

# TOPPERS 活用アイデア・アプリケーション開発 コンテスト

- 部門 : 活用アイデア部門
- 作品のタイトル : 組み込みシステムのプログラミング基礎演習セット
- 作成者 : 飯島 純一(いいじま じゅんいち)
- 共同作業 : なし
- 対象者 : 組み込みシステムのプログラ開発を志す初級者で、c 言語で数十行のプログラムを書いた経験がある者。具体例としては、以下のような人たち：  
・新入社員教育受講者  
・大学の情報系学科の 2, 3 年次の演習・実験受講者
- 使用する開発成果物 : TOPPERS ASP, BASE PLATFORM, 基礎 1, 2, 3 セミナー

## 目的・狙い

組み込みシステムを題材にした、プログラミングの基礎演習のための教材である。  
「ベアメタル」から始め、(世の中に例題があまり多くない)RTOS の基本機能(API)の初歩的な使い方までを演習する機材セット(例題、演習問題、プログラム検証用ハードウェア)を作成した。コンピュータの基本構造、アルゴリズムやデータ構造を含めて基礎的なプログラミングの学習を展開する構成になっている。プログラムの検証用ハードウェアの一例も開発した。

## アイデア/アプリケーションの概要

教材(現時点では、アイデア)は次のようなものから構成される。

- ・組み込みシステムを学習するための例題と演習問題のセット
- ・演習問題を解くために必要な知識やヒントなどを解説したテキスト
- ・実際にプログラムを作成して検証できるハードウェア(NUCLEO-F767ZI 版)

## 組込みシステムのプログラミング基礎演習セット

## 概要

本作品は、初学者が、組込みシステムのプログラミングの基礎を学ぶための道具一式である。次のようなものから構成されている。

- 例題と演習課題集
- 解説テキスト
- 演習課題の解を実際に実行して、結果を確認するためのハードウェア

例題や演習課題として、組込みシステムやプロセッサに対する理解を深められるような基礎的な話題を選び、解説を試みる。いわゆるベアメタルに対応する話題から始め、RTOS を利用する際の基本的な話題までを取り上げる。特に、RTOS の基本的な API の使い方を、できるだけ各 API ごとに例題を示して詳しく解説することを試みる。

プログラミングの学習には、作成したプログラムを実際に動かして結果を確認することが不可欠である。本作品で取り上げた例題や演習課題に対応できるハードウェアは、初学者用と称して販売されている製品には適当なものが見つからなかった。そこで、一例として、図1のようなシステムを設計・製作した。マイクロプロセッサの基板は、市販の製品 (STMicroelectronics 社、NUCLEO-F767ZI) を利用した。例題や演習課題に対応するハードウェアは、ドーターボードとして実装した。

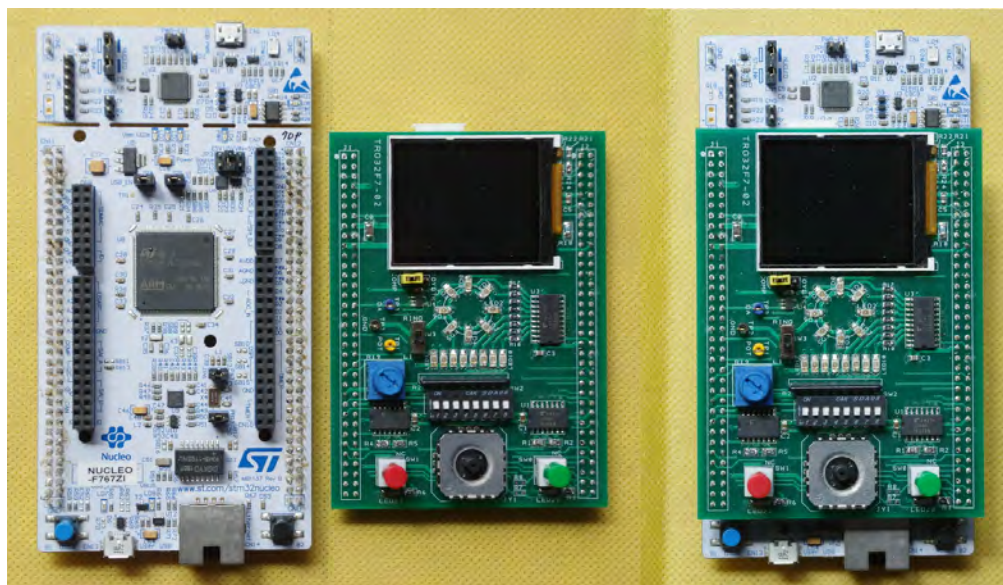


図 1: 演習課題の結果を確認するためのハードウェアの一例 (左) NUCLEO-F767ZI と製作したドーターボード (右) ドーターボードを NUCLEO ボードに載せた状態

このボードには、例題や演習課題のプログラムを実装するのに必要な I/O 装置が備わっている。それらを以下に示す。

- 8 個の LED (直線状とリング状を切り替え可能)
- 2 個の押し釦スイッチ
- 0~3.3 [V] の電圧を生成できる可変抵抗器
- アナログコンパレータ
- 8 素子の DIP スイッチ
- 160 × 128 pixel LCD
- アナログジョイスティック
- メモリーカードソケット

本作品は、2022 年 9 月 11 日時点では、例題と課題のプログラム例の作成と解説の作業中である。

## 制作目的と方針

組込みシステムのプログラミングの学習を通して、次のようなことを修得することが本作品の目的である。

- コンピュータの基本構造の深い理解
- プログラムを洗練 (sophistication) していく流れ
- センサやアクチュエータを扱うプログラムの特徴
- RTOS を使うための基本的な知識

## 通常の PC を利用したプログラミング学習と組込みシステムの学習

通常の PC を利用したプログラミング学習の方針として、次に示すような典型的なものがある。

- c または c++ 言語を使って、文字または簡単な図形を扱いながら学習する
- JAVA 言語を使って、簡単なゲーム作りなどを話題に学習を進める
- Python 言語を使って AI の話題を題材にして学習する

ゲームや AI の話題を取り上げる方針の学習では、かなり大規模で充実したシステムを短時間で構築できたりする。これは、すでに用意された高機能のライブラリを使うことで実現できている。既存のライブラリの使い方の習得に多くの時間をかけることになって、アルゴリズムやデータ構造の基本を学習する時間が短くなりがちである。

一方、多くに人たちが使うライブラリ群を構築したり、オペレーティングシステムなどの基本的なソフトウェアを開発する仕事も存在する。こうした分野の開発者を育てる過程では、コンピュータの基本的な構造、アルゴリズムやデータ構造の基本的な学習を充実する必要がある。

また、コンピュータが多種類の製品に膨大な量が使われている組込みシステムもある。この分野のプログラム開発では、コンピュータ (プロセッサ) の基本的な構造を深く理解することなしには仕事が成立しない。したがって、組込みシステムのプログラミング学習を進めるうえで、コンピュータの基本を構造を学ぶことは必須である。

視点を変えると、コンピュータの基本的な構造を学ぶために、組込みシステムが、おおいに役立つ可能性がありそうである。これは、PC のソフトウェア開発者にとっても、プログラミングの学習の初期段階で、組込みシステムを扱っておくことが有効に働く可能性があることを示唆している。

## 組込みシステムのプログラミングの課題

コンピュータの基本構造やアルゴリズム、データ構造などを学習するとき、どのような話題を取り上げたらよいだろうか。本作品では、いわゆるベアメタルを対象にしたプログラムから学習を始め、RTOS の基礎となる基本操作までを学ぶ方針を考えた。コンピュータの基本構造は、主にベアメタルを対象にした話題で取り上げることにした。RTOS を対象にした話題では、データ構造なども取り上げるようにした。

それぞれの分野で演習問題として取り上げる話題として以下のようなものを選んだ。

### <ベアメタル編の課題>

- LEDの点灯と消灯(いわゆるLチカ)
- スイッチの状態の取得と利用
- 情報のコンピュータ内の表現
- 基本的な通信
- タイマ、カウンタの利用
- 割り込み処理の基本
- 割り込みと割り込み処理の連携
- PWM信号の生成
- D/A変換とA/D変換

### <RTOS編の課題>

- タスクの起動と遅延(LEDの点滅)
- 複数タスクの起動と連携(環状LEDの順次点灯)
- スイッチの割り込み処理(イベントフラグ)
- 流れるウィンカ(セマフォ)
- 割り込み処理とタスクの連携
- プロデューサ・コンシューマ問題

ベアメタル編では、いわゆるLチカから始めることにした。ただし、1個のLEDの点灯・消灯だけでなく、8個のLED群を使った課題を用意した。1バイト(8bit)を同時に扱うことで、コンピュータ内の情報の表現の理解が進むように構成した。

また、「(一部の)変数は実世界の窓」という視点を大切にしたい。組込みシステムで取り扱うセンサやアクチュエータは、関連する変数を介してデータを取得したり、動作を指示したりできる。これは、センサやアクチュエータを構成する電子回路や機械系を、変数という結節点で抽象化を行ったと考えることができるからである。

## 開発環境

開発環境としてCUI(CLI)ベースのMSYS2-MinGW-64bitとgcc、適当なエディタを利用する。これは、TOPPERS教育ワーキンググループが開発した、基礎1, 2, 3セミナーやBASE PLATFORMの開発環境に準拠している。<sup>1</sup>

ベアメタルの演習を始めるときは、プロセッサの初期化や端子機能の選択などのプログラム片は、すでに用意されているものを使用する。演習が適度に進んだ後では、ソースコードを含めて解説を行う。

RTOSの基礎を学ぶ演習では、TOPPERSを利用した。可能な場合には、ベアメタル編で取り上げた話題を、視点を変えて再度取り上げるような構成にした。

## プログラム検証環境

概要でふれたが、これらの課題の回答プログラムを確認するためのハードウェアも製作した(図1)。プロセッサ基板には、STmicro社のNUCLEOシリーズの基板(NUCLEO-F767ZI)を利用した。演習課題用の入出力装置は、ドーターボードとして実装した。

NUCLEOシリーズ基板は、同社のSTM32シリーズのプロセッサの多くに対応した製品群がある。手軽な開発システムとして普及しているArduinoシリーズと共通する入出力端子を持っていて、値段も手ごろなので、使い勝手の良いシステムである。

TOPPERS教育ワーキンググループによって、NUCLEOシリーズに対応したBASE PLATFORMが開発されている。この成果を利用させていただいた。

<sup>1</sup>「製品」を作るためには、ハードウェアを抽象化するAPIやドライバのライブラリを利用することが必要不可欠でEclipseをベースにしたGUI環境などが使われる。しかし、組込みシステムやプロセッサについて学ぶためには、できるだけ裸に近い状態で使ってみる経験を一度は試みることに価値があると製作者は考えている。

## 例題と演習課題一覧

例題と演習課題を列挙する。必要に応じて、各課題を検証するためのハードウェアに対する要求とその理由を示した。

### 例題 1

線状に並んだ LED の一つ (例えば右端) を点滅するプログラムを示す (図 2 参照)。点滅の時間間隔を変化したり、別の LED を点滅するプログラムも示す。

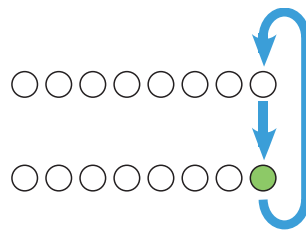


図 2: 線状に配置した LED の一つを点滅

### 課題 1

環状の LED を一つずつ順番に点灯するプログラムを書け。  
具体的には、図 3 に示すように、一つの LED を少しの時間点灯し、それを消灯すると同時に隣の LED を点灯することを、反時計回りに繰り返す。

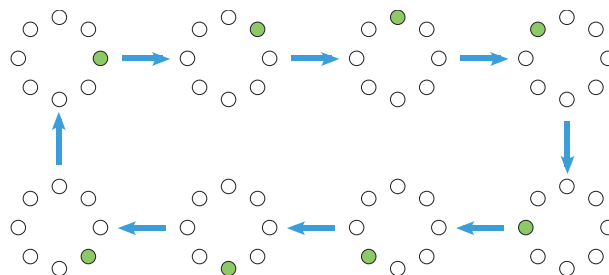


図 3: 環状に配置した LED の例

### 検討: LED の個数

多くの学習プログラムが、いわゆる「Lチカ」から始めるのは次のような理由による。

- 入力より出力のほうが扱いやすく、結果も理解しやすい
- LED の点滅という明確で達成感のある結果が得られる
- コンピュータの処理には時間がかかることが理解できる

LED を一つ点滅するプログラムは、出発点としては良いのだが、プロセッサの実際の入出力は 8 ビット幅で行われることが多いことを考えると、複数ビットに対応した LED を取り扱うことが望ましい。また、整数や文字などの内部表現を表示できるようにするためにも最低 8 個の LED を準備できると良い。

直線状の LED 群とリング状の LED 群を、スイッチで切り替えられるようにしておくと、点滅のパタンのバリエーションを増すことができる。

## 課題 2

(a) 赤または緑のスイッチを押し下げたとき、対応する赤または緑の LED を点灯するプログラムを書け。

(b) 緑のスイッチを押し下げたとき、対応する緑の LED が消灯していれば点灯し、点灯していれば消灯する (点灯と消灯を交互に繰り返す) プログラムを書け。

ポーリングで、スイッチの状態変化を検出する課題である。

## 課題 3

緑のスイッチを何回か押し下げた後、赤のスイッチを押し下げる。赤のスイッチを押し下げたとき、それまでに緑のスイッチを押し下げた回数を、線状の LED に表示するプログラムを書け。

コンピュータの内部表現を実体験する課題として設定した。赤のスイッチを押し下げたときに、緑のスイッチを押し下げた回数をリセットしても良いし、しなくても良い。

### 検討: スwitchの個数

プログラムで入出力可能なスイッチは、最低 2 個あることが望ましい。ポーリングで、スイッチの状態変化を検出するような場合に、複数のスイッチに対応するのは、一回り無図解医科大になる。これ以降でも 2 個のスイッチを利用した課題を提示する。

また、最初にスイッチを扱う段階では、回路でチャタリングを除去したほうが良いと考えている。

## 例題

シリアル通信回線を使って、一文字を出力するサブルーティン (関数) を提示する。

## 課題 4

緑のスイッチを押し下げたとき、それまでに白のスイッチを押し下げた回数をシリアル通信によって PC のウィンドウに表示するプログラムを書け。

コンピュータの内部表現で記録された整数値を、対応する十進数表記の文字 (数字) 列に変換して出力する課題である。c 言語の標準ライブラリ `i2a()` を作ることに相当する。

## 課題 5

キーボードから十進数を表す文字列を入力し、その数値を二進数表現で線状の LED に表示するプログラムを書け。

十進数を表している文字 (数字) 列を、対応する内部表現の 2 進数に変換する。  
c 言語の標準ライブラリ `a2i()` を作成することに相当する。

## 課題 6

キーボードから十進数を表す文字列を 2 組入力し、その 2 つの数値の和を求めて、PC に表示プログラムを書け。

標準ライブラリ `a2i()` と `i2a()` に相当する関数を作り、それらを使ってプログラムを構成する課題である。

## 例題

スイッチを押し下げたときに割り込みを生ずるようにプロセッサの入力端子の設定をし、割り込み処理プログラムの書き方を提示する。

## 課題 7

(a) 緑のスイッチを押し下げたとき、対応する緑の LED が消灯していれば点灯し、点灯していれば消灯する (点灯と消灯を交互に繰り返す) プログラムを割り込みを利用して書け。また、白のスイッチを押し下げたときは、赤の LED が同様に反応するようにプログラムを拡張せよ。

(b) 課題 1 で作成した、環状の LED を順番に点灯するプログラムと緑と白のスイッチを押し下げたときに対応する LED を点滅する処理とを同時に動かすようにプログラムを修正せよ。

(b) の課題は、「メイン」プログラムと割り込み処理を連携する課題である。この段階では、深い考察はせずに、メインプログラムで使っている変数の値を強制的に変更する割り込み処理を行えばよしとする。

## 例題

タイマの使い方の例を提示する。

## 課題 8

緑のスイッチを押して、次に白のスイッチを押す。このとき、2 つのスイッチを押した時間間隔に応じて、LED の表示を変化させるプログラムを書きなさい。このとき、時間の計測には、コンペアマッチタイマの割り込みを利用しなさい。

### 例題

タイマを利用して PWM 信号を生成し、LED の明るさを変えるプログラムを示す。

### 課題 9

キーボードから入力した値によって、LED の明るさを変えるプログラムを作成せよ。LED の明るさを変える手段として、タイマを利用して、PWM 信号のデューティ比を変える方法を使え。

### 例題

D/A 変換器を使って、0 [V] ~ 3.3 [V] の電圧を出力するプログラムで電圧を生成し、テスタを用いて電圧を測定してみる。  
ポテンショメータのつまみを回して、電圧が変化するのがテスタで確認する。鋸歯状波を生成するプログラムを動かして、コンパレータ回路の動作を確認する。(鋸歯状波の生成は、D/A 変換器のデジタル入力を 0 から最大値まで 1 ずつ変化することで実現できる。)

### 課題 10

(a) 測定対象の電圧と鋸歯状波をコンパレータの二つの入力端子にそれぞれ与え、コンパレータの出力が 0 から 1 に変化するときの鋸歯状波を生成しているデジタル入力の値を線状の LED に示すプログラムを書け。  
(b) 2 分探索の手法を使って、作成した A/D 変換プログラムを高速化せよ。

逐次比較型の A/D 変換器は、2 分探索と同じ手順が使われている。ハードウェアとソフトウェアの類似性に注目したい。

## ここからは、RTOS を使った課題

### 例題

一つの LED を点滅するプログラムをタスクとして実現する。  
時間つぶし型の遅延処理を用いた LED の点滅プログラムをタスクに書き換え、メインタスクから起動する。

### 例題

一つの LED を点滅するプログラムをタスクとして実現する。時間つぶし型の遅延ではなく、RTOS の遅延基本動作を利用する。

遅延の API を使う。こちらが RTOS での基本の方式。  
また、タスクは、ストレートフォワード型で構成することを示す。



### 例題

8 個の線状 LED のうちの適当な二つを交互に点滅するタスク群を構成する。一つの LED に対応するタスクを作り、もう一つの LED に対応したタスクを起動して自らは休止する手順を実現する。

手順の部分は、最初は 2 つのソースコードで書いてみるが、一つのソースコードで実現できることを示す。

### 課題 11

環状の LED を順次点灯するタスク群を構築しなさい。

8 個のタスクを順次起動する形で LED を順次点灯するプログラムを構成する。各タスクの手順の部分は、一つのソースコードで記述できる。各タスクの手順は、隣の LED を点灯するタスクを起動して自身は休止する

タスクが、どの LED と対応しているかは、起動時のパラメタで指定する。

### 例題

赤のスイッチを押したときに、スイッチに対応する赤の LED を、緑のスイッチを押したときには、緑の LED を点灯するタスク群と割込みハンドラを構成する。

### 課題 12

最近の自動車に搭載されている「流れるウィンカ」のように点滅する線状の LED 群を構成せよ。左側の赤のスイッチを押したときに、左向きに数回流れるように点灯し、右側の緑のスイッチを押したときに、右向きに数回流れるように点灯するタスク群を構成せよ。

赤のスイッチと緑のスイッチをほぼ同時に押したときのプログラムの挙動を推測しよう。

### 課題 13

「流れるウィンカ」で、赤のスイッチ緑のスイッチをほぼ同時に押したときに、先に押した方向に流れるように点灯する構成にせよ。

LED を排他的に点灯するように構成する。このために初期値 1 のセマフォを利用する。

### 課題 14

緑のスイッチを押すと生産が行われ、製品を一つ倉庫の棚に置く。倉庫の棚の数は 8 個と仮定する。赤のスイッチを押すと倉庫の棚から製品を取り出して出荷する。

倉庫の棚の埋まり具合を 8 個の LED の点灯で表示する。

緑と赤のスイッチを、いろいろな順序で押してみる。

## ハードウェア

演習課題のプログラムを作成して、実際に動かして検証するハードウェアを作製した。プロセッサ基板として、STmicro社のNUCLEO-F767ZIボードを利用した。演習用の回路は、ドータボードとして作製した。

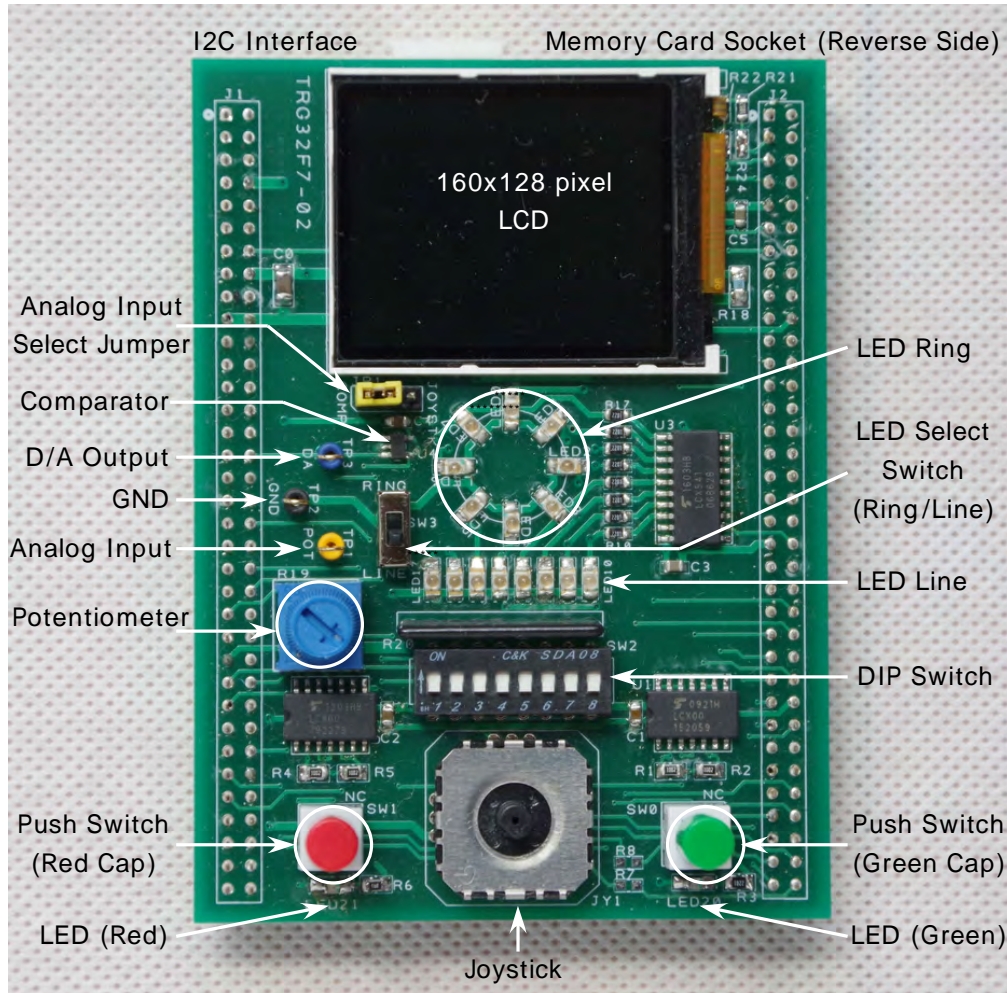


図 4: 演習課題の結果を確認するためのハードウェアの一例

このハードウェアは、演習に必要な次のような入出力装置を備えている。

- 直線状に並んだ 8 個の LED
- 環状に配置した 8 個の LED(直線状の LED 群とスイッチで切り替え可能)
- 2つの押し釦スイッチ (赤と緑のキャップを備えている)  
2つともチャタリング除去回路を備えている  
赤キャップのスイッチは、チャタリング除去回路なしの出力も備える
- 8素子の DIP スイッチ
- D/A 変換器の出力信号を観測するための端子
- 0 ~ 3 [V] の電圧を生成するつまみ付きポテンショメータ (測定用端子付き)
- アナログコンパレータ

また、発展的なプログラムを作るための装置として、次のようなものを備えている。

- 2次元のジョイスティック型ポテンシオメータ
- 160 × 128 ピクセルの液晶表示器 (LCD)(SPI 接続)
- メモリカードコネクタ
- I2C 外部接続用コネクタ

図 4 にそれらの入出力装置の配置を示す。

## 今後の課題

本作品は、大学の学生実験の課題を見直し・拡張したものである。学生実験用のハードウェアは SH 版と RX621 版を作成してきた。この ARM 版は、実際の現場で利用するには、解説テキストなどの洗練が必要だと考えている。

今回、組込みシステムの初級者向けの演習問題を解いて確認をする装置を設計する際に次のような点に留意した。

- プロセッサの選択 (ARM, RISC-V, RX, etc)
- メモリ容量
- クロック周波数
- プロセッサボードの入手性または製作可能性

プロセッサとして ARM を選択した時、利用できるプロセッサ基板は STMicroelectronics 社製のものが (価格を含めて) もっとも入手性がよい。今回は、演習問題に対応する入出力回路を実現できるプロセッサ基板として、NUCLEO-F767ZI を選択した。8 個の LED や 8 素子の DIP スイッチの入出力を実現するための端子の機能の配分を考えた結果である。

しかし、STmicro の STM32 シリーズのプロセッサは、汎用入出力レジスタの構成やそれぞれのレジスタと端子との対応が独特で、例題のプログラムの分かりにくさを生み出してしまうことにつながってしまった。分かりやすい例題を書くためには、同じ ARM 系でも別のシリーズを利用することも考えたい。もっとも、(本作品を制作した動機は、どちらかと言えば自己満足なので、) どなたかに学習用の教材として使っていただく機会があればの話であるが。

## 参考資料