第3章 ルネサスAIツール e-AI Translatorを使って Arduino UNO R4，RL78/G23，ラズパイPicoへ移植

ステップ③：軽量化した学習済みモデルをマイコンで動かす

関本 健太郎

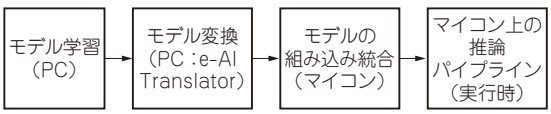

図1　マイコンによる推論処理の流れ

第1章では，ラズベリー・パイ5を用いて加速度センサの時系列データを分析し，物体の静止状態をモニタリングするとともに，静止状態からの急激な変化を異常として検知しました．本章では，加速度センサの計測とオートエンコーダによる推論処理をマイコンで行う手順について解説します．対象とするマイコンは，RA4M1，RL78/G23（いずれもルネサス エレクトロニクス），およびラズベリー・パイPico（以降，Pico）です．

● マイコンの推論処理の流れ

一般に，マイコンによる推論処理は**図1**のように分解できます．各処理では次のことを行います．

▶モデル学習

- PC上でモデルを学習（例：オートエンコーダ/分類器）
- 推論用にエクスポート（一般にONNX/TF Liteなど）

▶モデル変換

- モデルを読み込み，入出力形状/前処理（正規化/スケーリング）/量子化方式（多くはINT8）を設定
- 変換を実行→マイコン用のCヘッダ・ファイル/ソースコード（重み/ネットワーク定義/API）と小さなランタイムが生成される

▶モデルの組み込み統合

- 生成物をプロジェクトへ組み込み，センサ制御/バッファ管理/推論呼び出し/後段ロジックを実装
- 重みはフラッシュ・メモリ常駐（const int8_tなど），作業バッファはRAM．割り込み/リアルタイムOSやメイン・ループで周期実行

▶マイコン上の推論パイプライン

- 1，センサ取得
- 2，前処理
- 3，量子化（INT8想定）
- 4，推論
- 5，後処理
- 6，判定
- 7，監視/チューニング

モデルの推論には通常，多量の浮動小数点計算が必要になりますが，計算量を軽減するためにモデルの計算をINT8形式で行うことで，処理を大幅に削減できます．

マイコンにAIを搭載するための開発環境「e-AI Translator」

本章では，この最適化を支援するツールとして，e-AI Translator（ルネサス エレクトロニクス）を使用します．e-AI Translatorは，PC上で学習したAIモデルを，マイコンなどリソース制約のある組み込み環境へ移植/実行することを支援します．

● 処理フローとシステムの全体構成

e-AI Translatorを組み込んだシステムの全体像を**図2**に示します．

開発PCでは，PyTorch/Keras/TensorFlow，または量子化済みTensorFlow Lite（以降，TF Lite）で作成した学習済みモデルを，e² studioのプラグインであるe-AI Translatorに取り込みます．e-AI Translatorは静的チェック（e-AI Checker）を実行した上で，マイコン向けのC/C++ソースコードやヘッダ・ファイルなどの生成物を出力します．これらをe² studioでビルド（コンパイルおよびリンク）し，推論実行用のバイナリを生成します．このバイナリがセンサ/画像/信号などの入力データを前処理し，ランタイムでニューラル・ネットワークの計算を行い，その結果を後処理やアプリケーションに引き渡します．

e-AI Translatorの内部処理の流れを**図3**に示します．e-AI Translatorは，PyTorch/Keras/TensorFlow，またはINT8形式のTF Liteで作成された学習済みモデルを取り込みます．モデル・パーサがグラフ構造や各レイヤ，重み情報を解析し，その後，活性化関数[注1]