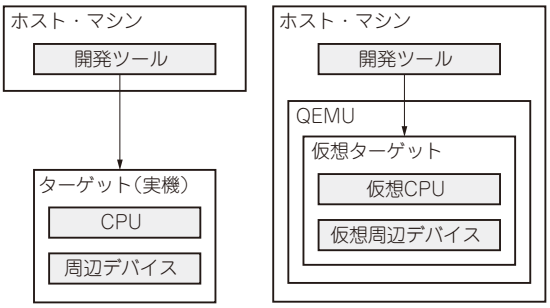
第1章 入手できるSTM32F407G-DISC1を例に仮想から実機評価まで

ダウンロード・データあります

エミュレータQEMUを活用した開発の手引き

菅原 政義



(a) 実機を使用する場合　(b) QEMUを使用する場合

図1　クロス開発(実機とQEMUの比較)

1. はじめに

本稿では，エミュレータQEMUを活用してCortex-M4マイコンのアプリケーションを開発し，QEMUの特徴やQEMUを使用した開発環境の構築手順を紹介します．QEMUを使用すれば，マイコン・ボードを用意しなくてもアプリケーションの開発を始めることができます．

QEMUは，Cortex-M3/M4の対応が強化されたxPack QEMU Armというものを使用します．評価ボードSTM32F4-Discovery(STマイクロエレクトロニクス)をターゲットに，LEDを利用したルーレット・ゲームを作成し，QEMUの仮想ボードと，実機のそれぞれで動作を確認します．

2. QEMUの概要

● 概要と特徴

QEMUは，オープンソースの仮想マシン・エミュレータです．そもそも仮想マシンとは，マシン(CPUや周辺デバイス)の動作をエミュレーションするソフトウェアのことです．ホスト・コンピュータ上で，別のアーキテクチャの仮想マシンを動作させたり，複数の仮想マシンを動作させたりすることが可能です．

QEMUは，さまざまなアーキテクチャ(Arm，RISC-V，RX，x86など)のエミュレートに対応しています．また，QEMUを実行するホスト環境は，Windows，macOS，Linuxなどがサポートされています．QEMUの特徴について，以下に紹介します．

▶利点1：実機が不要

組み込み開発では，開発環境と実行環境が異なるクロス開発が一般的です(**図1**)．実機を使用する場合は，開発用のホスト・マシンと実行用のターゲット・マシンを用意する必要があります．一方，QEMUを使用する場合は，ホスト・マシン上でQEMUを動かし，仮想ターゲット・マシンをエミュレートできます．そのため，実機がなくてもアプリケーションを開発・実行できます．

▶利点2：マシンの振る舞いを変更できる

QEMUは，ソフトウェアでマシンをエミュレートしています．そのためQEMU側をカスタマイズすれば，マシンの振る舞いを変更できます．例えば，意図的にハードウェアが異常な状態などを作り出すことも可能です．このようにわざと障害や誤りを注入するテスト手法を，フォールト・インジェクションと言います．実機では確認が難しいような異常系のテストで利用できます．フォールト・インジェクションの利用例については，「6. 応用例：QEMUをカスタマイズする」で解説します．

▶制約：周辺デバイスのエミュレート

QEMUでは，CPUに加えてマイコン基板レベルのエミュレートできますが，未実装の機能もあります．全てを完全にエミュレートできる訳ではないのが現状です．

QEMUを実機の完全な代替とするのは難しいですが，CPUに着目した動作確認や特定機能のテスト，OSの評価などの目的での活用が期待できます．

対応状況はターゲット・マシンによって異なりますが，例えばSTM32ボードの場合は以下に対応しています．

- CPU
  Arm Cortex-M3，Cortex-M4，SysTick，NVIC
- 周辺デバイス