

TOPPERS活用アイデア・アプリケーション開発 コンテスト

部門 : アプリケーション開発部門
作品のタイトル : NUCLEO F767ZI用mruby版sample1
作成者 : 山根 ゆりえ(yamanekko)
共同作業 :
対象者 : RTOS初心者、Rubyユーザー
使用する開発成果物 : ASP3 NUCLEO F401RE 簡易パッケージ (target_user.txtに従いF767ZI用に変更して使用)

目的・狙い

これまで、mrubyをTOPPERS上で動かす方法として、EV3RT (LEGO Mindstorms) や mruby on GR-PEACH+TECS があったが、ハードウェアが高価で気軽に買って試せるものではなく、もっと安価なボードで動くようにしてほしいという要望があった。また、これらはmrubyから利用できるTOPPERSのAPIが限られているようである。

そこで、3千円程度で購入できるNUCLEO F767ZIを使用し、sample1で利用されているAPIを使えるようにした。

RubyからTOPPERSのAPIを操作できるようになったため、RubyユーザーもTOPPERSのOS機能を利用したアプリケーションを作成することができる。

注) mruby on GR-PEACH+TECS(<https://tecs-docs.readthedocs.io/ja/latest/mruby-on-gr-peach+tecs/rtos.html>) mruby on ev3rt+tecs(<https://tecs-docs.readthedocs.io/ja/latest/mruby-on-ev3rt+tecs/rtos.html>) どちらもTOPPERS のAPIを動作させる目的で作られてはいないようである。

アイデア/アプリケーションの概要

TOPPERSを知る人の間では固有名詞のように語られている、あの「sample1」をmrubyで実装したものである。

sample1の概要は以下のとおりである。(sample1.cより引用)

「ユーザインタフェースを受け持つメインタスク (MAIN_TASK) と、3つの並行実行されるタスク (TASK1~TASK3) , 例外処理タスク (EXC_TASK) の5つのタスクを用いる。これらの他に、システムログタスクが動作する。また、周期ハンドラ、アラームハンドラ、割込みサービスルーチン、CPU例外ハンドラをそれぞれ1つ用いる。」

この中から、メインタスクと並列実行されるタスクをmrubyで実装している。

(周期ハンドラやアラームハンドラの処理もmrubyで書くことはできるが、後述のサイズの課題があるため、本件では実装していない)

本アプリケーションを実現するための課題

1. mrubyのRAM使用量

mrubyのTOPPERS上での実行は、タスクの中からmrubyを起動する方法をとっているため、タスクごとに別のmruby(mruby VM)が実行することになる。

これまでのmrubyではチューニングすることにより、RAMが128KBのボードで動作できるようになっていた。(NUCLEO F401REではNGだが、F411REでは起動できる)

その後、RAMの使用量を50KB程度まで減らすことができたが、現時点では、1タスクあたり50KBのRAMが必要である。

2. TOPPERS ASP3に対応しているNUCLEOボード

ASP3はNUCLEO F401RE用簡易パッケージが公開されているが、F401REではタスクを複数動かすにはRAMが少なすぎる。

F767ZIにポーティングすることにより、RAMが512KBに増えるため、mrubyが1タスクあたり50KB必要でも、RAM容量的には、sample1の実現は可能になった。

インストールから実行までの手順

NUCLEO-F767ZIでTOPPERS/ASP3とmrubyを動かす(<https://qiita.com/yamanekko/items/f50223b67b9ea881a2b5>)に記載している。

実行したときのシリアル出力の様子

起動時

```
TOPPERS/ASP3 Kernel Release 3.5.0 for NUCLEO(STM32F767ZI:ARM Cortex-M7) (Sep  7 2020, 02:13:56)
Copyright (C) 2000-2003 by Embedded and Real-Time Systems Laboratory
                        Toyohashi Univ. of Technology, JAPAN
Copyright (C) 2004-2019 by Embedded and Real-Time Systems Laboratory
                        Graduate School of Information Science, Nagoya Univ., JAPAN
```

```
System logging task is started.
Sample program starts (exinf = 0).
used:48728
### 0907 02:10 main task initialize: 2
### main task task1 active
C: task1 start
```

```
used:98144
###-- task1 active
```

```
task1 is running 1 : . |
task1 is running 2 : . |
task1 is running 3 : . |
task1 is running 4 : . |
task1 is running 5 : . |
task1 is running 6 : . |
task1 is running 7 : . |
task1 is running 8 : . |
task1 is running 9 : . |
task1 is running 10 : . |
task1 is running 11 : . |
task1 is running 12 : . |
```

slp_tskしたあと、wup_tsk
カウンターは連続している

```
task1 is running 23 : . |
task1 is running 24 : . |
read: 115
main:esSdyYzZ
que1: 115
task1 is running 25 : 115. |
task1:s slp_tsk
read: 119
main: w
task1 is running 26 : . |
task1 is running 27 : . |
task1 is running 28 : . |
task1 is running 29 : . |
task1 is running 30 : . |
```

ext_tskしたあと、act_tsk
カウンターがリセットされている

```
task1 is running 107 : . |
read: 101
main:esSdyYzZ
que1: 101
task1 is running 108 : 101. |
task1:e ext_tsk
read: 97
main: a :2
C: task1 start

used:147032
###-- task1 active

task1 is running 1 : . |
task1 is running 2 : . |
task1 is running 3 : . |
task1 is running 4 : . |
```

タスク1～タスク3にdly_tskを送る

(タスク1のみ稼働している。タスク2、3はキューのみ存在している)

```
task1 is running 42 : . |
read: 100
main:esSdyYzZ
que1: 100
task1 is running 43 : 100. |
task1:d dly_tsk(5000000)
task1 is running 44 : . |
task1 is running 45 : . |
task1 is running 46 : . |
task1 is running 47 : . |
task1 is running 48 : . |
read: 50
main: 2
task1 is running 49 : . |
task1 is running 50 : . |
read: 100
main:esSdyYzZ
que2: 100
task1 is running 51 : . |
task1 is running 52 : . |
task1 is running 53 : . |
task1 is running 54 : . |
read: 51
main: 3
task1 is running 55 : . |
task1 is running 56 : . |
read: 100
main:esSdyYzZ
que3: 100
task1 is running 57 : . |
task1 is running 58 : . |
task1 is running 59 : . |
```

100はd,
「read:100」はシリアル経由でdが
送信された
「que1: 100」はtask1用のデータキューにdが
セットされたことを示す

sample1を実装した経緯

TOPPRES利用者には馴染みのあるプログラムであり、これを実装することにより、mrubyにも関心を持ってもらえるのではないかと考えた。

TOPPERSについて書かれた資料では、sample1を使用しているものをよく見かけたため、mrubyユーザーもsample1の実装があると、これらの資料を参考にしやすくなるのではと考えた。

sample1を実現するために実装したもの

1. ASP3のNUCLEO F767ZIへのポーティング

ASP3 NUCLEO F401RE 簡易パッケージのtarget_user.txtに従いF767ZI用に変更した。(

<https://github.com/yurie/asp3-f767zi>)

主な変更箇所は以下を参照

<https://github.com/yurie/asp3-f767zi/commits/master>

2.mrubyの変更

OSC Hokkaido2020オンライン(https://dev.toppers.jp/trac_user/contrib/wiki/OSC2020Hokkaido)

で紹介した「NUCLEO F401RE上でmrubyを動かす」を実現するためのmrubyの改造案の一部（シンボル、irepのROM化）が本家に採用された。（2020/09/07現在ではmruby3ブランチ）

3.TOPPRESのAPIをmrubyから利用するためのmrbgemの作成

<https://github.com/yamanekko/mruby-stm-nucleo-toppers>

mrubyから呼べるようにするためのラッパーを作成した。

メソッド名について、RubyではTOPPERSのAPIの名称（3文字の略称）を使用するのは不自然に感じられたため、TOPPERS第三世代統合仕様書 5.10 省略名の元になった英語を参照してつけている。

mrubyからマイクロ秒単位でタスクの制御はしない（させたくない）と思われたこと、大きな数字を扱いたくない、ASPとの互換性などからRubyからの時間指定はミリ秒を使用している。

ただし、fch_hrt()相当のメソッドはマイクロ秒を返す。

sample1.cで使用されていたSVC_PERRORが便利だったため、現状は無条件につけてTOPPERSのAPIを呼ぶようにしている。（将来的には選択できるようにする予定）

4. sample1のmruby版の実装

C言語版をmruby版にポーティングした。そのままで置き換えられない部分については変更している。

例えば、メインタスクと各タスクへのシリアルから受信したコマンドの受け渡しにはmruby VM間のデータの受け渡しになるため、グローバル変数は使用できない。したがって、メッセージキューを利用して渡すようにした。

現時点での課題

1. 並行実行するタスクの複数化

単純な処理のプログラムであれば、メインタスクと3つの並行実行されるタスクを起動することができるが、sample1の実装の場合は、並行実行されるタスクを2つにすると例外が発生してしまう。

2. デバッグから転送して実行できないケースがある

STM32CubeIDEのデバッグ機能を利用してプログラムを書き込んだ場合、mrubyがバイトコードを読み込んだ際にcrcチェックでエラーになり実行できないが、リセットボタンを押して再起動すれば実行できる。もしくはasp.binを作成してUSB経由でコピーすると実行できるという課題がある。コード量が増えると発生が顕著になるように感じている。これについては、C言語で書いたオンボードの3色LEDを点滅させる、同じプログラムを何度か実行した際、青色のみ点滅する場合と点灯しない場合があるため、ハードウェアの問題かもしれない。

これらが解決すれば、並行実行されるタスクは3つの増やすことができる。

また、EV3RTではユーザードメインでmrubyを使ったプログラムを実行していたため、例えばStackを突き抜けるなどしてOSよりアプリが強制終了された場合でも、アプリが終了するのみでOSは動作しているため、不具合の解析がやりやすかった。

NUCLEO用ASP3の場合、mrubyのコードの実行が原因でcore_support.SのCPU例外になる場合、mrubyが例外をあげる前にCPU例外に行ってしまう、デバッグが困難になっているようだ。ここでmruby側の情報を拾うことができればデバッグの効率化に繋がりそうに思う。

謝辞

NUCLEO F767ZI対応に関して、本田 晋也先生には多大なご助言をいただきました。

mrubyの省メモリ化の実現に関して、まつもと ゆきひろさん及びNPO法人 軽量Rubyフォーラムにご支援いただきました。

ダウンロード先リスト

NUCLEO F767ZI用ASP3 <https://github.com/yurie/asp3-f767zi>

TOPPERS/NUCLEO mrbgem <https://github.com/yamanekko/mruby-stm-nucleo-toppers>

sample1用のmrbgem <https://github.com/yamanekko/mruby-toppers-sample1>

sample1アプリケーション <https://github.com/yamanekko/asp3-f767zi-application-samples>

動作方法などは <https://qiita.com/yamanekko/items/f50223b67b9ea881a2b5> を参照してください