

---

# LEGO SPIKE Prime向けソフトウェアプラットフォーム SPIKE-RT の紹介

2024/06/22(土)

@ETロボコン向け TOPPERS活用セミナープログラム

朱 義文(TOPPERS/名古屋大学)

# LEGO Education SPIKE Prime

- ここでは、SPIKE Primeと呼ぶ。
- 「小学校高学年から中高生向けのSTEAM学習セット」 ([1]より引用)
- ET ロボコン[2]などのロボットコンテストでも活用。



画像は, [1] より引用

- SPIKE Prime Hub
  - プログラム可能
    - SoC:STM32F413VG
    - Arm Cortex-M4
    - 最大 100 MHz
    - RAM : 320 KB, ROM : 1 MB
  - 6 つ I/O ポート (PUPデバイス接続用)
  - 32 MB の外部記憶領域 (フラッシュメモリ)
  - 5x5 LED マトリックス表示器
  - IMU (加速度センサジャイロスコープ)
  - 3 つの制御用ボタン (1つはライトを含む)
  - スピーカ
  - USB
  - Bluetooth LE
- PUP(Powered Up) デバイス
  - 超音波センサ
  - カラーセンサ
  - フォースセンサ
  - サーボモータ

[1] <https://education.lego.com/ja-jp/products/-spike-/45678#spike%E3%83%97%E3%83%A9%E3%82%A4%E3%83%A0>

[2] <https://www.etrobo.jp/>

# SPIKE-RT

- 概要
  - LEGO SPIKE Prime 向けの RTOS ベースのソフトウェアプラットフォーム (SPF).
  - SPF... アプリケーションを開発する上で基盤となるソフトウェア
  - EV3RTの後継プロジェクト
- 特徴
  - TOPPERS/ASP3 RTOS (次スライドで説明)
  - C 言語プログラミング環境
  - MIT ライセンス
- 開発状態
  - デバイス制御の対応: APIのほとんどが対応済み.
  - 他のコンピュータからSPIKE Primeを制御するための基盤が完成.
  - SPIKE Prime単体でのアプリケーション開発
    - 可能だが, 開発のしやすさに改善の余地あり
- ETロボコンで, 使用可能。
- その他
  - GitHub: <https://github.com/spike-rt/spike-rt>
  - Slack(日本語): [https://join.slack.com/t/spike-rt/shared\\_invite/zt-1lzsqg9zm-\\_C3sPe\\_hFsXAWERdSVXKjg](https://join.slack.com/t/spike-rt/shared_invite/zt-1lzsqg9zm-_C3sPe_hFsXAWERdSVXKjg)



Slack

# TOPPERS/ASP3

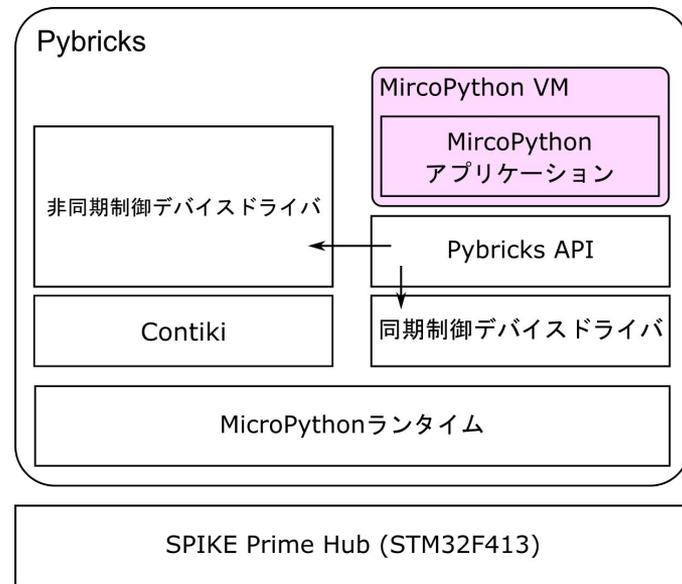
---

- 概要
  - 高い信頼性・安全性・リアルタイム性を要求される組込みシステムを対象としたリアルタイム OS.
  - リアルタイム性を維持しつつ, 複雑なアプリケーションの開発を支援.
  - $\mu$ ITRON 4.0 拡張仕様.
- マルチタスク機能
  - プログラムの実行単位をタスクと呼ぶ.
  - デフォルトで 16 段階の優先度ベース・スケジューリング.
  - プリエンプティブ・マルチタスク.

[1] <https://www.toppers.jp/asp3-kernel.html>

# 既存のソフトウェアプラットフォーム

- LEGO Education SPIKE App
  - 公式のSPF
  - 特徴
    - MicroPython や Scratch によるプログラミング環境を提供.
    - MicroPython ランタイムがベースの模様.
    - (公式仕様書[1]に「 Embedded MicroPython operating system」とある)
- Pybricks[2]
  - LEGOのコンピュータ向けOSSのSPF.
  - 特徴
    - MicroPython によるプログラミング環境を提供.
    - MicroPythonランタイムが一番下で動作.



[1]

<https://education.lego.com/ja-jp/product-resources/spike-prime/%E3%83%80%E3%82%A6%E3%83%B3%E3%83%AD%E3%83%BC%E3%83%89/%E6%8A%80%E8%A1%93%E8%A6%81%E4%BB%B6>

[2] <https://pybricks.com/>

# 既存のソフトウェアプラットフォームの問題点

---

- どちらも MicroPython ランタイムがベース.
  - (1) 他の言語に対応するのが難しい.
    - C言語で記述された既存のソフトウェア資産をアプリケーションで再利用することができない
  - (2) リアルタイム性の要求されるアプリケーションには不向き.
    - ガーベジコレクションなどの影響.

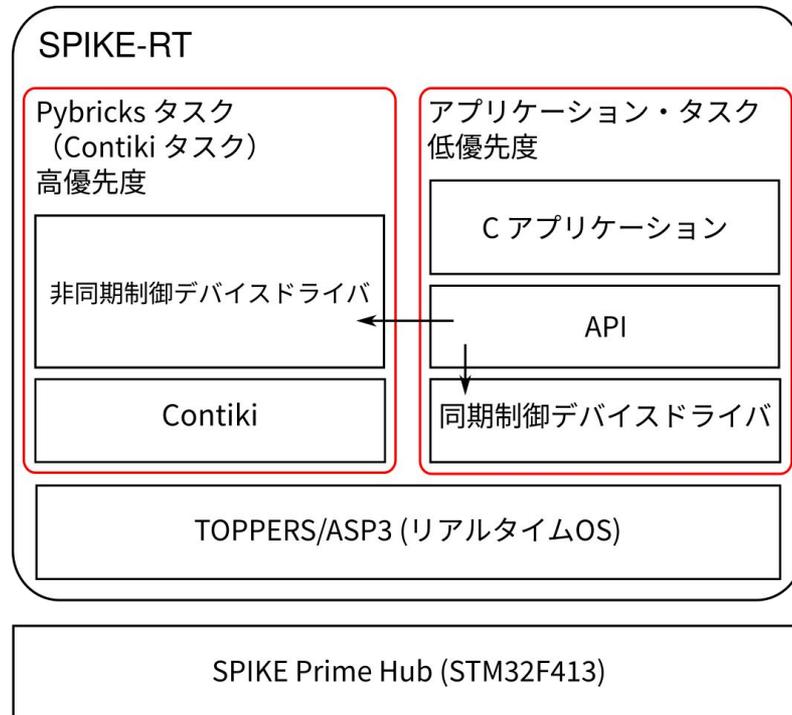
# 目標

---

C言語によるリアルタイムプログラミング環境の提供

# SPIKE-RTの内部実装

- デバイス・ドライバの実装コストの問題.
- →Pybricks のデバイス・ドライバを再利用.
  - Contiki(カーネル)をASP3タスク上に移植.
    - 実は難しくない.
  - Contikiプロセスとして実装されているデバイス・ドライバをほぼ無修正で動作させる.



# デバイス対応状況

Hub 内蔵デバイス	動作	API対応
マトリックスLED	○	○
単一 LED	○	○
ボタン	○	○
加速度センサ/ ジャイロスコープ	○	○
スピーカ	○	○
外部フラッシュメモリ	X	X
バッテリー充電	○(※1)	X(※1)
USB	CDC ACM	Newlib
Bluetooth	Nordic UART	Newlib

PUPデバイス	動作	API対応
カラーセンサ	○	○
超音波センサ	○	○
フォースセンサ	○	○
モータ	○	○

※1 ... バッテリー充電は自動で行われる

ほとんどが対応済み。

# SPIKE-RTのユーザ・インタフェース

---

- API
  - TOPPERS/ASP3 API
  - C標準ライブラリNewlib
  - SPIKE API
    - アプリケーションからSPIKE Pimeのデバイスを制御するためのAPI
- シリアル通信
  - PUPポートF(USART)
  - USB仮想COMポート
  - Bluetoothシリアル
    - いずれもNewlib(i.e. fprintf()/fscanf())経由で入出力可能
- 開発環境: コンテナイメージを提供
- コンパイラ: GCC
- ビルドシステム: GNU Make
- プログラムの書き込み : USB DFUを使用.

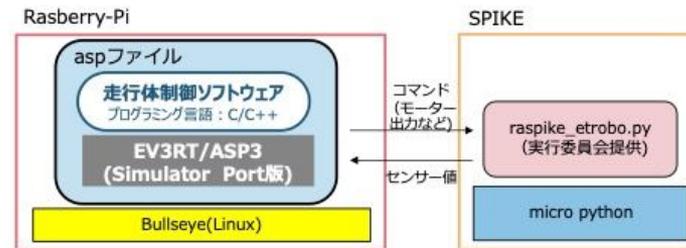
# アプリケーション

---

- SPIKE Prime単体で動作するアプリケーション
  - サンプル(モータ, LED, ボタン): <https://github.com/spike-rt/spike-rt/tree/main/sample/motor>
  - 樋山氏にSPIKE-RTを遊んで頂いた際のコード: <https://github.com/Hiyama1026/spike-rt-sample>
- 他のコンピュータ(Raspberry Piなど)からSPIKE Primeを用いたアプリケーションを作るための基盤
  - micro-ROS
    - SPIKE-RTに移植
    - <https://micro.ros.org/>
    - [https://github.com/exshonda/micro-ROS\\_ASP3](https://github.com/exshonda/micro-ROS_ASP3)
  - RasPikeのSPIKE-RT移植版(次スライドで説明)

# 活用事例1/2: RasPike における問題の解決

- RasPike[1]
  - SPIKE Prime のデバイスを Raspberry Pi から利用するためのアプリケーション.
  - LEGO Education SPIKE App を使用.
  - ETロボコンで使用
- 問題
  - UART 通信においてデータの取りこぼしが発生.
  - SPIKE Prime Hub 側のファームウェアが問題の模様.
- 解決
  - SPIKE Prime Hub 側のファームウェアをSPIKE-RTに移植[2].
  - 通信の取りこぼしが改善(解決?).
  - 応答性が改善.



Copyright(c) ETロボコン実行委員会 All rights reserved.

画像は, [1] より引用

[1] <https://github.com/ETrobocon/RasPike>

[2] [https://github.com/Hiyama1026/spike-rt-sample/tree/master/raspike\\_spikert](https://github.com/Hiyama1026/spike-rt-sample/tree/master/raspike_spikert)

# 活用事例2/2: 学習教材への活用

## 株式会社アフレルより SPIKE-RTを用いた学習教材が発売[1]

2023年1月10日 SPIKE™ キャンペーン 製品サービス

世界初、レゴ® SPIKE™ プライムC言語学習教材を1月に新発売!

株式会社アフレル（福井県福井市、代表取締役社長 小林靖英、以降「アフレル」）は、レゴ® エデュケーション教材のSPIKE™ プライム（以下、SPIKE プライム※2）をC言語※1で動かす「SPIKE™ プライム C言語 プログラミングブック」の販売を開始します。本商品は1月10日より発売し、3月上旬より順次出荷を開始します。



画像は[1]より引用

[1] <https://afrel.co.jp/information/press/product-support/53774/>

# 論文

---

## 研究会報告

- 朱 義文, 李 奕驍, 松原 豊, 本田 晋也: SPIKE-RT: LEGO SPIKE Prime向けリアルタイムソフトウェアプラットフォーム, 情報処理学会研究会研究報告, Vol.2022-EMB-61 No.11, pp.1–6 (2022), <http://id.nii.ac.jp/1001/00222073/>.
  - 内部実装について詳しく説明.

# その他のリソース

---

- テストアプリケーション
  - 朱が、全機能を動作確認するのに使用
  - <https://github.com/spike-rt/spike-rt/tree/main/test>
  - <https://github.com/spike-rt/spike-rt/blob/main/docs/ja/Test.md>
  - CI(自動ビルド)による最新コミットのバイナリが取得可能
    - <https://github.com/spike-rt/spike-rt/actions/runs/9592801470#artifacts>
    - <https://github.com/spike-rt/spike-rt/actions/runs/9592801470/artifacts/1619534437>
- ドキュメント
  - ビルド手順などについて説明
  - <https://github.com/spike-rt/spike-rt/tree/main/docs/ja>
- DoxygenによるAPIドキュメント
  - <https://spike-rt.github.io/spike-rt/ja/html/modules.html>

# APIを少しだけ紹介

---

## 超音波センサのAPI

```
pup_device_t *eyes;  
int distance;  
  
eyes = pup_ultrasonic_sensor_get_device(PBIO_PORT_ID_A);  
  
distance = pup_ultrasonic_sensor_distance(eyes);
```

# APIを少しだけ紹介

---

## Newlibによるシリアル通信

```
extern FILE* serial_open_newlib_file(ID portid);  
  
FILE *fd = serial_open_newlib_file(SIO_UART_F_PORTIDD);  
retval = fprintf(fd, "FPRINTF TEST to SERIAL!!!\n");
```

# テストアプリケーションのプログラムサイズ

---

- 3割以下の使用量.
- アプリケーションはメモリを十分に使用可能.

	使用量 [KB]	搭載量 [KB]
ROM	228	1024
RAM	88	320

# 最近のアップデート

---

## マイナーな変更

- v0.1.1 (2023年8月)
  - <https://github.com/spike-rt/spike-rt/releases/tag/v0.1.1>
  - Bluetoothのコネクションを制御するAPIを追加(ユーザが意識する必要はない)
- v0.2.0 (2024年6月)
  - <https://github.com/spike-rt/spike-rt/releases/tag/v0.2.0>
  - TOPPERS/ASP3カーネルを3.5.0から3.7.0に更新
    - コンテナを使用しない場合の環境構築が楽に
    - (ruby 2.7.xの代わりにruby 3.x.xが使用可能になったため)
  - コンテナイメージ内のソフトウェアを更新
  - ビルドコマンドの簡略化

# 今後の展開(1/2)

---

- 一年ほど、開発が停滞
  - 既に他のコンピュータから制御するための基盤はできている
  - SPIKE Prime単体でのアプリケーション開発にどれだけの需要があるのか？
- WRO(World Robot Olympiad)などでの需要や、ユーザの直接の要望
  - 機能追加・改善のモチベーション

[1] WRO Japan 公式サイト <https://www.wroj.org/>

# 今後の展開(2/2)

---

## 改良・機能追加のアイデア

- APIの改善
- ビルドシステムの改善
  - CMake対応
- ドキュメントの改善
- サンプルアプリケーションの拡充
- USB複合デバイスの実装
  - TinyUSBの移植
  - 複数のUSB仮想COMポートの実現など
- 外部フラッシュメモリの対応
- 他のプログラミング言語の対応
  - LLVMランタイム対応
  - WebAssembly 対応
    - WebAssembly Micro Runtime (WAMR) の移植。
    - wit-bindgenによって、複数言語のAPI(バインディング)の自動生成。
- etc.

# まとめ

---

- SPIKE Prime 向けソフトウェアプラットフォーム SPIKE-RT を紹介.
  - C 言語で制御アプリケーションを開発可能.
  - リアルタイム性が向上.

以上