



Raspberry Pi

ベアメタルプログラミングのススメ

プログラマの日 合宿

2017/09/16(土)@湯涌温泉

お題目

- Raspberry Pi ～ベアメタルとの出会い～
- ベアメタルとは
- 起動までの流れ
- 開発環境
- 類似マイコンとの比較
- まとめ

Raspberry Pi ～ベアメタルとの出会い～

- きっかけ

Raspberry Pi3購入時にスペックを調べていた時、
ふと見知らぬ単語を発見した

Raspbianは標準でソフトリアルタイムシステムとなっているが、ユーザーモードプログラムでのハードリアルタイム性能が必要な場合はlinux-rtカーネルを入れることもできる^{[35][36]}。また、**ベアメタル（OS無し）**で使用することも可能^[37]。

引用：

https://ja.wikipedia.org/wiki/Raspberry_Pi

組み込み屋として知らないはまずいな。。。

ベアメタルとは

- ベアメタル(bare metal or bare machine)とはOSを搭載せず、直接ハードウェアにアクセスするコンピュータを指す
- Non-OSと呼んだりもする

(所感：もとよりOSを搭載しないマイコン環境などでは呼ばない印象)

起動までの流れ

- 電源(USB) – ON
↓
- 1stブート : FAT32 SDカードのbootパーティションをマウント
↓
- 2ndブート : SDカード内のbootcode.binがGPUファームウェアを起動
↓
- start.elf(GPUファームウェア)が設定ファイル(fixup.dat)に基づきCPU/GPUが使うSDRAM領域を区切る
↓
- config.txtに記載されたイメージ(kernel.imgなど)を起動する



ココが開発対象

開発環境

- GNU ARM(gcc-arm-none-eabi)

Raspberry Pi Zero向けのイメージファイル作成例

```
# Cをコンパイル
```

```
$ arm-none-eabi-gcc -mcpu=arm1176jzf-s -mfpv=vfp -mfloat-abi=hard  
-mtune=arm1176jzf-s -nostdlib -ffreestanding -c -o main.o main.c
```

```
# ブート(レジスタ初期化と、main関数まで飛ぶコード)をアセンブル
```

```
$ arm-none-eabi-as -o boot.o boot.s
```

```
# オブジェクトファイルとリンカスクリプトからelf生成
```

```
$ arm-none-eabi-ld boot.o main.o -T link.ld -o app.elf
```

```
# elfから生の実行部分のみ取り出す
```

```
$ arm-none-eabi-objcopy app.elf -O binary kernel.img
```

類似マイコンとの比較

	Raspberry Pi		mbed	Arduino
	3	Zero	LPC1768	Uno R3
価格(円)	4,500	1,200	5,940	3,240
CPUスペック	ARM Cortex-A53 (1.2GHz)	ARM11 (1GHz)	ARM Cortex-M3 (96MHz)	ATmega328P (16MHz)
ROM/RAM	microSD/ 1GB	microSD/ 512MB	512KB/ 32KB	32KB/ 2KB
電源	1.4A Wifi: 2.5A	140mA	120mA	50mA
ピン数	40	40	40	Dig:14 + Ana:6

そこそこのスペックで、低価格。ホビー用に最適

まとめ

- ちょっとmbedをデイスって見たかった



まとめ

- Good

- 安価・手軽にハード環境が手に入る
- そこそこ情報が豊富にある
(マイコン用途がメインのmbedやArduinoに比べたらベアメタルの情報少なめ...)
- 飽きたら、Linuxサーバーとして運用など幅広い用途に使える
- これからはARMの時代

- Bad

- GPU Coreのファームウェア部分がオープンでない
- 保証とかがないため、クリティカルな仕事では使用できない
- Raspberry Pi Zero (W)は日本での供給が少なく手に入りづらい
(公式HPでは、お一人様1個まで)