

第4章

リアルタイムOS TOPPERS/ASPを利用する

128×128ドット・マトリクスLEDによる掲示版

石岡 之也

adc	flash	multicore
blink	gpio	pico_w
clocks	hello_world	picoboard
cmake	i2c	pio
divider	ide/vscode	pwm
dma	interp	reset

図1 pico-examplesはたくさんある

● ラズベリーパイ財団が提供する豊富なサンプルがある

ラズベリー・パイ Pico (以後, Pico) には, Pico に搭載するマイコン RP2040 上でデバイスの利用を容易にするための pico-sdk^{注1} というサブルーチン群が, ラズベリーパイ財団から提供されています。また, pico-sdk を使ったさらにアプリケーションに近いソフトウェアのサンプルとして pico-examples^{注2} というソフトウェア群も提供されています。この pico-examples には, I²C や SPI を使ったセンサ制御からタイマや GPIO などの利用例が多数収録されています。

● pico-examples にあったマトリクスLED制御のためのサンプルを利用する

筆者の手元にあったマトリクスLEDパネルを制御するためのサンプル・プログラムが pio/hub75 に用意されていたことと, pico-examples を使ったソフトウェアの開発例がほとんどなかったことから, このコードを使って電光掲示版を作ってみようと考えました。

注1: <https://github.com/raspberrypi/pico-sdk>注2: <https://github.com/raspberrypi/pico-examples>

master		pico-examples / pio / hub75 /	
kilogram Cleanup 3rd party samples; update			
..			
📄	CMakeLists.txt		Init
📄	Readme.md		Init
📄	hub75.c		Init
📄	hub75.pio		Cle
📄	mountains_128x64.png		Init
📄	mountains_128x64_rgb565.h		Init

図2 pioディレクトリ配下のhub75

PIOを使ってマトリクスLEDに出力するサンプルhub75

● サンプルに加える改造

pico-examples は, 図1に示したような RP2040 上のデバイスを使ったプログラムが, ソースコードとしてネットから取得できます。今回は pio ディレクトリ配下の hub75 というソースコード (図2) を流用して, 電光掲示版を実現しました。

この hub75 は, ソースコード内の配列データとして埋め込まれた RGB565 形式の画像データを PIO を使って 128×64 ピクセルのマトリクスLEDパネルへ出力するプログラムです。これの画像出力部に関して,

- 出力するピクセル数を縦方向2倍にする
- BMP形式のデータを扱えるようにする
- SDカードからBMP形式のデータをファイルとして読み込む

これら変更を施すことで, 比較的容易に移植できました。