参考資料

Application Note

エッジ人工知能アプリケーションでの *MSPM0G5187* の使用方法

概要

このアプリケーション ノートでは、MSPM0G5187 マイコンにエッジ人工知能 (AI) アプリケーションを実装する方法について説明します。このマイコンは、TinyEngine™ ニューラル プロセッシング ユニット (NPU) を搭載しており、リソースに制約のある組込みシステムで効率的なニューラル ネットワーク推論を実現するよう設計された専用のハードウェア アクセラレータです。この資料では、MSPM0G5187 のハードウェアアーキテクチャの概要と NPU の主な機能について説明し、その後に TI のエッジ AI ソフトウェアエコシステムについての包括的な紹介が続きます。その中には、TI の Edge AI Studio、Tiny ML Tensorlab、ニューラル ネットワーク コンパイラ (NNC) などが含まれます。次に、MSPM0G5187 のエッジ AI 機能を示すために、2 つの代表的なアプリケーション例を紹介します。1 つは、LeNet–5 CNN モデルに基づく手書きの数字を認識する設計、もう 1 つは時系列 CNN モデルに基づく波形分類の設計です。NPU/CPU ベースの推論の実装を比較した定量的性能ベンチマークが提供されており、推論レイテンシとエネルギー効率における NPU ハードウェア アクセラレーションの大きな利点が示されています。これらの結果は、組込みプラットフォームでエッジ AI 設計を開発するエンジニアにとって、実践的なガイダンスと設計リファレンスになります。

目次

1 概要 2
2 TinyEngine NPU を搭載した MSPM0G5187 3
3 エッジ AI ツールチェーン 4
3.1 TI Edge AI Studio 4
3.2 TI Tiny ML Tensorlab 5
3.3 TI ニューラル ネットワーク コンパイラ 5
4 エッジ AI アプリケーション：数字認識 6
4.1 LeNet–5 バリアント CNN モデル 6
4.2 NPU/CPU の性能比較 7
5 エッジ AI アプリケーション：波形分類 8
5.1 特徴抽出 9
5.2 時系列分類モデル 11
5.3 モデル メモリに関する検討事項 12
5.4 NPU/CPU の性能比較 13
6 まとめ 14
7 参考資料 14

商標

1 概要

エッジ人工知能 (AI) とは、データが生成されるデバイス、またはデータ ソースの近くにあるデバイス上で、AI 推論モデルをローカルに実行する技術を指します。クラウドベースの AI と比較して、エッジ AI には、レイテンシ低減による応答性の向上、帯域幅要件の低減による運用効率の向上、クラウド依存性の低減、ネットワーク中断時の継続的な運用によるシステムの信頼性の向上、データ転送の最小化とプライバシー保護の強化など、大きな利点があります。

TinyEngine NPU を搭載した MSPM0G5187 を活用することにより、コンシューマ エレクトロニクス、スマートハウス デバイス、産業用システム、車載システムなど、さまざまなドメインにおけるエンド デバイスに AI 機能を直接統合することができます。専用のハードウェア NPU が並列動作するので、ハードウェア アクセラレーションが搭載されていない同等のマイコンに比べて、推論レイテンシを最大 90 分の 1 に短縮し、1 回当たりの推論に使用するエネルギー消費量を 120 分の 1 まで低減できます。

ただし、エッジ AI は単なるハードウェアではありません。ユーザーの開発と展開を真の意味で加速するためには、包括的なエコシステムが必要です。一般的な分類タスクから、回帰タスク、予測タスクに至るまで、TI の包括的なソフトウェア エコシステムを活用することで、エンジニアは最小限の複雑さで AI ソリューションの迅速なプロトタイプ製作、最適化、展開を実現できます。

- **TI Edge AI Studio**:モデル開発、トレーニング、コンパイル、展開のための直観的なグラフィカル環境を提供し
- **TI Tiny ML Tensorlab**:コマンド ライン ツールを介して、モデルのトレーニング、量子化、コンパイル、展開を網羅するエンド ツー エンドのワークフローを提供
- **TI ニューラル ネットワーク コンパイラ**:シンプルな構成により、NPU または CPU にシームレスに展開が可能

MSPM0 SDK には、すぐに使用できるエッジ AI の豊富なサンプル プロジェクト集が含まれており、開発の迅速化に貢献します。開発者は、リファレンス コードをマイコンに迅速に展開したり、再トレーニング用の事前ビルド済みモデルをインポートしたり、ユーザー開発したカスタム AI モデルをアプリケーションに統合したりできます。

2 TinyEngine NPU を搭載した MSPM0G5187

MSPM0G5187 マイコン (MCU) は、最大 80MHz の周波数で動作する拡張 Arm® Cortex®-M0+ 32 ビット コア プラットフォームをベースにした、MSP 高集積超低消費電力 32 ビット MCU ファミリに含まれます。これらのマイコンは、小型パッケージまたはピン数の多いパッケージ (最大 64 ピン) で最大 128KB のフラッシュ メモリを必要とするアプリケーション向けで、コストの最適化と柔軟な設計を両立できます。これらのデバイスには、USB2.0-FS インターフェイス、デジタル オーディオ インターフェイス、TinyEngine NPU が含まれており、動作温度範囲全体にわたって優れた低消費電力性能を実現します。

ハードウェア NPU は、深い畳み込みニューラル ネットワーク (CNN) 向けに高度に最適化されたコアで、事前トレーニング済みのモデルを使用した機械学習推論をサポートしています。オンチップ CPU との組み合わせで動作して、CNN 推論で高い性能を発揮して、消費電力を低減します。640 ～ 2560MOPS (1 秒あたりオペレーション数) の性能を備えた NPU は、ソフトウェア実装に比べて 90 分の 1 のレイテンシを実現し、スタンバイモード時の消費電流は 2µA 未満です。

図 2-1 に、チップ内のモジュールのトップレベル構成を示します。

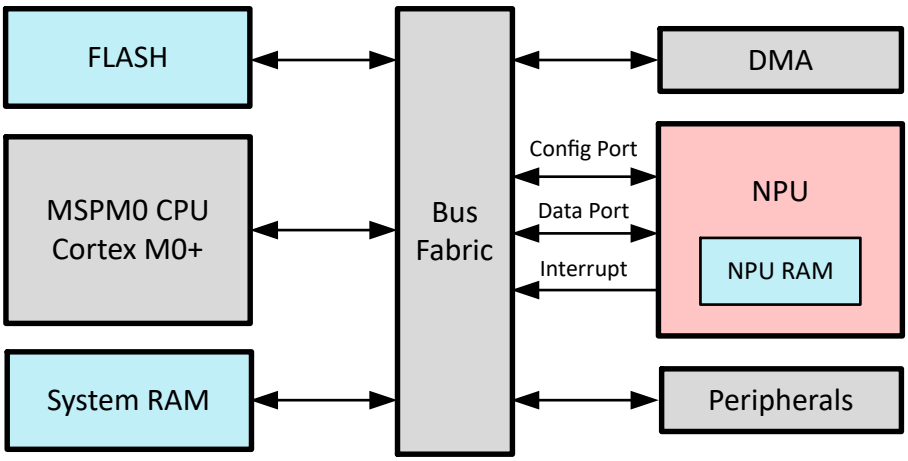

図 2-1. MSP CPU/NPU システム構造

NPU システムを搭載した MSPM0 は、不揮発性フラッシュ メモリと揮発性 RAM メモリを含むオンチップ メモリを共有しています。システム RAM は、CPU または NPU による読み取り / 書き込みをサポートしています。さらに、NPU には、CNN 推論専用 RAM が搭載されています。エッジ AI モデルは、利用可能なフラッシュ メモリの容量内に収まる必要があります。また、RAM の最大使用量も利用可能なシステム RAM の容量内に収まる必要があります。

3 エッジ AI ツールチェーン

3.1 TI Edge AI Studio

TI Edge AI Studio は、テキサス インスツルメンツの幅広いプロセッサ、マイコン、レーダー センサを対象に AI モデルの開発、学習、コンパイル、展開を簡素化するための、グラフィカル ツールおよびコマンド ライン ツールからなる包括的なスイートです。

Edge AI Studio は 2 つのバージョンが用意されており、さまざまな開発環境で利用できます。クラウドベースのバージョンは、ローカル環境の設定なしで完全にオンラインで動作し、開発者は最小限の構成で迅速に開発を開始できます。デスクトップのバージョンは、ユーザーのマシン上でローカルに実行されるため、すべてのデータとトレーニング済みモデルがオンプレミス内にとどまり、TI にアクセスされることはないため、厳格なデータ機密性が求められるプロジェクトに適しています。

Edge AI Studio は、以下の機能をサポートしています。

- データ キャプチャ、アノテーション、および視覚化
- ユーザーが用意したデータ セットを用いた、TI 提供モデルのトレーニング (BYOD：Bring Your Own Data、独自データの導入)
- カスタマイズされたトレーニング パラメータを用いたトレーニング モデル
- トレーニング済みモデルのコンパイル
- ライブ推論のプレビュー

現在、Edge AI Studio は以下の機能をサポートしていません。

- ユーザー定義モデルのトレーニング (TYOM：Train Your Own Model、独自モデルのトレーニング)
- TI モデルのアーキテクチャ変更
- 特徴抽出パイプラインまたは量子化フローのカスタマイズ
- 事前トレーニング済みまたはお客様が用意したモデルの直接コンパイル

3.2 TI Tiny ML Tensorlab

テキサス インスツルメンツの Tiny ML Tensorlab は、AI をマイコンに展開するための包括的な設計で、ユーザーは以下のことを実現できます。

- 時系列タスクと画像分類タスクに関する機械学習モデルのトレーニング
- 組込みデバイスへの展開を目的とする、(2 ビット、4 ビット、8 ビット) の量子化を用いたモデルの最適化
- TI のマイコンでモデルを効率的に実行するためのコンパイル (オプションとして、NPU アクセラレーションも使用可能)
- Code Composer Studio (CCS) を使用したモデルの展開

これは、次の機械学習 (ML) タスク タイプをサポートしています。

表 3-1. サポートされているタスク タイプ

タスク タイプ	説明
時系列分類	時系列データを個別のクラスに分類 (故障検出、アクティビティ認識など)
時系列回帰	時系列入力から連続値を予測 (トルク推定など)
時系列予測	過去のパターンに基づいた将来値の予測 (温度予測など)
異常検知	オートエンコーダベースのモデルを使用した異常パターンの特定 (機器のモニタリングなど)
画像分類	画像をクラスに分類 (目視検査、数字認識など)

その他資料

- 『Tiny ML Tensorlab GitHub リポジトリ』
- 『Tiny ML Tensorlab ユーザー ガイド』

3.3 TI ニューラル ネットワーク コンパイラ

TI ニューラル ネットワーク コンパイラ (NNC) は、TVM コンパイラ フレームワーク上に構築された、マイコン デバイス向けの TI 製組込み AI コンパイル ツールチェーンです。TI NNC を使用すると、シンプルなコンパイル構成で AI モデルをハードウェア NPU アクセラレータまたはホスト CPU に柔軟に展開できます。アプリケーション コードやモデル側に変更を加える必要はありません。推論は常に統合型インターフェイスを経由して呼び出されるため、開発者は基盤となるハードウェア アーキテクチャに関する深い知識がなくても、組込みアプリケーションに AI 推論を統合できます。

表 3-2. NPU/CPU 展開構成の比較

対象の展開先	構成シンボル	初期化要件	推論の呼び出し
ホスト CPU	TVMGEN_DEFAULT_TI_NPU_SOFT	あり。推論の前に NPU を初期化	tvmgen_default_run()
ハードウェア NPU	TVMGEN_DEFAULT_TI_NPU	なし	

詳細については、『マイコン用 TI ニューラル ネットワーク コンパイラ ユーザー ガイド』を参照してください。

4 エッジ AI アプリケーション：数字認識

数字認識は、組込み機器市場やコンシューマ市場の幅広いアプリケーションで活用される基盤技術です。しかし、信頼性の高い認識を組込みシステムに展開することは容易ではありません。従来のテンプレート マッチング方式では、人の手書き文字に発生しがちなバラツキに十分対応できないためです。そうした中で登場したエッジ AI により、有力な設計が可能になりました。クラウド接続に依存することなく、デバイス上で直接、高精度かつリアルタイムの推論を実行できるようになったのです。CPU ベースの推論パイプラインは膨大な演算リソースを必要とするため、リソース制約のある一般的なマイコンの処理能力では対応できません。

MSPM0G5187 マイコンはこの課題に対応するため、最小限の電力とメモリ オーバーヘッドで ML のワークロードを高速化するように設計された専用の TinyEngine NPU を採用しています。この設計では、1 桁の数字が書かれている画像を 10 種類のカテゴリ (0 ～ 9) のいずれかに分類する処理を実現します。このアプリケーションは、UART バック チャネル経由で 28×28 のグレイスケール画像をピクセル データとして受信し、オンチップの NPU を使用してハードウェア アクセラレーション モデルの推論を実行し、認識された数字クラスをホスト GUI に送信して表示します。

オンチップ NPU を活用することで、この数字認識システムは、約 99% の分類精度と 6.05ms の推論レイテンシを実現しています。また、メモリの消費量はフラッシュ メモリでわずか 73KB、RAM で 10.9KB です (表 4-1 を参照)。これらの結果から、リソース制約のあるマイコン上でもオンデバイス機械学習が実用可能であることが分かります。

表 4-1. 数字認識エッジ AI 設計のパフォーマンス

メトリック	値
精度	約 99%
フラッシュの使用率	73KB
RAM の使用法	10.9KB
推論レイテンシ (NPU)	6.05ms
推論消費電力 (平均)	424.65uJ

表 4-2 に、モードの主な情報を示します。

表 4-2. 数字認識 AI モデルの情報

特性	値
モデル アーキテクチャ	CNN
パラメータ数	60,000
入力形状	(1, 28, 28)
出力クラス	10
量子化	INT8

4.1 LeNet–5 バリアント CNN モデル

LeNet は、小さなグレイスケール画像から手書きの数字と文字を認識するために設計された、畳み込みニューラル ネットワーク (CNN) アーキテクチャの先駆的なシリーズです。数字認識エッジ AI モデルは、TI のニューラル ネットワーク コンパイラを使用して NPU ネイティブ命令へコンパイルされた LeNet-5 派生の CNN モデルです。

図 4-1 に、そのワークフローを示します。このソリューションでは、784 要素の float32 入力としてフラット化された 28×28 グレイスケール画像を受け取り、クラスごとの信頼スコアを表す 10 要素の float32 出力を生成します。このネットワークは、2 つの連続した畳み込みブロックで構成され、それぞれバイアス付きの Conv2D レイヤ、ReLU 活性化、および 2×2 MaxPooling で構成されています。これらのブロックは、空間分解能を段階ごとに半分にしながら、空間的特徴を徐々に抽出します。その後、得られた特徴マップはフラット化され、一連の完全に接続されたレイヤを通過して渡されます。これにより、高次元の特徴が 10 クラスの出力にマッピングされます。すべてのレイヤは INT8 の量子化重み付けと活性化で動作するため、完全に接続された最終レイヤの後に量子化の後処理ステップが適用され、出力を float32 に再スケーリングします。予測される数字は、ホスト側で 10 個の出力スコアに対して argmax を適用することで決定します。

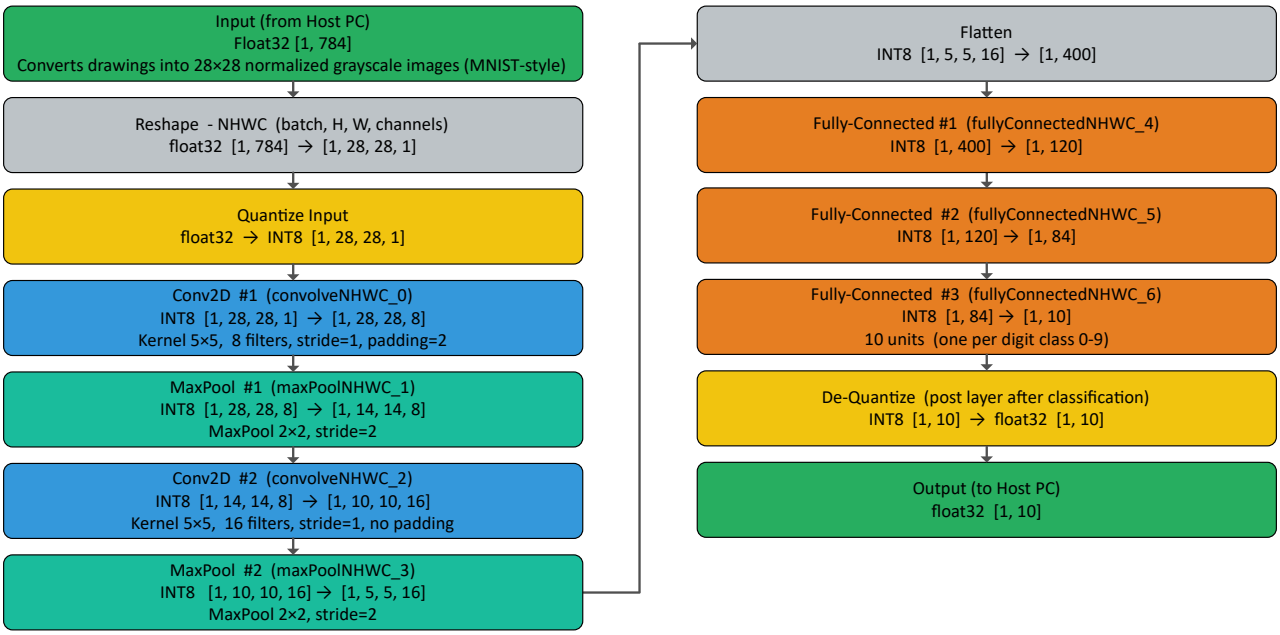

図 4-1. 数字認識 AI モデルのフロー チャート

4.2 NPU/CPU の性能比較

エッジ AI モデルは、TI ニューラル ネットワーク コンパイラを用いてハードウェアにデプロイでき、専用ハードウェア NPU またはホスト CPU のいずれかをターゲットとします。TinyEngine NPU は、高効率でニューラル ネットワーク計算を実行する専用ハードウェア アクセラレータであり、汎用 CPU に比べて推論レイテンシと消費電力が大幅に低減されます。

性能評価を容易にするため、MSPM0 SDK はユーザー ベンチマーク用として NPU と CPU の両方の実装例を提供しています。以下を参照してください。

- NPU ベースの数字認識エッジ AI の例
- CPU ベースの数字認識エッジ AI の例

表 4-3 に、数字認識アプリケーションにおける NPU ベースと CPU ベースのエッジ AI 設計の性能比較を示します。

表 4-3. NPU/CPU の性能比較

性能指標	NPU ベースの設計	CPU ベースの設計
精度	約 99%	約 99%
フラッシュの使用率	73KB	68KB
RAM の使用法	10.9KB	8.1KB
推論レイテンシ	6.05ms	89.81ms
推論消費電力 (平均)	424.65uJ	6,265.32uJ

NPU ベースの設計は、CPU ベースの実装に比べて、推論レイテンシが約 14 分の 1 に短縮されます。また、推論 1 回の電力消費が約 93% 低減するという実質的な効果もあるため、電力制約の厳しいエッジ アプリケーションにおいて高い効率性を実現します。

5 エッジ AI アプリケーション：波形分類

MSPM0 は、入力信号をリアルタイムかつ連続的に検出、分類する機能を備えた実験的な波形分類設計です。正弦波、三角波、のこぎり波などの一般的な波形を識別できます。この設計は、13K パラメータの時系列分類モデルに基づいており、高いスケーラビリティと拡張性を備えています。

この波形分類システムは、100% (R の 2 乗) の分類精度を達成し、推論レイテンシはわずか 5.84ms です。また、メモリの消費量はフラッシュで 21.5KB、RAM で 3.3KB のみです (表 5-1 を参照)。

表 5-1. 波形分類エッジ AI ソリューションの性能

メトリック	値
精度	100%
フラッシュの使用率	21.5KB
RAM の使用法	3.3KB
推論レイテンシ (NPU)	5.84ms
推論消費電力 (平均)	412.90uJ

表 5-2 に、モードの主な情報を示します。

表 5-2. 波形分類 AI モデルの情報

特性	値
モデル アーキテクチャ	CNN
パラメータ数	14,124
入力形状	テンソル [(1, 1, 128, 1)]
出力クラス	3 (のこぎり波、正弦波、方形波)
量子化	INT8

5.1 特徴抽出

特徴抽出は、一般的なエッジ AI アプリケーション、特にオーディオと時系列の信号処理にとって重要なコンポーネントです。この特徴抽出モジュールは、組込みシステム向けに最適化された効率的な信号処理機能を提供し、ARM ライブラリを活用して最適な性能を実現します。使用可能な特徴抽出機能には、次のものが含まれます。

- 実数高速フーリエ変換 (RFFT)：時間ドメイン信号を周波数ドメイン表現に変換します
- 振幅スケーリング：信号の大きさを適切な範囲に正規化するためのスケーリングを適用し、以後の処理でのオーバーフロー / アンダーフローを防止します
- 複素振幅の計算：FFT により得られる複素数の振幅を計算します
- DC コンポーネントの除去：特徴量の精度を向上させるため、信号の DC バイアスを除去します
- ビン計算：定義されたビン内の周波数成分を平均化し、次元数を削減します

図 5-1 に、特徴抽出処理のパイプラインを示します。

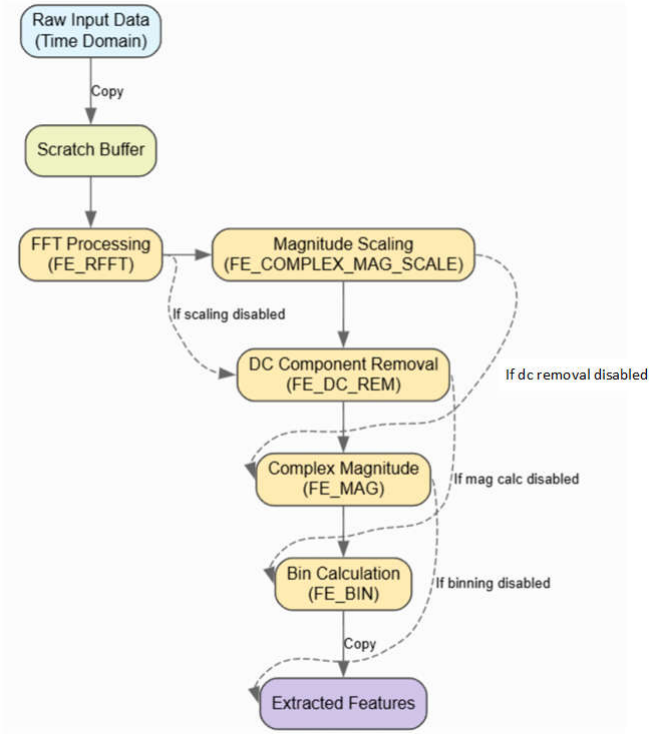

図 5-1. 特徴抽出処理のパイプライン

表 5-3 に、波形分類用に設計された特徴抽出処理パイプライン構成を示します。対応する構成シンボルが定義されると、関連タスクが処理パイプライン内で実行されます。

表 5-3. 波形分類のパイプライン構成

処理タスク	構成シンボル	定義の有無
RFFT	FE_RFFT	Y
振幅スケーリング	FE_COMPLEX_MAG_SCALE	Y
DC コンポーネントの除去	FE_DC_REM	Y
複素振幅の計算	FE_MAG	Y
ビン計算	FE_BIN	Y

表 5-4 に、波形分類用に設計された特徴抽出構成パラメータを示します。推論タスク 1 回あたり、8 フレームを連結します。各フレームでは 256 個のデータ ポイントを処理して 16 次元の特徴量を生成し、合計 128 次元の特徴量を波形分類 AI モデルに入力します。

表 5-4. 波形分類特徴の抽出パラメータ

メトリック	構成シンボル	値
入力フレーム サイズ	FE_FRAME_SIZE	256
フレームあたりの特徴サイズ	FE_FEATURE_SIZE_PER_FRAM	16
周波数ビン サイズ	FE_BIN_SIZE	8
ビン値を正規化します	FE_BIN_NORMALIZE	1
連結するフレーム	FE_NUM_FRAME_CONCAT	8
入力特徴サイズ	FE_STACKING_FRAME_WIDTH	128

注

表 5-4 で構成されている特徴量抽出パラメータは、モデル学習時のパラメータと同一に設定する必要があります。これにより、一貫した結果を確認できます。

性能に関する検討事項

- 最適化された計算:効率的な処理のために、ARM 最適化済みのライブラリを活用してください
- メモリ効率:バッファのスワップにより、メモリ コピーのオーバーヘッドが最小限に抑えられます
- 次元の削減:ビン計算により、特徴の次元が小さくなり、下流のステージでのメモリ使用量と計算量が減少します
- パラメータ調整:使用可能なメモリと計算リソースに合わせて構成パラメータを調整する必要があります

5.2 時系列分類モデル

時系列分類は、時系列データ内の特定のカテゴリ、運用状態、または事前定義されたラベルを識別し、ヘルスケア、ワイヤレス モニタリング、スマート セキュリティ、自動テストなどの領域に広く適用されています。

波形分類ソリューションでは、軽量な 13K パラメータの時系列分類モデル (CLS_13k_NPU) が実装されており、組込みニューラル プロセッシング ユニット (NPU) への効率的な展開を目的として設計されています。この設計では、256 ポイントの実数高速フーリエ変換 (RFFT) による特徴抽出パイプラインと、AI 推論タスクごとに 8 フレームを連結する方式を採用しており、入力として 128 要素の 1 次元時系列特徴 ベクトルを受け取り、出力として 3 要素のベクトルを生成します。各要素は、それぞれ対応するクラスの信頼度スコアを表します。

エクスポートされたモデル ファイルは、完全に接続された最終レイヤから INT8 (8 ビット量子化) テンソルを出力します。生の出力は、量子化された形式のクラス別信頼スコアで構成されているため、最終的な分類結果を導き出すには後処理ステップが必要です。

波形分類モデルのアーキテクチャ

モード ネットワークはシーケンシャル特徴変換パイプラインとして構成されており、各ステージは前の出力に直接入力として構築され、入力表現を段階的に洗練、抽象化しながら、最終的な分類判断を行います。図 5-2 にワークフローを示します。

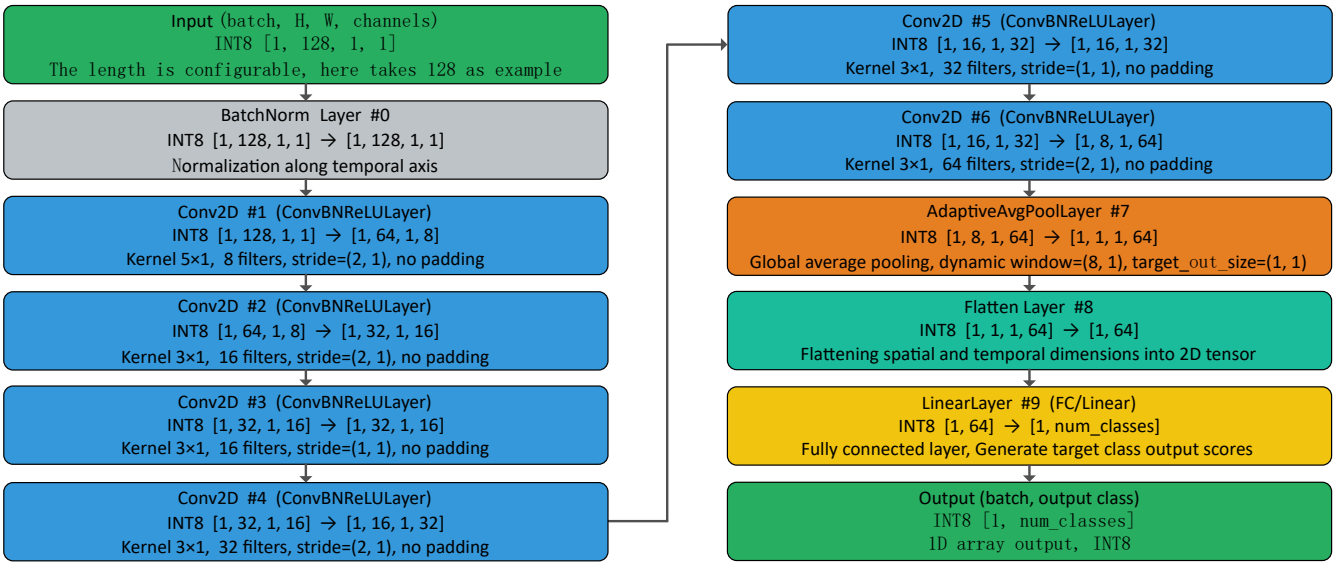

図 5-2. 時系列分類モデル (CLS_13k) のフロー チャート

このパイプラインは**バッチ正規化レイヤ**から始まります。このレイヤは生の入力シーケンスをリアルタイムで標準化し、学習済みの変換を適用する前に入力信号の振幅が変動しても一貫した特徴スケーリングを行います。入力が安定すると、ネットワークは 6 つの**連続した畳み込みブロック**を介して特徴表現を段階的に深めていきます。各ブロックは一貫した内部構造 (バイアスのある **Conv2D レイヤ**、**バッチ正規化**、**ReLU** 活性化) の順で構成されています。この繰り返し構造により、各ステージを経るごとに、抽象度を高めた時系列パターンをキャプチャします。各ブロック内の**バッチ正規化**は、深層化してもトレーニングの安定性を維持しますが、**ReLU** 活性化は、モデルが複雑な信号特性を学習するために必要な非線形性を付与します。

6 つの畳み込みステージの後、得られた特徴マップは**アダプティブ平均プーリング レイヤ**を通過します。このレイヤは、前の時間的長さに関係なく、空間分解能を 1x1 の固定サイズに縮約します。この設計の大きな特徴は、モデルを固定の時間次元に依存させない点です。これにより、長さの異なる時系列入力との互換性が本質的に確保され、アーキテクチャを変更することなく、将来的にさまざまな展開シナリオに柔軟に対応できます。

固定サイズにプールされた特徴マップは、2D テンソルにフラット化され、最終的な**完全接続 (FC) レイヤ**を通過します。このレイヤは、圧縮された高次元特徴表現をターゲット クラスの出力スコアに直接マッピングします。

モデル出力と後処理

エクスポートされたモデル ファイルは、完全に接続された最終レイヤから INT8 (8 ビット量子化) テンソルを出力します。生の出力は、量子化された形式のクラス別信頼スコアで構成されているため、最終的な分類結果を導き出すには後処理ステップが必要です。通常は、出力ベクトルに argmax 演算を適用し、最も高い信頼度スコアを持つクラスのインデックスを、予測分類結果とします。

5.3 モデル メモリに関する検討事項

リソースに制約のある組込みプラットフォーム上でエッジ AI ソリューションを開発するには、アルゴリズムの性能 (精度、レイテンシ) 、不揮発性ストレージ (フラッシュ / ROM)、ランタイム メモリ (SRAM) という 3 つの要素を厳密にトレードオフしながら最適化する必要があります。

- **アルゴリズムのパフォーマンス**:推論の精度とレイテンシの両方が含まれます。これらは主に、ネットワークの深さ、レイヤ幅、演算子の複雑さなどのモデル アーキテクチャによって形成されます。
- **静的ストレージ メモリ (フラッシュ / ROM)**:これは主にモデルのパラメータの合計数 (重みとバイアス) によって決まります。完全に量子化された INT8 パイプラインでは、各パラメータはちょうど 1 バイトのフラッシュ ストレージに直接マップされます。そのため、チャネル幅が拡張されたより深いネットワークでは、静的バイナリ サイズが線形的に増加します。場合によっては、入力となる特徴の次元の変化が、この静的メモリ使用量に影響を与えることがあります (例えば、完全に接続されたレイヤ)。
- **動的ランタイム メモリ (SRAM/RAM)**:これは、入力した特徴のサイズと、中間層をまたぐ活性化テンソル (特徴マップ) によって制御されます。時系列スライスがネットワークを介して伝播するとき、処理コアはレイヤの入力と出力を格納するために一時的なワークスペースを割り当てる必要があります。入力時間ウィンドウが長いか、特徴の次元が大きいと、このピーク ランタイム RAM 要件が指数関数的に増大します。

このようにハードウェアの現実的な制約があるため、エッジ環境への展開では従来の精度重視の考え方から脱却する必要があります。展開を成功させるには、静的フラッシュおよびピーク時の動的 SRAM という厳しい物理的な制約と、未加工モデルのパフォーマンスとの間で、きめ細かくバランスを取る必要があります。

この関係を説明するために、表 5-5 では、さまざまなモデル サイズと入力した特徴の構成に対して波形分類のメモリ フットプリントとリソース使用率を定量化しています。

表 5-5. モデル メモリ解析

モデル バリアント (パラメータ)	フラッシュ (ROM) のサイズ	RAM サイズ @ 入力 = 64	RAM サイズ @ 入力 = 128	RAM サイズ @ 入力 = 256
CLS_1K	約 5.6 KB	約 2.7 KB	約 5.3 KB	約 10.4 KB
CLS_4K	約 9.9 KB	約 1.2 KB	約 1.7 KB	約 2.7 KB
CLS_13K	約 21.4 KB	約 2.3 KB	約 3.3 KB	約 5.3 KB

すべての入力次元において、最小モデル (CLS_1K) は、大規模モデル (CLS_4K および CLS_13K) の 2 倍から 4 倍のランタイム RAM を消費します。一見直感に反するこの挙動は、パラメータ数とランタイム メモリが根本的に異なるアーキテクチャ設計に基づいているという事実に起因します。CLS_1K は、初期にダウン サンプリングしない浅いトポロジを使用しているため、高分解能の大きなフィーチャ マップが中間レイヤ全体のメモリに保持されます。これとは対照的に、CLS_4K と CLS_13K は、ネットワークのフロントエンドにストライド付き畳み込みを用いた深いアーキテクチャを採用しているため、時間次元が直ちに縮減され、中間のランタイム テンソルのサイズが大幅に小さくなります。

5.4 NPU/CPU の性能比較

エッジ AI モデルは、TI ニューラル ネットワーク コンパイラを用いてハードウェアにデプロイでき、専用ハードウェア NPU またはホスト CPU のいずれかをターゲットとします。性能評価を容易にするため、MSPM0 SDK はユーザー ベンチマーク用として NPU と CPU の両方の実装例を提供しています。以下を参照してください。

- NPU ベースの波形分類エッジ AI の例
- CPU ベースの波形分類エッジ AI の例

表 5-6 に、波形分類アプリケーションにおける NPU ベースと CPU ベースのエッジ AI 設計の性能比較を示します。

表 5-6. NPU/CPU の性能比較

性能指標	NPU ベースの設計	CPU ベースの設計
精度	100%	100%
フラッシュの使用率	21.5KB	21.7KB
RAM の使用法	3.3KB	3.1KB
特徴抽出レイテンシ	10.75ms	10.93ms
特徴抽出時の消費電力 (平均)	752.92uJ	755.8suJ
推論レイテンシ	5.77ms	54.42ms
推論消費電力 (平均)	412.90uJ	4559.68uJ

ハードウェアの演算効率を比較したところ、NPU アクセラレーションを採用したアーキテクチャは、ベースラインの CPU 実装に対して、性能と電力特性の両方で大きな優位性を示しました。

推論処理のみを見ると、NPU ベースの設計は優れた高速化を実現しており、実行レイテンシを約 8 分の 1 に低減するとともに、推論 1 回当たりのエネルギー消費量を約 91% 削減しています。

フロントエンドの特徴抽出処理を含めたエンド ツー エンドのシステム全体で評価すると、CPU が実行する信号前処理の負荷が一定であるため、全体の効率向上効果は一部相殺されます。それでも、NPU ベースのパイプラインでは、レイテンシが約 3 分の 1 に低減し、実行 1 回当たりのエネルギー消費量が約 78% 低減しています。

6 まとめ

このアプリケーション ノートは、MSPM0G5187 マイコンをベースとするエッジ AI ソリューションの包括的な概要を紹介し、ハードウェア特性、ソフトウェア ツール チェーン、代表的なアプリケーション例を説明します。

プラットフォームの性能を実証するために、2 種類のアプリケーションを評価しました。数字認識ソリューションでは、約 99% の分類精度を達成し、ハードウェア NPU は CPU ベースの実装と比べて、推論レイテンシを 15 分の 1 に低減する一方で、エネルギー効率を 93% 向上させています。一方、波形分類では 100% の分類精度を達成し、NPU は推論レイテンシを 8 分の 1 に低減する一方で、エネルギー効率を 91% 向上させています。

これらの結果から、TinyEngine NPU を搭載した MSPM0G5187 は、リソース制約のある組込み機器へ AI を効率的に導入するための有力なプラットフォームであることが確認されました。

7 参考資料

1. テキサス インスツルメンツ、『128kB フラッシュ、32kB SRAM、USB 2.0 FS、I2S、ADC、TinyEngine™ NPU 搭載、80MHz Arm® Cortex®-M0+ マイコン』、製品ページ
2. テキサス インスツルメンツ、『エッジ AI アクセラレーションを採用した Arm® Cortex®-M0+ マイコンはどのようにエレクトロニクスの脳力を向上させるか』、技術記事
3. テキサス インスツルメンツ、『TI の TinyEngine™ NPU により、多くの組込みシステムでエッジ AI アクセラレーションを実現』、製品概要
4. テキサス インスツルメンツ、『MSPM0 低コスト マイコンでのエッジ AI を使用した AC アーク フォルト検出』、アプリケーション ノート
5. テキサス インスツルメンツ、MSPM0 G シリーズ 80MHz マイコン、テクニカル リファレンス マニュアル。
6. テキサス インスツルメンツ、『TI のマイコン (MCU) による数字認識』、製品ページ
7. テキサス インスツルメンツ、『Tiny ML Tensorlab GitHub リポジトリ』、Web ページ
8. テキサス インスツルメンツ、『Tiny ML Tensorlab ユーザー ガイド』、Web ページ
9. テキサス インスツルメンツ、『MSPM0 エッジ AI ユーザー ガイド』、Resource Explorer

重要なお知らせと免責事項

TI は、技術データと信頼性データ (データシートを含みます)、設計リソース (リファレンス デザインを含みます)、アプリケーションや設計に関する各種アドバイス、Web ツール、安全性情報、その他のリソースを、欠陥が存在する可能性のある「現状のまま」提供しており、商品性および特定目的に対する適合性の黙示保証、第三者の知的財産権の非侵害保証を含むいかなる保証も、明示的または黙示的にかかわらず拒否します。

これらのリソースは、TI 製品を使用する設計の経験を積んだ開発者への提供を意図したものです。(1) お客様のアプリケーションに適した TI 製品の選定、(2) お客様のアプリケーションの設計、検証、試験、(3) お客様のアプリケーションに該当する各種規格や、その他のあらゆる安全性、セキュリティ、規制、または他の要件への確実な適合に関する責任を、お客様のみが単独で負うものとします。

上記の各種リソースは、予告なく変更される可能性があります。これらのリソースは、リソースで説明されている TI 製品を使用するアプリケーションの開発の目的でのみ、TI はその使用をお客様に許諾します。これらのリソースに関して、他の目的で複製することや掲載することは禁止されています。TI や第三者の知的財産権のライセンスが付与されている訳ではありません。お客様は、これらのリソースを自身で使用した結果発生するあらゆる申し立て、損害、費用、損失、責任について、TI およびその代理人を完全に補償するものとし、TI は一切の責任を拒否します。

TI の製品は、 TI の販売条件 、 TI の総合的な品質ガイドライン 、 ti.com または TI 製品などに関連して提供される他の適用条件に従い提供されます。TI がこれらのリソースを提供することは、適用される TI の保証または他の保証の放棄の拡大や変更を意味するものではありません。 TI がカスタム、またはカスタマー仕様として明示的に指定していない限り、TI の製品は標準的なカタログに掲載される汎用機器です。

お客様がいかなる追加条項または代替条項を提案する場合も、TI はそれらに異議を唱え、拒否します。

最終更新日：2025 年 10 月