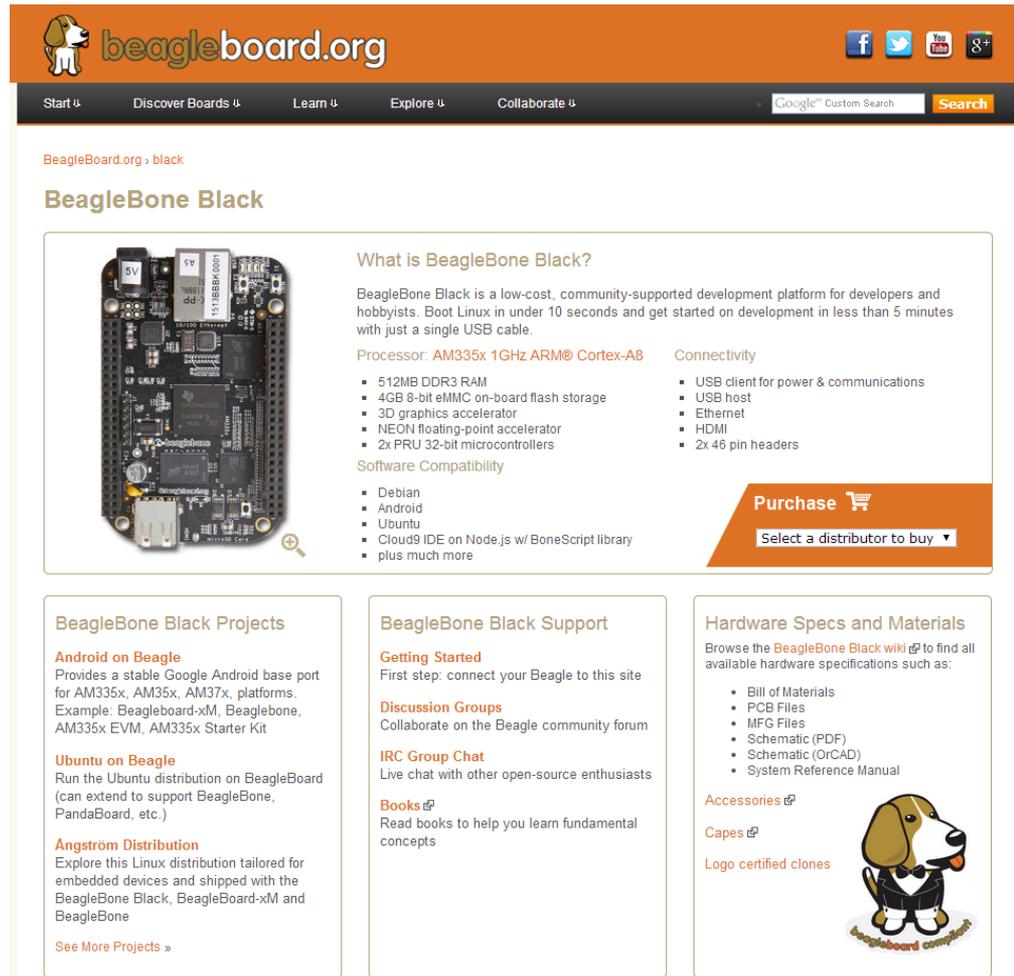
安い:「市販品」の威力絶大  
壊れたら交換する方針で。  
Criticalな所に使わなければ良い。  
自分で一から製作したからといって  
信頼性があるとは言えない。

# 本日の内容

1. 開発の動機
2. ハードウェア候補と選定
3. **ハードウェア設計・製作**
4. ソフトウェア
5. デバイスサーバとしての利用実績
6. その他
7. まとめと情報公開について

# BeagleBone Black を選定したその他の理由

- 開発者コミュニティがある
- オープンソースハードウェア：情報が全て公開されている
- “Cape”という拡張ボードの規格が整備されている
- Cape開発者が多数存在し、情報も公開されている
- OSは Linux
  - ✓ Angstrom
  - ✓ Debian
- 各種I/Oコネクタ配置が組み込みに適している



The screenshot shows the BeagleBoard.org website for the BeagleBone Black. The page features a navigation bar with links for Start, Discover Boards, Learn, Explore, and Collaborate, along with a search bar. The main content area includes a product image of the BeagleBone Black, a description of the board, and a list of features and connectivity options. A 'Purchase' button is visible, along with a dropdown menu to select a distributor. The page also includes sections for BeagleBone Black Projects, BeagleBone Black Support, and Hardware Specs and Materials.

beagleboard.org

Start Discover Boards Learn Explore Collaborate Google Custom Search Search

BeagleBoard.org > black

## BeagleBone Black



What is BeagleBone Black?

BeagleBone Black is a low-cost, community-supported development platform for developers and hobbyists. Boot Linux in under 10 seconds and get started on development in less than 5 minutes with just a single USB cable.

Processor: **AM335x 1GHz ARM® Cortex-A8** Connectivity

- 512MB DDR3 RAM
- 4GB 8-bit eMMC on-board flash storage
- 3D graphics accelerator
- NEON floating-point accelerator
- 2x PRU 32-bit microcontrollers

- USB client for power & communications
- USB host
- Ethernet
- HDMI
- 2x 46 pin headers

Software Compatibility

- Debian
- Android
- Ubuntu
- Cloud9 IDE on Node.js w/ BoneScript library
- plus much more

**Purchase** 

Select a distributor to buy ▾

### BeagleBone Black Projects

**Android on Beagle**  
Provides a stable Google Android base port for AM335x, AM35x, AM37x, platforms.  
Example: Beagleboard-xM, Beaglebone, AM335x EVM, AM335x Starter Kit

**Ubuntu on Beagle**  
Run the Ubuntu distribution on BeagleBoard (can extend to support BeagleBone, PandaBoard, etc.)

**Angstrom Distribution**  
Explore this Linux distribution tailored for embedded devices and shipped with the BeagleBone Black, BeagleBoard-xM and BeagleBone

[See More Projects >](#)

### BeagleBone Black Support

**Getting Started**  
First step: connect your Beagle to this site

**Discussion Groups**  
Collaborate on the Beagle community forum

**IRC Group Chat**  
Live chat with other open-source enthusiasts

**Books**  
Read books to help you learn fundamental concepts

### Hardware Specs and Materials

Browse the [BeagleBone Black wiki](#) to find all available hardware specifications such as:

- Bill of Materials
- PCB Files
- MFG Files
- Schematic (PDF)
- Schematic (OrCAD)
- System Reference Manual

**Accessories**

**Capes**

Logo certified clones



# Capeとは？

- BBB拡張ボードの規格で、ボード上のEEPROMに対応するdriverが自動的に読み込まれる仕組み
- EEPROMをI2Cの特定アドレスに接続
- “Cape”を変えることで簡単に機能を追加、変更できる

REF: BBONEBLK\_SRM BeagleBone Black System Reference Manual Rev A5.6

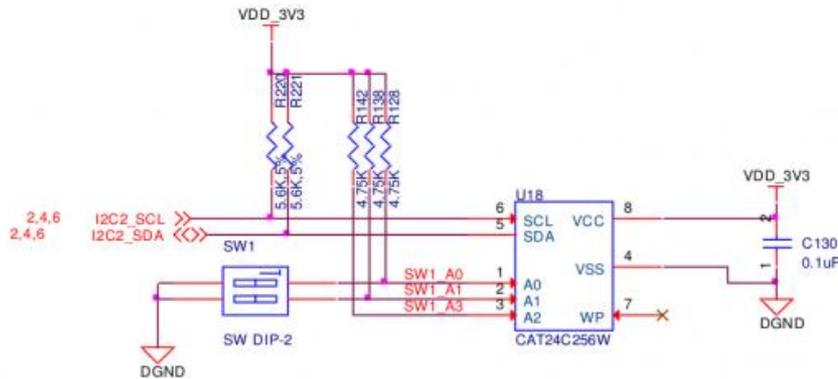
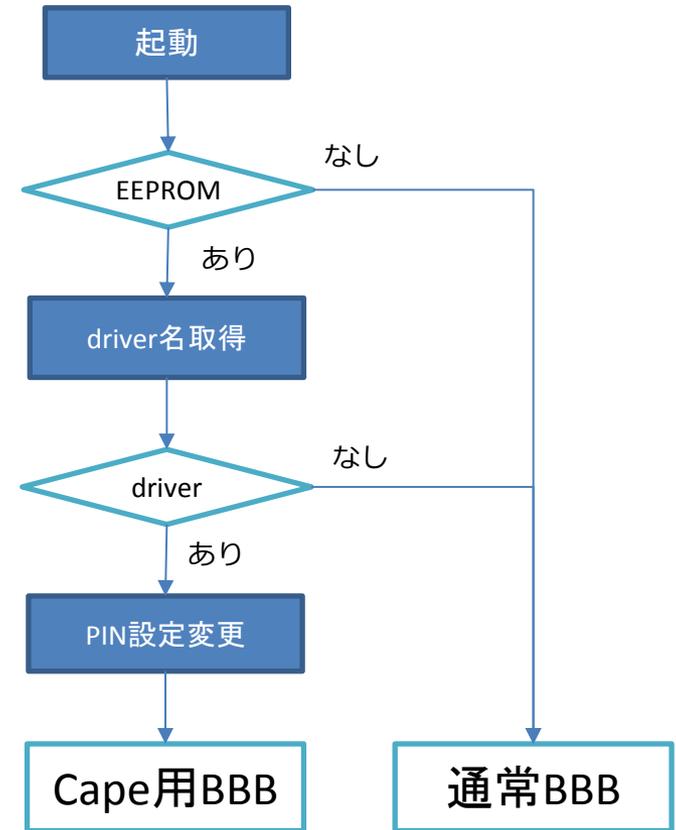


Figure 58. Expansion Board EEPROM Without Write Protect

CC-SA-BY -- Gerald Cooley Beagleboard.org



# BBB拡張ボード Header Mode

- “Cape”でHeaderのモードを設定
- 拡張Header Modeは0～7まで
- モードの変更はEEPROMかdriverで行う
- 各PIN毎に別のモードを設定可能

## Black eMMC and HDMI pins

P9				P8				
Pin	Color	Label	Pin	Color	Label	Pin	Color	Label
DGND	1	DGND	DGND	1	DGND	DGND	1	DGND
VDD_3V3	3	VDD_3V3	MMC1_DAT6	3	MMC1_DAT7	MMC1_DAT6	3	MMC1_DAT7
VDD_5V	5	VDD_5V	MMC1_DAT2	5	MMC1_DAT3	MMC1_DAT2	5	MMC1_DAT3
SYS_5V	7	SYS_5V	GPIO_66	7	GPIO_67	GPIO_66	7	GPIO_67
PWR_BUTTON	9	SYS_RESETN	GPIO_69	9	GPIO_68	GPIO_69	9	GPIO_68
GPIO_30	11	GPIO_60	GPIO_45	11	GPIO_44	GPIO_45	11	GPIO_44
GPIO_31	13	GPIO_40	GPIO_23	13	GPIO_26	GPIO_23	13	GPIO_26
GPIO_48	15	GPIO_51	GPIO_47	15	GPIO_46	GPIO_47	15	GPIO_46
GPIO_4	17	GPIO_5	GPIO_27	17	GPIO_65	GPIO_27	17	GPIO_65
I2C2_SCL	19	I2C2_SDA	GPIO_22	19	MMC1_CMD	GPIO_22	19	MMC1_CMD
GPIO_3	21	GPIO_2	MMC1_CLK	21	MMC1_DAT5	MMC1_CLK	21	MMC1_DAT5
GPIO_49	23	GPIO_15	MMC1_DAT4	23	MMC1_DAT1	MMC1_DAT4	23	MMC1_DAT1
GPIO_117	25	GPIO_14	MMC1_DAT0	25	GPIO_61	MMC1_DAT0	25	GPIO_61
GPIO_125	27	SPI1_CS0	LCD_VSYNC	27	LCD_PCLK	LCD_VSYNC	27	LCD_PCLK
SPI1_DO	29	GPIO_122	LCD_HSYNC	29	LCD_AC_BIAS_E	LCD_HSYNC	29	LCD_AC_BIAS_E
SPI1_SCLK	31	VDD_ADC	LCD_DATA14	31	LCD_DATA15	LCD_DATA14	31	LCD_DATA15
AIN4	33	GND_ADC	LCD_DATA13	33	LCD_DATA11	LCD_DATA13	33	LCD_DATA11
AIN6	35	AIN5	LCD_DATA12	35	LCD_DATA10	LCD_DATA12	35	LCD_DATA10
AIN2	37	AIN3	LCD_DATA8	37	LCD_DATA9	LCD_DATA8	37	LCD_DATA9
AIN0	39	AIN1	LCD_DATA6	39	LCD_DATA7	LCD_DATA6	39	LCD_DATA7
GPIO_20	41	GPIO_7	LCD_DATA4	41	LCD_DATA5	LCD_DATA4	41	LCD_DATA5
DGND	43	DGND	LCD_DATA2	43	LCD_DATA3	LCD_DATA2	43	LCD_DATA3
DGND	45	DGND	LCD_DATA0	45	LCD_DATA1	LCD_DATA0	45	LCD_DATA1

Mode 0

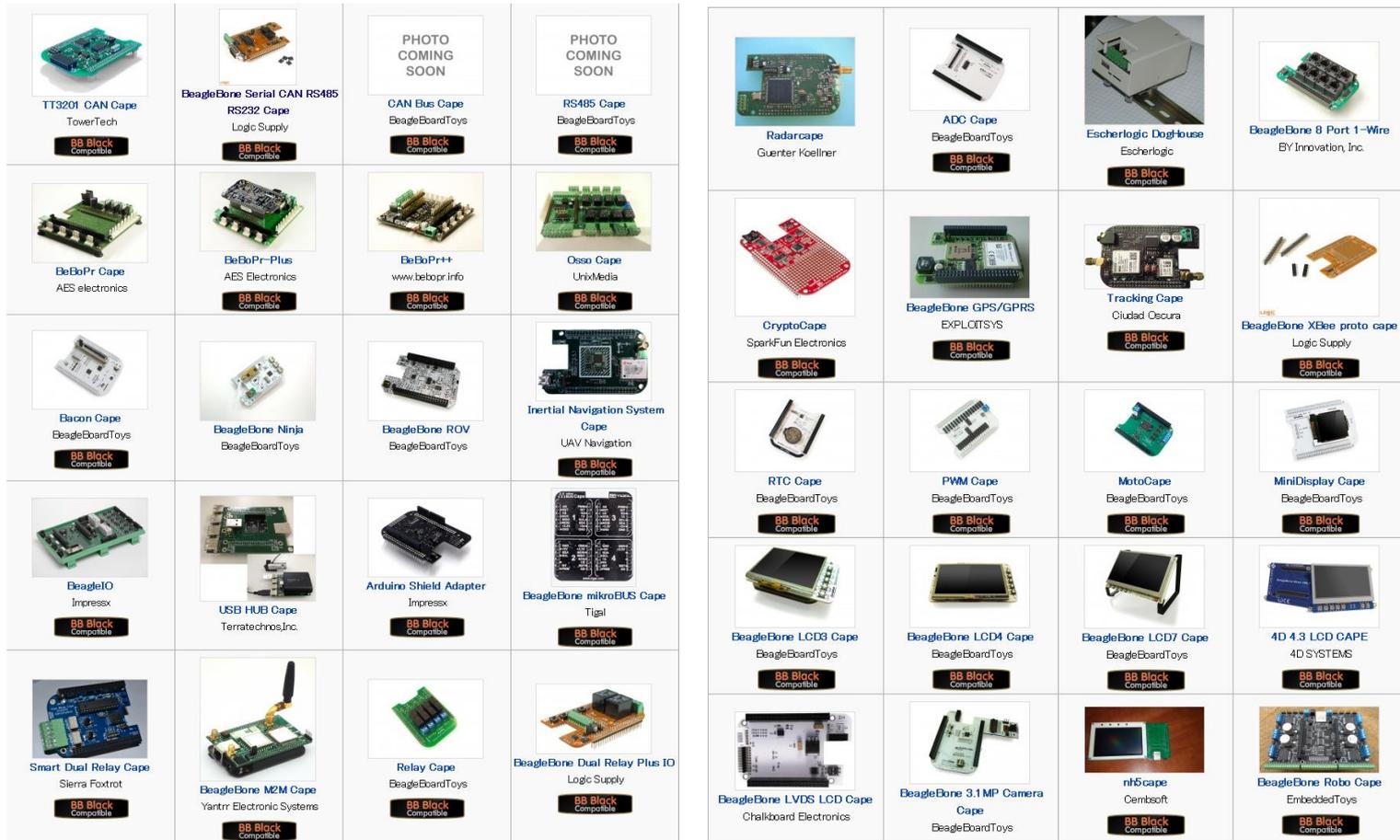
## 4.5 serial UARTs

P9				P8				
Pin	Color	Label	Pin	Color	Label	Pin	Color	Label
DGND	1	DGND	DGND	1	DGND	DGND	1	DGND
VDD_3V3	3	VDD_3V3	GPIO_38	3	GPIO_39	GPIO_38	3	GPIO_39
VDD_5V	5	VDD_5V	GPIO_34	5	GPIO_35	GPIO_34	5	GPIO_35
SYS_5V	7	SYS_5V	GPIO_66	7	GPIO_67	GPIO_66	7	GPIO_67
PWR_BUTTON	9	SYS_RESETN	GPIO_69	9	GPIO_68	GPIO_69	9	GPIO_68
UART4_RXD	11	GPIO_60	GPIO_45	11	GPIO_44	GPIO_45	11	GPIO_44
UART4_TXD	13	GPIO_40	GPIO_23	13	GPIO_26	GPIO_23	13	GPIO_26
GPIO_48	15	GPIO_51	GPIO_47	15	GPIO_46	GPIO_47	15	GPIO_46
GPIO_4	17	GPIO_5	GPIO_27	17	GPIO_65	GPIO_27	17	GPIO_65
UART1_RTSN	19	UART1_CTSN	GPIO_22	19	GPIO_63	GPIO_22	19	GPIO_63
UART2_TXD	21	UART2_RXD	GPIO_62	21	GPIO_37	GPIO_62	21	GPIO_37
GPIO_49	23	UART1_TXD	GPIO_36	23	GPIO_33	GPIO_36	23	GPIO_33
GPIO_117	25	UART1_RXD	GPIO_32	25	GPIO_61	GPIO_32	25	GPIO_61
GPIO_125	27	GPIO_123	GPIO_86	27	GPIO_88	GPIO_86	27	GPIO_88
GPIO_121	29	GPIO_122	GPIO_87	29	GPIO_89	GPIO_87	29	GPIO_89
GPIO_120	31	VDD_ADC	UART5_CTSN+	31	UART5_RTSN	UART5_CTSN+	31	UART5_RTSN
AIN4	33	GND_ADC	UART4_RTSN	33	UART3_RTSN	UART4_RTSN	33	UART3_RTSN
AIN6	35	AIN5	UART4_CTSN	35	UART3_CTSN	UART4_CTSN	35	UART3_CTSN
AIN2	37	AIN3	UART5_TXD+	37	UART5_RXD+	UART5_TXD+	37	UART5_RXD+
AIN0	39	AIN1	GPIO_76	39	GPIO_77	GPIO_76	39	GPIO_77
GPIO_20	41	UART3_TXD	GPIO_74	41	GPIO_75	GPIO_74	41	GPIO_75
DGND	43	DGND	GPIO_72	43	GPIO_73	GPIO_72	43	GPIO_73
DGND	45	DGND	GPIO_70	45	GPIO_71	GPIO_70	45	GPIO_71

Mode 6

# 市販品Cape

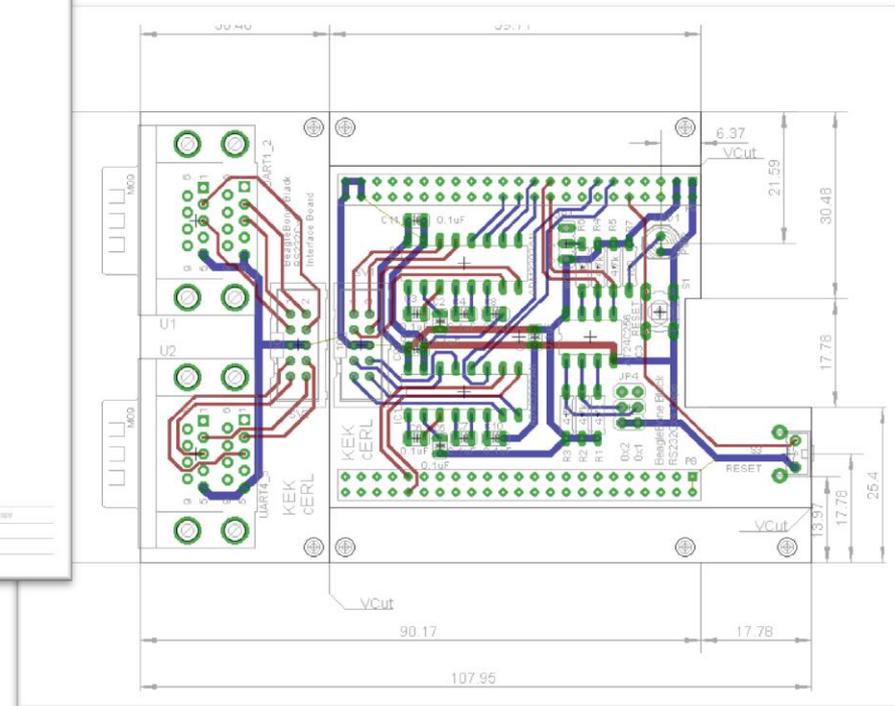
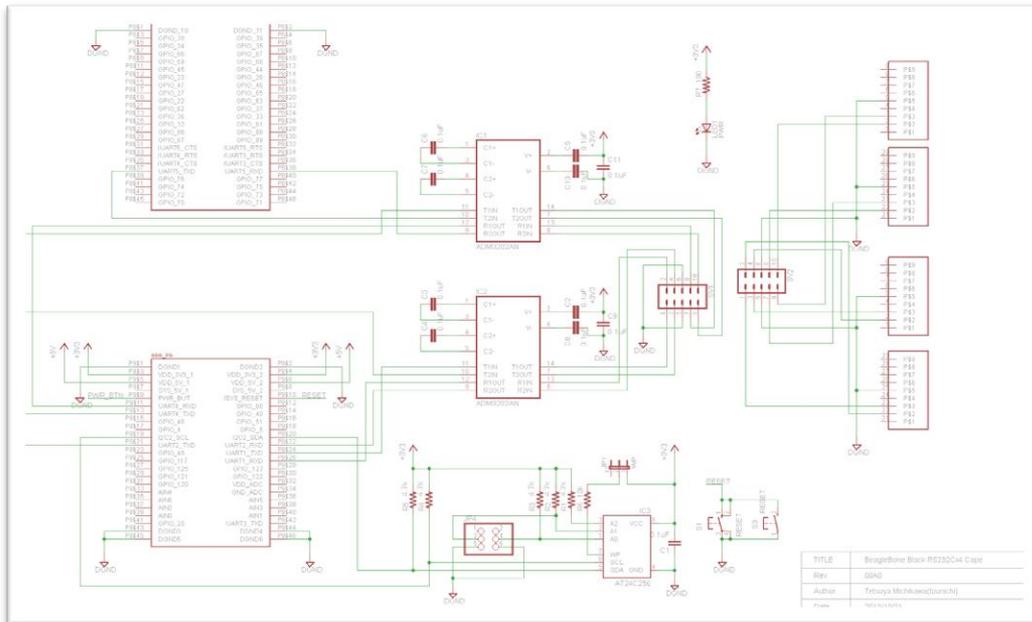
- 国内外で色々なCapeが市販されている



- 今回は必要な機能をもったCapeがなかったので、新規に設計・作成することにした。(単純な回路のみ)

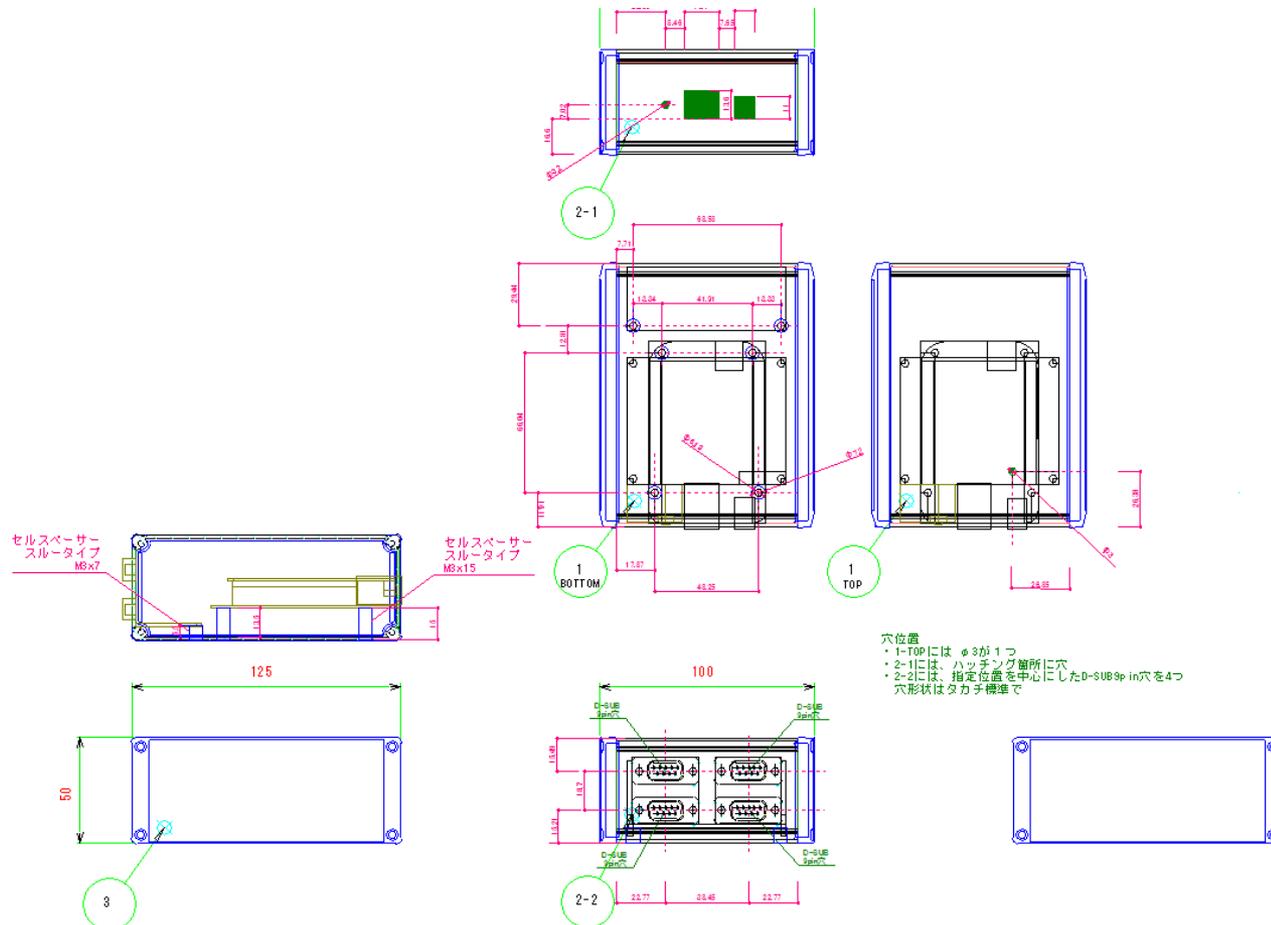
# Serial x4 Cape製作

- Eagle CAD Free版で基板を設計
- 2層基板で実装部品はDIPで設計
  - 基板はプリント基板メーカーに外注(約6,000円/枚)
  - 部品実装は自前(実装部品代は約1,500円/枚)
  - 次回作成時はSMT (表面実装) 化予定

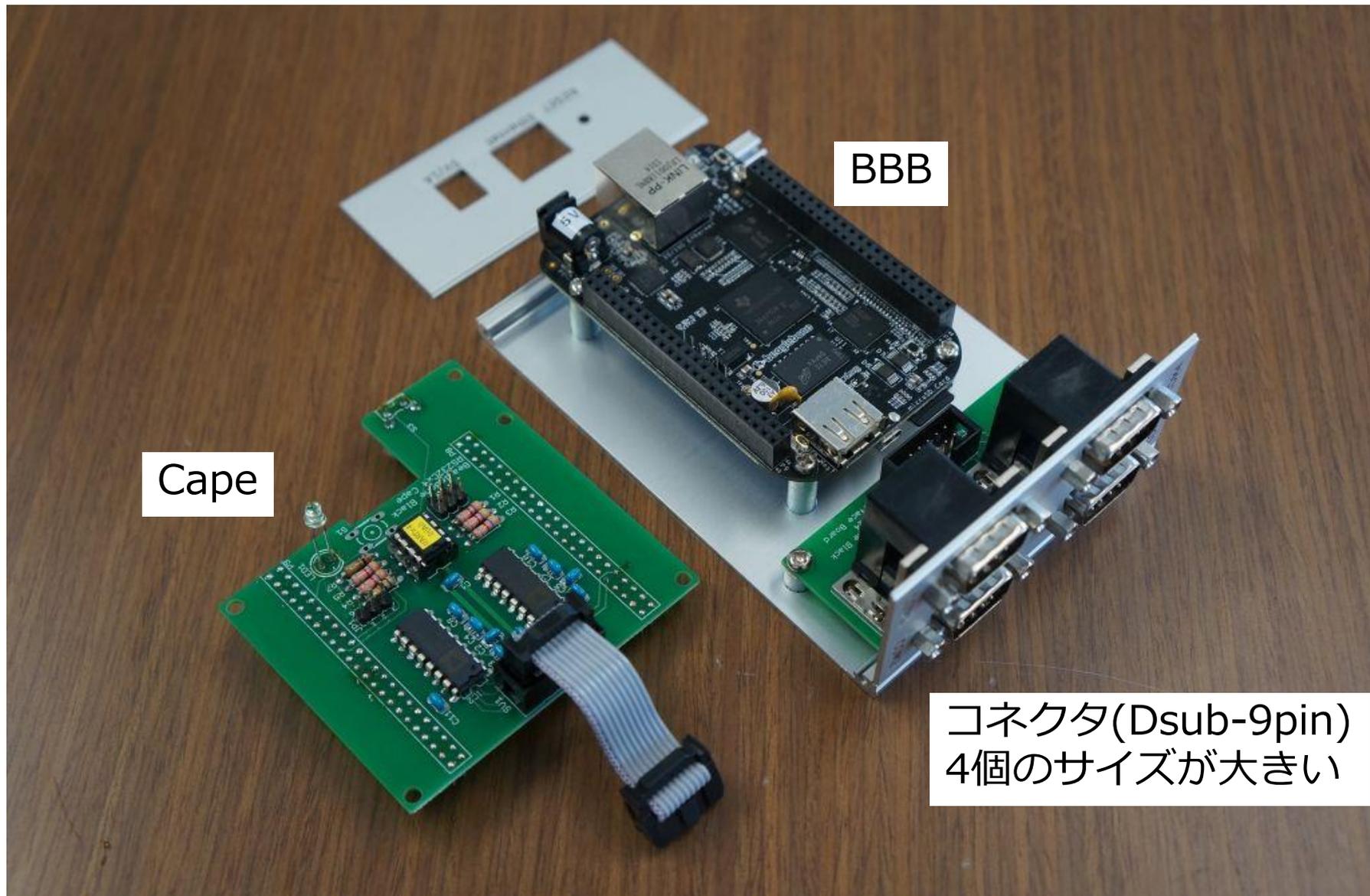


# ケース製作

- タカチ kc5-13-10をベースにして作成
- 穴位置、スペーサ等のデータを作成して追加工
- CADソフトはAR-CAD一般版を使用



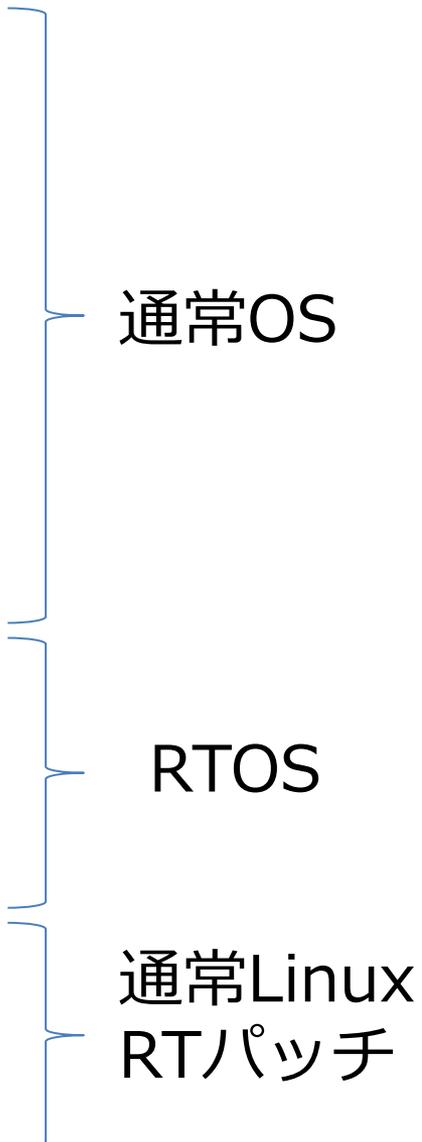
# ハードウェア完成写真



# 本日の内容

1. 開発の動機
2. ハードウェア候補と選定
3. ハードウェア設計・製作
4. ソフトウェア
5. デバイスサーバとしての利用実績
6. その他
7. まとめと情報公開について

# OSについて

- 手軽に使える選択肢として
    - AngstromLinux (Rev A,B でのデフォルト)
    - Debian (Rev C でのデフォルト)
  - その他には
    - Fedora
    - Android
  - VxWorks
  - OpenRTM-aist
  - RT-PREEMPT
  - Xenomai
- 
- 通常OS
- RTOS
- 通常Linux  
RTパッチ

# ソフトウェア開発

- RTOSが必要？ → 今回は不要
- デフォルトのAngstrom Linuxで作成した
  - 情報は多い
  - よく使われるLinux（RedHat, Debian等）と設定方法が少し違うので、最初は手間取った
- 開発環境
  - **セルフコンパイル**環境。BBBにログインして開発
  - ApplicationはNFSマウントしたディスクに配置
  - X環境は使わない（遅いし、メモリの無駄）

# EPICS 関連

- EPICS Base
  - EPICS base 3.14.12.4以降ではターゲット環境としてLinux-ARMが最初から用意されており、初期設定は楽
  - インストール方法の詳細はwiki を参照  
<http://cerldev.kek.jp/trac/EpicsUsersJP/wiki/epics/bbb/epics>
  - コンパイル時に追加Perlモジュールが必要なので、個別にインストールしていく
- EPICS Extensions
  - よく使う Asyn, StreamDevice, Sequencer 等は全て問題なくコンパイル、実行可能
- Rev. C デフォルトOSのDebianではもっと楽
  - 開発環境を入れて、CONFIG\_SITE編集、make一発

# 本日の内容

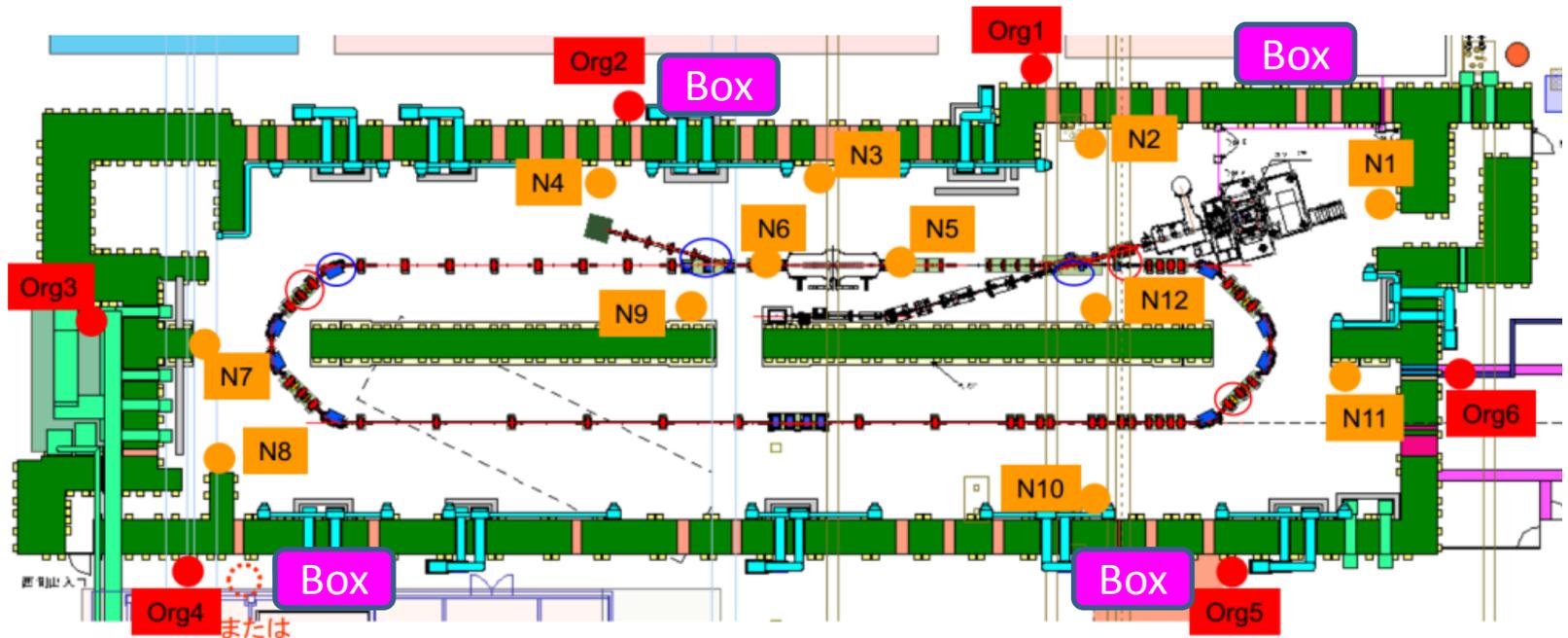
1. 開発の動機
2. ハードウェア候補と選定
3. ハードウェア設計・製作
4. ソフトウェア
5. デバイスサーバとしての利用実績
6. その他
7. まとめと情報公開について

# ALOKA社製エリアモニタ：cERL 配置図

- センサーは加速器室内に設置し、壁外に計測・表示器を取り付け
- インターロックは別系統でリレー接続。RS232Cは数値取得のみ
- 合計12台（橙色表示）

● 放管設置モニタ(Orange Monitor, 6台)

● 加速器室内エリアモニター(ALOKA, 10+2台)



N1 : NE1	入口	N7 : SW1	第1Arc西壁
N2 : NE2	入射空洞北壁	N8 : SW2	西出入口
N3 : NW1	主空洞北壁	N9 : SW3	ダンプ分岐中壁
N4 : NW2	ダンプ北壁	N10 : SE1	LCS南壁
N5 : NW3	MLSC 上流	N11 : SE2	第2Arc東壁
N6 : NW4	MLSC 下流	N12 : SE3	CircFC中壁

# 設置写真

製作したボックスを固定（すこし強引に仮設置...のはずがそのまま）

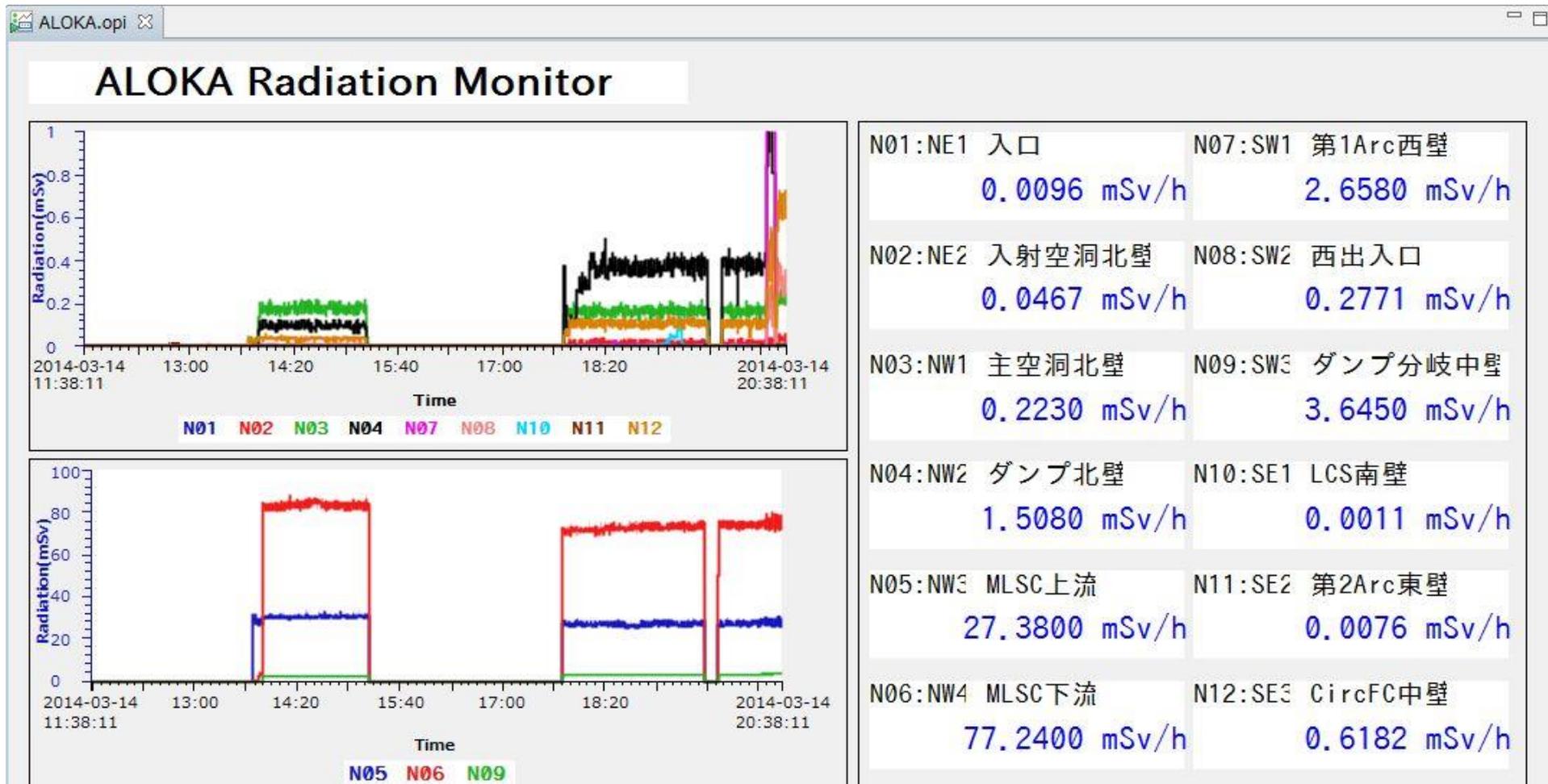


## ソフトウェア：EPICS IOC

- Asyn+StreamDeviceで作成
  - 放射線測定値を1秒毎に読み出す
  - SerialPort毎に Soft IOC を作成
- IOCプログラム本体はNFSサーバに置いた
- 電源ONで自動的にEPICSが動作するように設定

# 結果：データ取得例

- リアルタイム数値表示と時系列プロット（CSSで作成）
- 半年以上再起動なしで連続稼働。特に問題なし。



# 本日の内容

1. 開発の動機
2. ハードウェア候補と選定
3. ハードウェア設計・製作
4. ソフトウェア
5. デバイスサーバとしての利用実績
6. その他
7. まとめと情報公開について

## 6. その他：問題点、雑感など

- Onboard eMMC (Flash) からOSが立ち上がらない
  - 今回作成したCapeを使用したときの話
  - UARTで使用するGPIOのpinがeMMCと被っていた
  - microSDカードからは立ち上がるのでとりあえずOK
  - 回避策はあるが ... 次回製作時に検討する予定
- ケース加工が基板に比べて高い
  - ケース自体は1,000円、加工費が10,000円
  - CNCや3Dプリンタ等で安く作れないか、要検討
  - ケース無しでよければ安価
- 「入手性が良いこと」を条件にスタートしたのに、人気がため(!?)入手困難に。今は多少改善された。

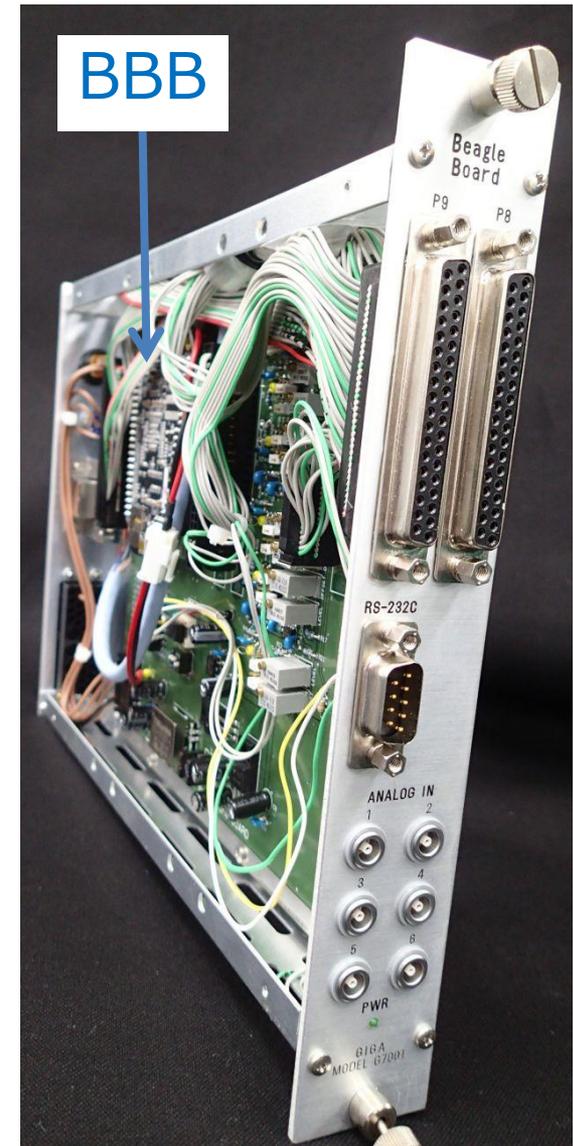
# 応用：多機能バージョンの製作

- NIMケース版を製作した
  - 60 bit 3.3 V TTL Digital I/O
  - 6 Channel Analog IN ( $\pm 10V$ )
  - 1 port RS232C (D-Sub 9pin)
- 基本動作は問題無し  
あとは実運用

単体でも動かせるように  
DC5V入力と切り替えSW

LANコネクタも背面

背面



## 7. まとめ

- 市販の名刺サイズ小型ボード BeagleBone Black を使用した小型デバイスサーバを製作した
- 思い立ってから設計・製作・運用まで短期間に実現できた
- 4ポートのシリアルデバイスサーバとしてcERLの運転で使用し、半年以上問題なく連続稼働中
- NIM版なども製作
- BBBは Quick Prototype ベースとして非常に良い

# 製作費用

	今回(試作)	量産
BeagleBone Black本体	4,500	6,500
microSD(4GB)	500	500
Cape基板 製造	6,000	200
部品	1,500	2,000
実装	-	1,000
ケース	11,000	3,000
合計	23,500	13,200

今回は4台製作した。表は1台あたりの値段

試作時のBBB本体価格は ¥4,500 であったが  
現在のRevisionで価格UPした  
フラッシュメモリ容量増加 2 GB → 4 GB

# 参考：BeagleBone Black購入先

## 国内

- 秋月電子通商 <http://akizukidenshi.com/catalog/g/gM-06867/>
- RS <http://jp.rs-online.com/web/p/processor-microcontroller-development-kits/7753805/>
- マルツ [http://www.marutsu.co.jp/shohin\\_238963/](http://www.marutsu.co.jp/shohin_238963/)
- 若松電商 <http://www.wakamatsu-net.com/cgi-bin/biz/pageshousai.cgi?code=38330030&CATE=3833>
- 共立エレシヨップ <http://eleshop.jp/shop/g/gE2P361/>



ホーム > 半導体 > 半導体開発キット > プロセッサ/マイクロコントローラ開発キット

## 半導体, BeagleBone Black 開発



購入単位は 1 個

RSオンラインでのご注文にはユーザー登録をお勧めします。オンライン注文なら通常配送料が安くなるほか、注文履歴の確認やパーツリストの作成など便利な機能が無料でご利用できます。ログイン/ユーザー登録(無料)はこちら

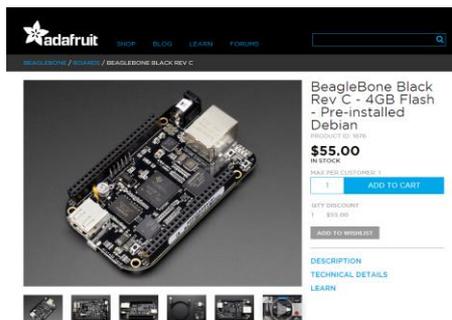
数量	単価
1	¥6,300

一時的な在庫切れ、もしくはお取寄せ商品 入荷次第お届け

1

## 海外

- Adafruit <https://www.adafruit.com/products/1876>
- Digi-key <http://www.digikey.jp/product-detail/ja/BB-BBLK-000/BB-BBLK-000-REVC-ND/4842211>
- Special Computing <http://specialcomp.com/beaglebone/>
- Spark fun <https://www.sparkfun.com/products/12857>



## 情報共有、今後の予定

- 製作した基板(Cape)の情報やケース図面など、全て公開している
    - EPICS Users JP wiki を参照
  - 誰かすでに作っている例があれば情報を共有し、活発な開発につなげたい
  - 興味がある方は、連絡をお願いします
  - 今後の予定
    - シリアル 1 portモデル製作、4 port タイプ量産
    - USBホスト（既に実績あり）
    - PoE対応
    - 様々な I/O に使用 (Digital/Analog)
    - 単純な表示端末や入退室の表示用PCの置き換え
- …その他、色々アイデアあり

Thank you!