

RL78/F23, F24

R01AN6846JJ0100

Rev.1.00

アプリケーション・アクセラレータ・ユニットの使用例

2023.7.30

要旨

RL78/F23, F24 のアプリケーション・アクセラレータ・ユニット（以下 AAU）は、モータ制御や DC/DC コンバータ制御のアルゴリズム演算のためのソフトウェア負荷を軽減することを目的とした算術支援ハードウェアです。

AAU は下記のアルゴリズム演算に対応しています。

アルゴリズム演算	主な用途	実行サイクル数	参照
Sine 演算	モータのベクトル制御における座標変換など	1 クロック	1.1
Cosine 演算		1 クロック	
クラーク/パーク変換	モータのベクトル制御（三相 ⇄ 二相変換、および PI 制御）で使用	7 クロック	1.1
逆パーク変換		6 クロック	
逆クラーク変換		5 クロック	
ベクトル制御向け PI 演算		15 クロック	
クラーク/パーク変換、PI 演算一括		22 クロック	
パーク/クラーク逆変換		11 クロック	
DC/DC 制御向け PI 演算 (1ch, 2ch, 3ch)	DC/DC コンバータなどによる電源制御	6 クロック (1ch 時) 12 クロック (2ch 時) 18 クロック (3ch 時)	1.2
32 ビット乗算 (32 ビット × 32 ビット = 64 ビット)	32 ビット精度の符号付きデータを使用した数値演算、デジタルフィルタの積和演算など	5 クロック	1.3

目次

1. AAU の使用例	3
1.1 モータ制御の使用例	3
1.2 電源制御の使用例	4
1.3 デジタルフィルタの使用例	5
2. AAU 使用上の注意	11
3. 参考文献	12
改訂記録	13

1. AAU の使用例

AAU 演算モードの使用例として、「モータ制御の使用例」、「電源制御の使用例」、「デジタルフィルタの使用例」を示します。

1.1 モータ制御の使用例

モータ（ベクトル制御）で AAU を使用する例を図 1-1 に示します。

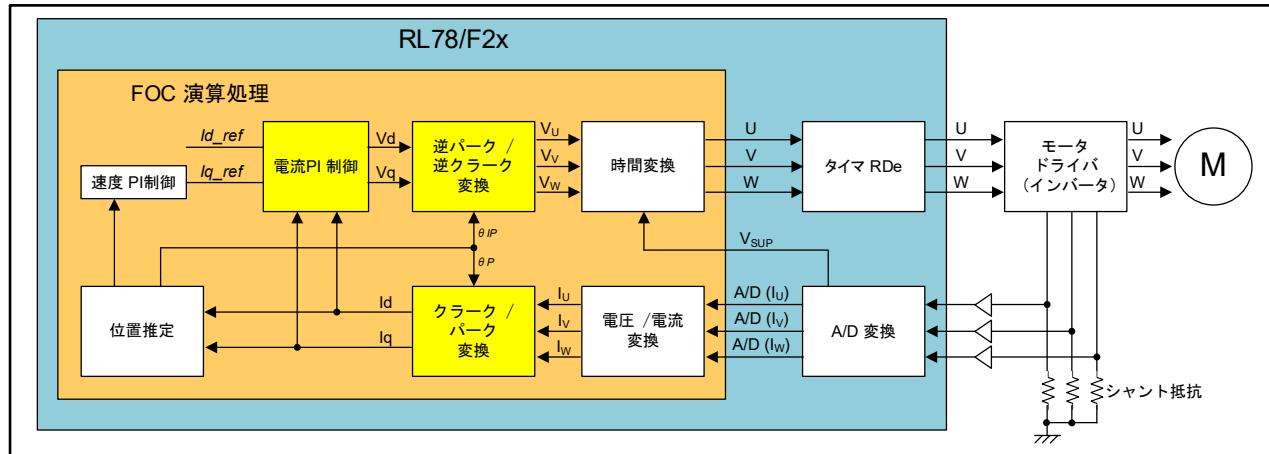


図 1-1 モータ（ベクトル制御）での AAU 使用例

表 1-1 モータ（ベクトル制御）で使用する AAU 演算モード

	演算モード	モード動作	備考
1	Sine 演算	位相角 (θ) に対する Sine 値を取得します。	モータ（ベクトル制御）のクラーク/パーク変換等で使用します。
2	Cosine 演算	位相角 (θ) に対する Cosine 値を取得します。	
3	クラーク/パーク変換モード	3 相 2 相変換（クラーク変換）と座標軸変換（パーク変換）を順次実行します。 ^注	クラーク変換：3 相電流 (I_u 、 I_v 、 I_w) を 2 相電流 (I_α 、 I_β) に変換します。 パーク変換：固定座標 (I_α 、 I_β) に回転角 (θ) を加え回転座標 (I_d 、 I_q) に変換します。
4	逆パーク変換モード	固定座標変換（逆パーク変換）を実行します。	逆パーク変換：回転座標 (V_d 、 V_q) から固定座標 (V_α 、 V_β) に変換します。
5	逆クラーク変換モード	2 相 3 相変換（逆クラーク変換）を実行します。 ^注	逆クラーク変換：2 相電圧 (V_α 、 V_β) を 3 相電圧 (V_u 、 V_v 、 V_w) に変換します。
6	ベクトル制御向け PI 演算モード	モータの d 軸（磁束方向）、q 軸（d 軸と直交）の電流を目標電流値に近づけるため、P 制御（比例制御）、I 制御（積分制御）を実行します。	PI 演算：モータの d 軸、q 軸の電流指令値（安定動作のための目標値）にするための PI 制御を行います。PI 制御により指令電圧 (V_d^* 、 V_q^*) を得ます。
7	クラーク/パーク変換、PI 演算一括モード	クラーク/パーク変換とベクトル制御向け PI 演算を順次実行します。 ^注	上記 3. と 6. を参照してください。
8	パーク/クラーク逆変換モード	逆パーク変換、逆クラーク変換を順次実行します。 ^注	上記 4. と 5. を参照してください。

【注】クラーク変換または逆クラーク変換時、絶対変換モードまたは相対変換モードを選択可能です。

本使用例の詳細につきましては、下記のドキュメントをご参照ください。

- 「RL78/F24 MCU による PMSM モータのセンサレスベクトル制御編（单一シャント）」
- 「RL78/F24 MCU による PMSM モータのセンサレスベクトル制御編（3 シャント）」

1.2 電源制御の使用例

LED の DC/DC 電源制御で AAU を使用する例を図 1-2 に示します。

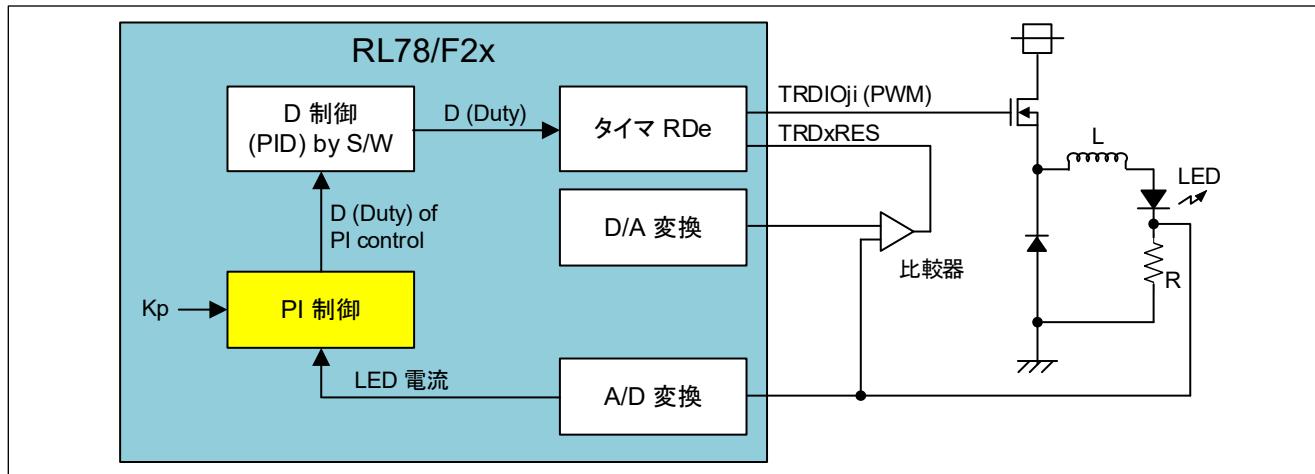


図 1-2 LED 電源制御の AAU 使用例

表 1-2 LED 電源制御で使用する AAU 演算モード

演算モード	モード動作	備考
DC/DC 制御向け PI 演算	サンプリングの前後で変化した出力電圧を元に PI 演算を行い、目標の出力電圧で安定させるためのスイッチング周期のデューティ値を算出します。	PI 演算：DC/DC コンバータによる電源制御（PWM によるスイッチング）により、LED に安定した電圧を供給するため PI 制御を行います。PI 制御によりデューティ値（電圧 ON 時間 : D(n)）を得ます。（図 1-2 は PI 制御後にソフトウェアによる D（微分）制御を実施しています）

LED 電源制御の使用例（詳細）については、アプリケーションノート「RL78/F23, F24 タイマ RDe、AAU を使用した DC/DC 制御（R01AN6838）」を参照ください。

1.3 デジタルフィルタの使用例

デジタルフィルタの IIR フィルタで AAU を使用する例を図 1-3 に示します。

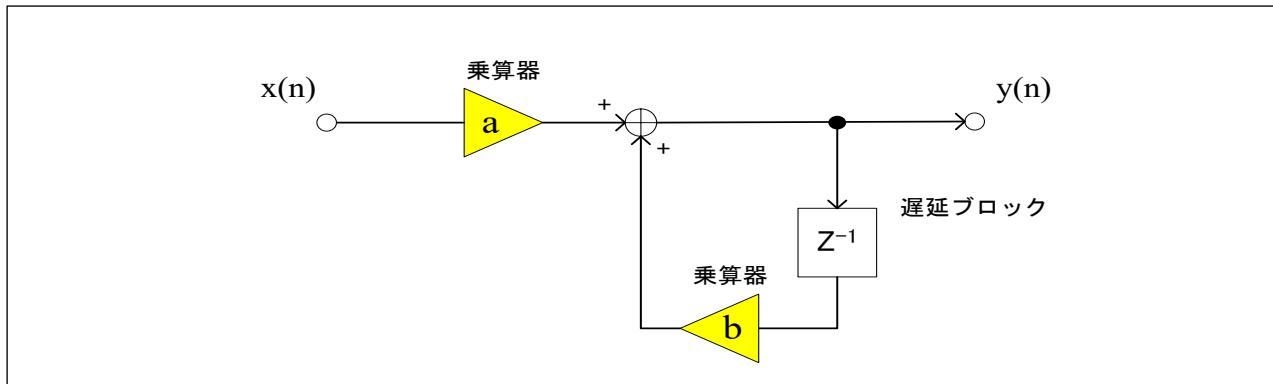


図 1-3 単極 IIR フィルタのブロック図

表 1-3 デジタルフィルタ(IIR フィルタ)で使用する AAU 演算モード

演算モード	モード動作	備考
32 ビット乗算	乗数（符号あり 32 ビット）、被乗数（符号あり 32 ビット）の乗算を行います。	デジタルフィルタの乗算器などの演算処理で使用します。

図の単極 IIR フィルタは、フィードバックループを持ち、遅延ブロックにより前回の出力 $y(n-1)$ が出力 $y(n)$ に影響する再帰型のデジタルフィルタです。

この単極 IIR フィルタの出力は、次の式で計算されます。

$$y(n) = ax(n) + by(n - 1)$$

$x(n)$ は入力信号、 a と b はフィルタ特性を表す定数です。

IIR フィルタは、A/D コンバータを通した音声データを入力信号とし、デジタルフィルタにより高周波成分をカットするなどの信号処理に使います。

単極 IIR フィルタの出力を AAU の 32 ビット乗算を使用して実装した場合のフローチャートを図 1-4 に、ソースコードを図 1-5 に示します。

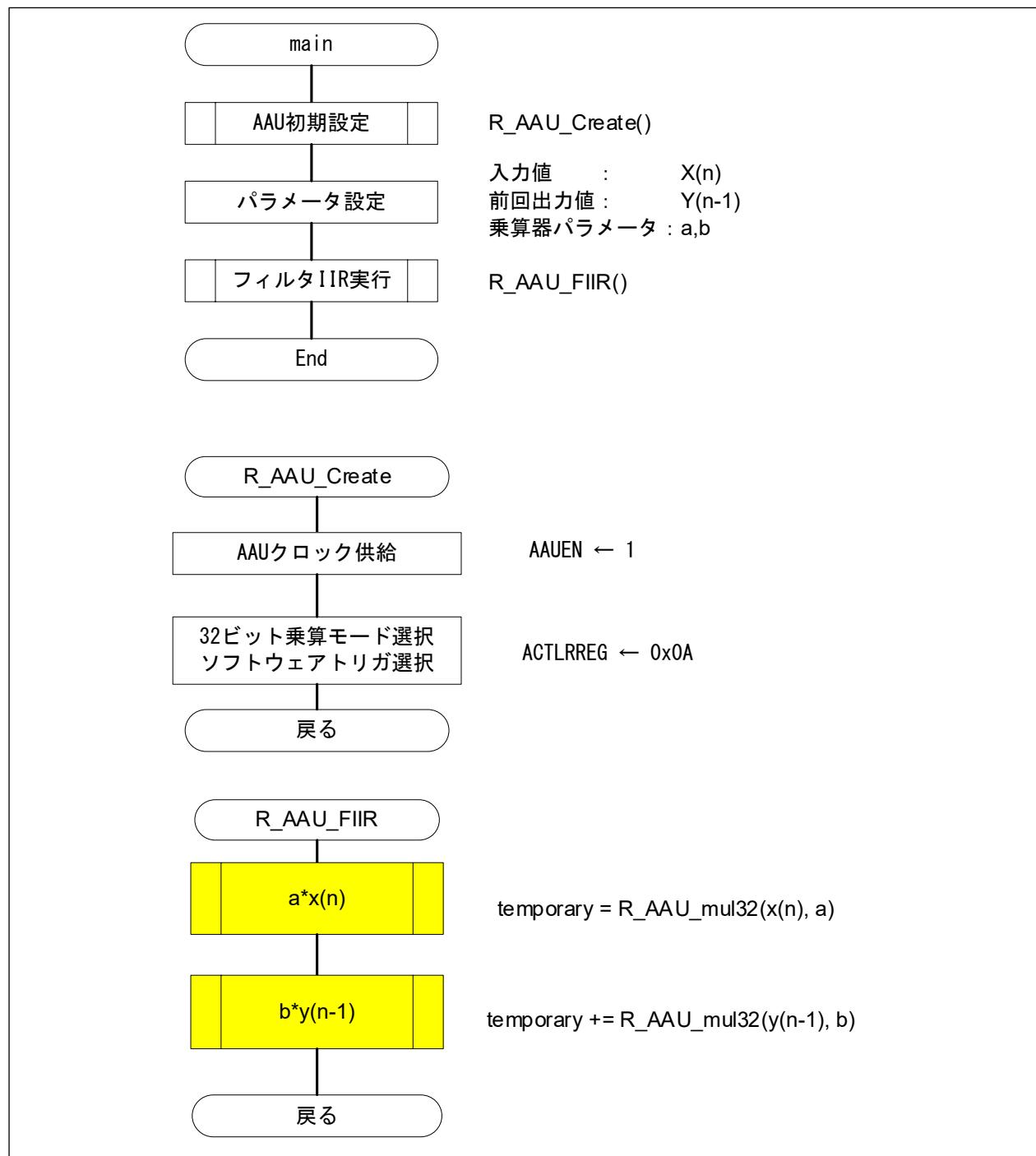


図 1-4 単極 IIR フィルタフローチャート (1/2)

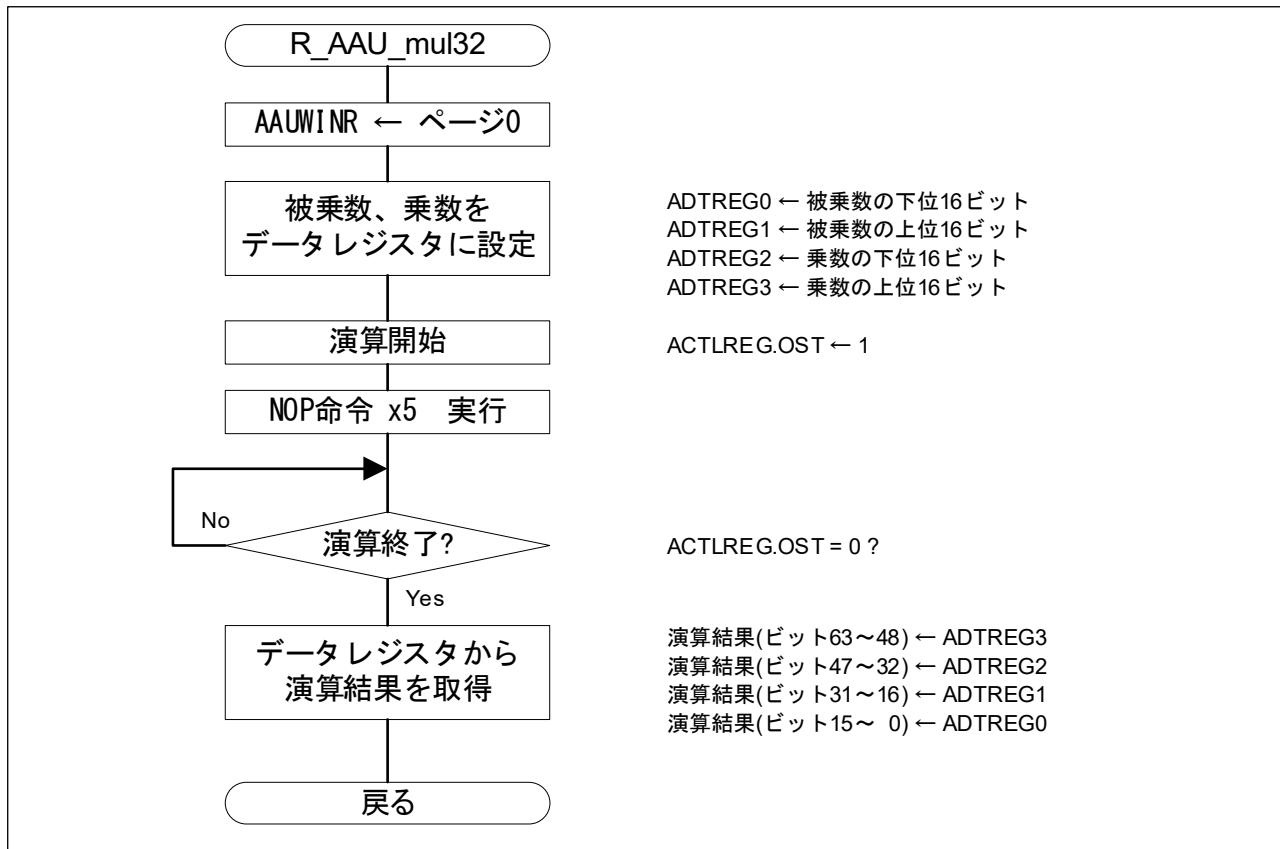


図 1-4 單極 IIR フィルタフローチャート (2/2)

```

int64_t R_AAU_mul32(int32_t src, int32_t src2)
{
    int64_t temporary = 0;

    AAUWINR = _00_AAU_WINDOW_PAGE0;

    /* Set calculation data */
    ADTREG0 = (uint16_t)(src & 0x0000FFFF);
    ADTREG1 = (uint16_t)((src & 0xFFFF0000) >> 16U);
    ADTREG2 = (uint16_t)(src2 & 0x0000FFFF);
    ADTREG3 = (uint16_t)((src2 & 0xFFFF0000) >> 16U);

    ACTLREG |= _01_AAU_OPERATION_START;

    /* wait for calculation end */
    NOP(); NOP(); NOP(); NOP(); NOP();

    while((ACTLREG & _01_AAU_OPERATION_START) != 0x01);

    temporary = ((int64_t)ADTREG3 << (48U)) | ((int64_t)ADTREG2 << (32U)) |
                ((int64_t)ADTREG1 << (16U)) | (int64_t)ADTREG0;
    return temporary;
}

void R_AAU_Create(void)
{
    AAUEN = 1U;
    /* Set the multiply operation and start trigger */
    AAUWINR = _00_AAU_WINDOW_PAGE0;
    ACTLREG = (_02_AAU_START_TRG_OST | _00_AAU_START_TRG_DATA_SET);
    ACTLREG |=(_08_AAU_OPERATION_MULW | _00_AAU_TRANS_MODE_POWER );
}

void R_AAU_FIIR(int32_t input)
{
    int64_t temporary;

    temporary = R_AAU_mul32(input, param[0]);           /* ax(n) */
    temporary += R_AAU_mul32(g_delay, param[1]);        /* by(n-1) */

    g_delay = (int32_t)(temporary & 0x00000000FFFFFFFF);
}

void main(void)
{
    R_AAU_Create();
    /* initial value */
    input = 0x01234567;                                /* x(n) */
    g_delay = 0x00003333;                               /* y(n-1) */
    param[0] = 5;                                       /* a */
    param[1] = 1;                                       /* b */

    R_AAU_FIIR(input);                                 /* y(n) */
    while(1);
}

```

図 1-5 デジタルフィルタ (IIR フィルタ) の実装例

32 ビット乗算モードは、符号付き 32 ビットデータの乗算（32 ビット × 32 ビット = 64 ビット）をサポートします。32 ビット乗算を実行する際、乗数と被乗数は図 1-6 の様に 2 つのレジスタを連結して使用します。負のデータを使用する場合は、2 の補数でレジスタに設定してください。

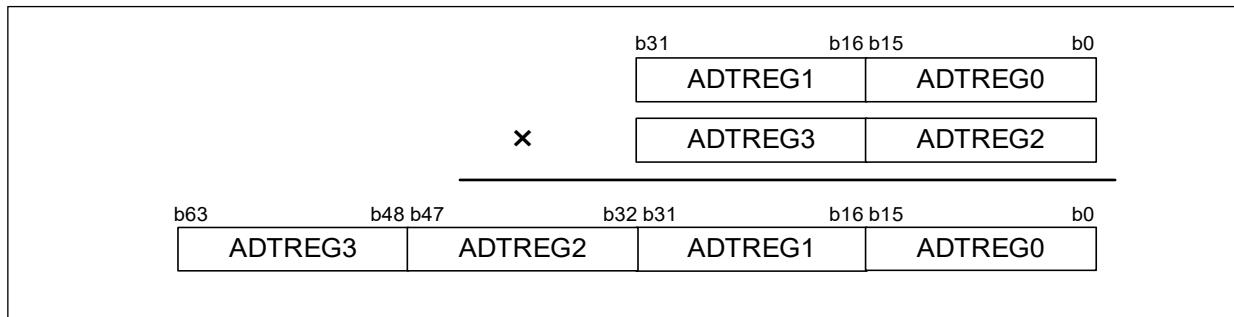


図 1-6 32 ビット乗算モードで使用するレジスタ

以下に 32 ビット乗算使用時のレジスタ設定、結果出力例を示します。

設定例(1)：正のデータ × 正のデータ

01234567H × 01234567H の実施例を図 1-7 に示します。乗数、被乗数のデータをレジスタにセットする場合、連結した 2 つのレジスタの最上位ビット(b31)は符号ビットとして使用されるため、乗数、被乗数のデータで使用しないよう注意が必要です。

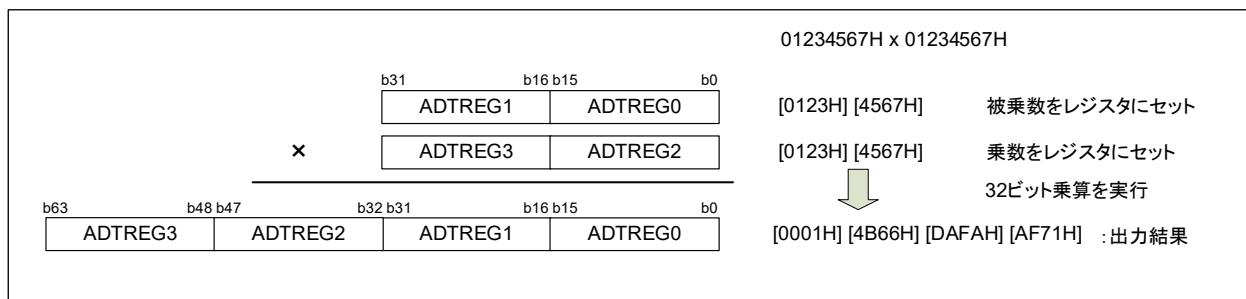


図 1-7 正×正 のレジスタ設定、結果出力例

設定例(2)：正のデータ × 負のデータ

01234567H × (-1)の実施例を図 1-8 に示します。乗数に負数を使用する場合、2 の補数でデータをレジスタにセットします。また、結果が負となる演算を実施した場合、結果データは 2 の補数データとしてレジスタに出力されます。

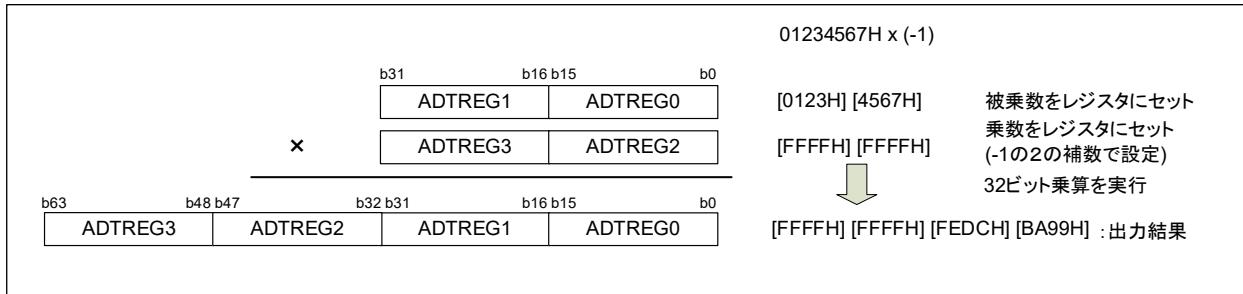


図 1-8 正×負 のレジスタ設定、結果出力例

設定例(3)：負のデータ × 負のデータ

89ABCDEFH × (-1) の実施例を図 1-9 に示します。被乗数、乗数に負数を使用する場合、2 の補数でデータをレジスタにセットします。

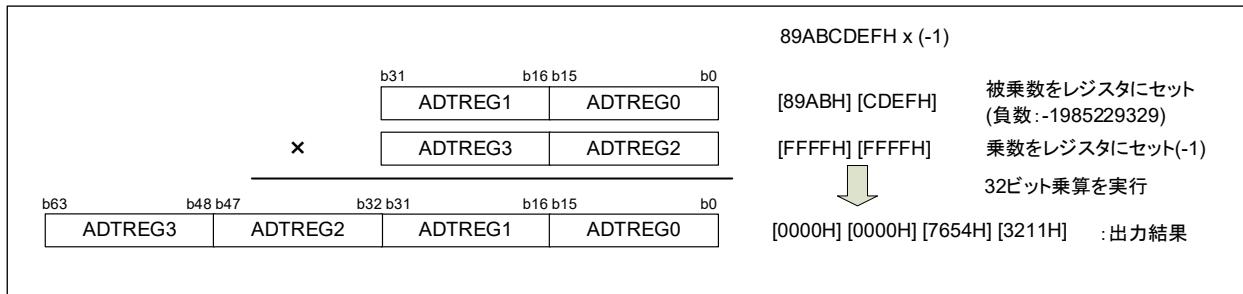


図 1-9 負×負 演算時のレジスタ設定、結果出力例

2. AAU 使用上の注意

- (1) AAU の各レジスタは、AAUWINR レジスタの AAUPAGE [1:0] ビットでアクセス・ページを選択してから対象のレジスタをアクセスしてください
- (2) AAU が演算中 (ACTLREG レジスタの OST ビットが “1”) の場合、AAU の各レジスタを書き換えないでください
- (3) AAU の演算開始条件
 - ACTLREG レジスタの STM ビットが “0” の場合、最終オペランドトリガ・レジスタ（詳細はユーザーズマニュアルハードウェア編を確認）への書き込みで演算を開始します
 - STM ビットが “1” の場合、ACTLREG レジスタの OST ビットに “1” を書き込むことで、演算を開始します
- (4) AAU の演算終了判定
 - 対象の演算の実行サイクル分ソフトウェアで待機します。各演算の実行サイクルは、ユーザーズマニュアルハードウェア編で確認してください（例：NOP 命令 × 実行サイクル分ウェイト）
 - ACTLREG レジスタの OST ビットが “0” になるまで待機します
- (5) 複数の演算モードを使用する場合は、必要に応じてレジスタの退避・復帰を行ってください。下表に各演算モードで使用するレジスタを示します

表 2-1 各演算モードの使用レジスタ

演算モード	入力	出力
Sine 演算	ADTREG0	ADTREG0
Cosine 演算	ADTREG0	ADTREG0
32 ビット乗算	ADTREG0, ADTREG1, ADTREG2, ADTREG3	ADTREG0, ADTREG1, ADTREG2, ADTREG3
クラーク/パーク変換	ADTREG0, ADTREG1, ADTREG2	ADTREG0, ADTREG1
逆パーク変換	ADTREG0, ADTREG1, ADTREG2	ADTREG0, ADTREG1
逆クラーク変換	ADTREG0, ADTREG1,	ADTREG0, ADTREG1, ADTREG2
PI 演算（ベクトル制御向け）	ADTREG0, ADTREG1 初期設定 : AIDREF, AIQREF, AKPD, AKID, AKPQ, AKIQ, AILIM, APILIM, AKRAG	ADTREG0, ADTREG1 内部使用 : AI DBFL, AI DBFH, AI QBFL, AI QBFH, ADOVER, AQOVER
クラーク/パーク変換、PI 演算	ADTREG0, ADTREG1, ADTREG2 初期設定 : AIDREF, AIQREF, AKPD, AKID, AKPQ, AKIQ, AILIM, APILIM, AKRAG	ADTREG0, ADTREG1, ADTREG2, ADTREG3 内部使用 : AI DBFL, AI DBFH, AI QBFL, AI QBFH, ADOVER, AQOVER
パーク/クラーク逆変換	ADTREG0, ADTREG1, ADTREG2	ADTREG0, ADTREG1, ADTREG2
PI 演算（DC/DC 制御向け） ^注	ADTREGn 初期設定 : ALnREF, ALnOFS, AKI1, AKI2, ADUTYMX	ADTREGn 内部使用 : ADUTYLn, APILn

【注】ACTLREG レジスタの演算モードにより使用するレジスタは異なります。

対応チャネルが 1 チャネルの場合 n = 1 に、2 チャネルの場合 n = 1, 2 になります。また、3 チャネルの場合は n = 1, 2, 3 になります。

3. 参考文献

本アプリケーションノートにおける参考資料を以下に示します。参照の際は、ルネサスエレクトロニクスホームページから最新版を入手してください。

- RL78/F23, F24 ユーザーズマニュアルハードウェア編 Rev. 1.00
- RL78/F24 MCU による PMSM モータのセンサレスベクトル制御編（単一シャント） Rev.1.10
- RL78/F24 MCU による PMSM モータのセンサレスベクトル制御編（3 シャント） Rev.1.10
- RL78/F23, F24 タイマ RDe と AAU を用いた DC/DC 制御 Rev.1.00

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2023. 7.30	—	初版発行

製品ご使用上の注意事項

ここでは、マイコン製品全体に適用する「使用上の注意事項」について説明します。個別の使用上の注意事項については、本ドキュメントおよびテクニカルアップデートを参照してください。

● 1. 静電気対策

CMOS 製品の取り扱いの際は静電気防止を心がけてください。CMOS 製品は強い静電気によってゲート絶縁破壊を生じることがあります。運搬や保存の際には、当社が出荷梱包に使用している導電性のトレーヤマガジンケース、導電性の緩衝材、金属ケースなどを利用し、組み立て工程にはアースを施してください。プラスチック板上に放置したり、端子を触ったりしないでください。また、CMOS 製品を実装したボードについても同様の扱いをしてください。

● 2. 電源投入時の処置

電源投入時は、製品の状態は不定です。電源投入時には、LSI の内部回路の状態は不確定であり、レジスタの設定や各端子の状態は不定です。外部リセット端子でリセットする製品の場合、電源投入からリセットが有効になるまでの期間、端子の状態は保証できません。同様に、内蔵パワーオンリセット機能を使用してリセットする製品の場合、電源投入からリセットのかかる一定電圧に達するまでの期間、端子の状態は保証できません。

● 3. 電源オフ時における入力信号

当該製品の電源がオフ状態のときに、入力信号や入出力プルアップ電源を入れないでください。入力信号や入出力プルアップ電源からの電流注入により、誤動作を引き起こしたり、異常電流が流れ内部素子を劣化させたりする場合があります。資料中に「電源オフ時における入力信号」についての記載のある製品は、その内容を守ってください。

● 4. 未使用端子の処理

未使用端子は、「未使用端子の処理」に従って処理してください。CMOS 製品の入力端子のインピーダンスは、一般に、ハイインピーダンスとなっています。未使用端子を開放状態で動作させると、誘導現象により、LSI 周辺のノイズが印加され、LSI 内部で貫通電流が流れたり、入力信号と認識されて誤動作を起こす恐れがあります。

● 5. クロックについて

リセット時は、クロックが安定した後、リセットを解除してください。プログラム実行中のクロック切り替え時は、切り替え先クロックが安定した後に切り替えてください。リセット時、外部発振子（または外部発振回路）を用いたクロックで動作を開始するシステムでは、クロックが十分安定した後、リセットを解除してください。また、プログラムの途中で外部発振子（または外部発振回路）を用いたクロックに切り替える場合は、切り替え先のクロックが十分安定してから切り替えてください。

● 6. 入力端子の印加波形

入力ノイズや反射波による波形歪みは誤動作の原因になりますので注意してください。CMOS 製品の入力がノイズなどに起因して、 V_{IL} (Max.) から V_{IH} (Min.) までの領域にとどまるような場合は、誤動作を引き起こす恐れがあります。入力レベルが固定の場合はもちろん、 V_{IL} (Max.) から V_{IH} (Min.) までの領域を通過する遷移期間中にチャタリングノイズなどが入らないように使用してください。

● 7. リザーブアドレス（予約領域）のアクセス禁止

リザーブアドレス（予約領域）のアクセスを禁止します。アドレス領域には、将来の拡張機能用に割り付けられている リザーブアドレス（予約領域）があります。これらのアドレスをアクセスしたときの動作については、保証できませんので、アクセスしないようにしてください。

● 8. 製品間の相違について

型名の異なる製品に変更する場合は、製品型名ごとにシステム評価試験を実施してください。同じグループのマイコンでも型名が違うと、フラッシュメモリ、レイアウトパターンの相違などにより、電気的特性の範囲で、特性値、動作マージン、ノイズ耐量、ノイズ幅射量などが異なる場合があります。型名が違う製品に変更する場合は、個々の製品ごとにシステム評価試験を実施してください。

ご注意書き

- 1. 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。回路、ソフトウェアおよびこれらに関連する情報を使用する場合、お客様の責任において、お客様の機器・システムを設計ください。これらの使用に起因して生じた損害（お客様または第三者いずれに生じた損害も含みます。以下同じです。）に関し、当社は、一切その責任を負いません。
- 2. 当社製品または本資料に記載された製品データ、図、表、プログラム、アルゴリズム、応用回路例等の情報の使用に起因して発生した第三者の特許権、著作権その他の知的財産権に対する侵害またはこれらに関する紛争について、当社は、何らの保証を行ふものではなく、また責任を負うものではありません。
- 3. 当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
- 4. 当社製品を組み込んだ製品の輸出入、製造、販売、利用、配布その他の行為を行うにあたり、第三者保有の技術の利用に関するライセンスが必要となる場合、当該ライセンス取得の判断および取得はお客様の責任において行ってください。
- 5. 当社製品を、全部または一部を問わず、改造、改変、複製、リバースエンジニアリング、その他、不適切に使用しないでください。かかる改造、改変、複製、リバースエンジニアリング等により生じた損害に関し、当社は、一切その責任を負いません。
- 6. 当社は、当社製品の品質水準を「標準水準」および「高品質水準」に分類しており、各品質水準は、以下に示す用途に製品が使用されることを意図しております。

標準水準：コンピュータ、OA機器、通信機器、計測機器、AV機器、家電、工作機械、パーソナル機器、産業用ロボット等
高品質水準：輸送機器（自動車、電車、船舶等）、交通制御（信号）、大規模通信機器、金融端末基幹システム、各種安全制御装置等
当社製品は、データシート等により高信頼性、Harsh environment 向け製品と定義しているものを除き、直接生命・身体に危害を及ぼす可能性のある機器・システム（生命維持装置、人体に埋め込み使用するもの等）、もしくは多大な物的損害を発生させるおそれのある機器・システム（宇宙機器と、海底中継器、原子力制御システム、航空機制御システム、プラント基幹システム、軍事機器等）に使用されることを意図しておらず、これらの用途に使用することは想定していません。たとえ、当社が想定していない用途に当社製品を使用したことにより損害が生じても、当社は一切その責任を負いません。
- 7. あらゆる半導体製品は、外部攻撃からの安全性を 100% 保証されているわけではありません。当社ハードウェア／ソフトウェア製品にはセキュリティ対策が組み込まれているものもありますが、これによって、当社は、セキュリティ脆弱性または侵害（当社製品または当社製品が使用されているシステムに対する不正アクセス・不正使用を含みますが、これに限りません。）から生じる責任を負うものではありません。当社は、当社製品または当社製品が使用されたあらゆるシステムが、不正な改変、攻撃、ウイルス、干渉、ハッキング、データの破壊または窃盗その他の不正な侵入行為（「脆弱性問題」といいます。）によって影響を受けないことを保証しません。当社は、脆弱性問題に起因した場合はこれに関連して生じた損害について、一切責任を負いません。また、法令において認められる限りにおいて、本資料および当社ハードウェア／ソフトウェア製品について、商品性および特定目的との合致に関する保証ならびに第三者の権利を侵害しないことの保証を含め、明示または黙示のいかなる保証も行いません。
- 8. 当社製品をご使用の際は、最新の製品情報（データシート、ユーザーズマニュアル、アプリケーションノート、信頼性ハンドブックに記載の「半導体デバイスの使用上的一般的な注意事項」等）をご確認の上、当社が指定する最大定格、動作電源電圧範囲、放熱特性、実装条件その他指定条件の範囲内でご使用ください。指定条件の範囲を超えて当社製品をご使用された場合の故障、誤動作の不具合および事故につきましては、当社は、一切その責任を負いません。
- 9. 当社は、当社製品の品質および信頼性の向上に努めていますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は、データシート等において高信頼性、Harsh environment 向け製品と定義しているものを除き、耐放射線設計を行っておりません。仮に当社製品の故障または誤動作が生じた場合であっても、人身事故、火災事故その他社会的損害等を生じさせないよう、お客様の責任において、冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、お客様の機器・システムとしての出荷保証を行ってください。特に、マイコンソフトウェアは、単独での検証は困難なため、お客様の機器・システムとしての安全検証をお客様の責任で行ってください。
- 10. 当社製品の環境適合性等の詳細につきましては、製品個別に必ず当社営業窓口までお問合せください。ご使用に際しては、特定の物質の含有・使用を規制する RoHS 指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。かかる法令を遵守しないことにより生じた損害に関して、当社は、一切その責任を負いません。
- 11. 当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器・システムに使用することはできません。当社製品および技術を輸出、販売または移転等する場合は、「外国為替及び外国貿易法」その他日本国および適用される外国の輸出管理関連法規を遵守し、それらの定めるところに従い必要な手続きを行ってください。
- 12. お客様が当社製品を第三者に転売等される場合には、事前に当該第三者に対して、本ご注意書き記載の諸条件を通知する責任を負うものいたします。
- 13. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを禁じます。
- 14. 本資料に記載されている内容または当社製品についてご不明な点がございましたら、当社の営業担当者までお問合せください。
- 注 1. 本資料において使用されている「当社」とは、ルネサス エレクトロニクス株式会社およびルネサス エレクトロニクス株式会社が直接的、間接的に支配する会社をいいます。
- 注 2. 本資料において使用されている「当社製品」とは、注 1において定義された当社の開発、製造製品をいいます。

●

(Rev.5.0-1 2020.10)

本社所在地

〒135-0061 東京都江東区豊洲 3-2-24 (豊洲フォレシア)

www.renesas.com

お問合せ窓口

弊社の製品や技術、ドキュメントの最新情報、最寄の営業お問合せ窓口に関する情報などは、弊社ウェブサイトをご覧ください。

www.renesas.com/contact/

商標について

ルネサスおよびルネサスロゴはルネサス エレクトロニクス株式会社の商標です。すべての商標および登録商標は、それぞれの所有者に帰属します。