

セクション 16. アナログ/デジタル コンバータ (ADC)

ハイライト

本セクションには以下の主要項目を記載しています。

16.1 はじめに	16-2
16.2 制御レジスタ	16-6
16.3 サンプリング/変換シーケンスの概要	16-17
16.4 ADC のコンフィグレーション	16-27
16.5 ADC 割り込みの生成	16-33
16.6 変換するアナログ入力の選択	16-35
16.7 DMA 付きデバイスにおける変換結果のバッファリングの指定	16-44
16.8 ADC のコンフィグレーション例	16-48
16.9 1.1 Msps 用 ADC コンフィグレーション	16-49
16.10 DMA なしデバイスのサンプリング/変換シーケンス例	16-51
16.11 DMA 付きデバイスのサンプリング/変換シーケンス例	16-63
16.12 A/D サンプリングの要件	16-73
16.13 ADC 結果バッファの読み出し	16-74
16.14 変換関数	16-76
16.15 ADC の精度と誤差	16-78
16.16 接続に関する注意事項	16-78
16.17 スリープおよびアイドルモード時の動作	16-79
16.18 リセットの影響	16-79
16.19 特殊機能レジスタ	16-80
16.20 設計のヒント	16-81
16.21 関連アプリケーション ノート	16-82
16.22 改訂履歴	16-83
各国の営業所とサービス	16-88

Note: ファミリ リファレンス マニュアルの本セクションは、デバイス データシートの補足を目的としています。本セクションの内容は、dsPIC33F/PIC24H ファミリの一部のデバイスには対応していません。

本書の内容がお客様のご使用になるデバイスに対応しているかどうかは、最新デバイス データシート内の「**アナログ / デジタル コンバータ (ADC)**」の冒頭に記載している注意書きでご確認ください。

デバイス データシートとファミリ リファレンス マニュアルの各セクションは、マイクロチップ社のウェブサイト (<http://www.microchip.com>) からダウンロードできます。

16.1 はじめに

本書は、dsPIC33F/PIC24H ファミリのデバイスが内蔵する逐次比較型 (SAR) アナログ / デジタル コンバータ (ADC) の機能と動作モードについて説明します。

ユーザ アプリケーションは、ADC モジュールを 10 ビット 4 チャンネル ADC または 12 ビット 1 チャンネル ADC して機能するようにコンフィグレーションできます (12 ビットモードは一部のデバイスでのみ利用可能です)。

図 16-1 に、DMA 付きデバイスの ADC ブロック図を示します。図 16-2 に、DMA なしデバイスの ADC ブロック図を示します。

dsPIC33F/PIC24H の ADC モジュールは下記の特長を備えます。

- SAR 変換
- 最大 1.1 Msps の変換速度
- 最大 32 本のアナログ入力ピン
- 外部電圧リファレンス入力ピン
- 4 つのユニポーラ差動サンプル / ホールド (S&H) アンプ
- 最大 4 本のアナログ入力ピンの同時サンプリング
- 自動チャンネル スキャンモード
- 選択可能な変換トリガ源
- 最大 16 ワードの変換結果バッファ
- 選択可能なバッファ書き込みモード (一部のデバイスのみ)
- 周辺モジュール間接アドレッシングを含む DMA サポート (一部のデバイスのみ)
- CPU スリープおよびアイドルモード時の動作

デバイスのバージョンによって異なりますが、ADC モジュールは最大 32 本のアナログ入力ピン (AN0 ~ AN31) を備えます。これらのアナログ入力はマルチプレクサを介して 4 つの S&H アンプ (CH0 ~ CH3) に接続されます。アナログ入力マルチプレクサは 2 組の制御ビット MUXA (CHySA/CHyNA) と MUXB (CHySB/CHyNB) を持ちます。これらの制御ビットは変換を行うアナログ入力を選択します。MUXA および MUXB 制御ビットを交互に使用して変換用アナログ入力を選択できます。特定入力ピンを使用して全チャンネルでユニポーラ差動変換が可能です (図 16-1 と図 16-2 参照)。

CH0 S&H アンプではチャンネル スキャンモードを使用できます。ユーザ アプリケーションは、デバイスが備える全てのアナログ入力 (最大 AN0 ~ AN31) の任意のサブセットを選択できます。選択された入力は CH0 を使用して昇順に変換されます。

ADC モジュールは、複数 S&H チャンネルを使用する同時サンプリングをサポートします。同時サンプリングは、複数入力を同時にサンプリングした後、各チャンネルを逐次変換します。既定値では、複数チャンネルは逐次サンプリング / 変換されます。

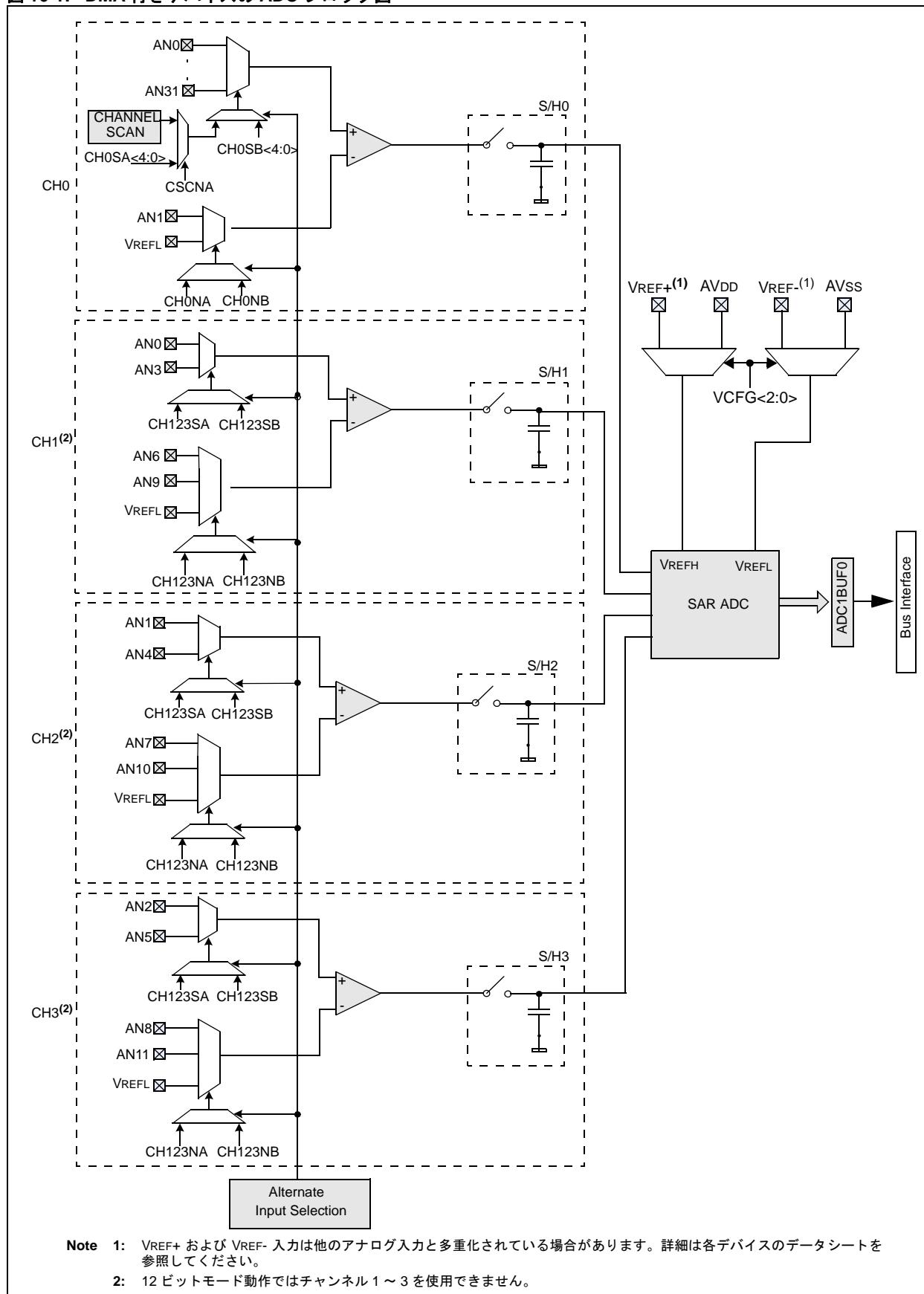
DMA を備えたデバイスの場合、ADC モジュールは 1 ワードの結果バッファに接続されます。結果バッファは 1 ワードですが、ADC モジュールと一緒に DMA を使用すると、CPU に負荷をかけずに複数変換結果を DMA RAM バッファに格納できます。各変換結果は、バッファからの読み出し時に 4 種類の 16 ビット出力フォーマットのいずれかに変換されます。

DMA を備えないデバイスの場合、ADC モジュールは 16 ワードの結果バッファに接続されます。ADC 結果は 4 種類の数値フォーマットで読み出せます (図 16-14)。

Note 1: MUXA および MUXB 制御ビットの表記には、S&H チャンネル番号指定するために添え字「y」を使用します (y = 0 または 123)。

2: デバイスのバージョンによって異なりますが、ADC モジュールは最大 32 本のアナログ入力ピン (AN0 ~ AN31) を備えます。これに加えて、外部電圧リファレンスの接続用に 2 本のアナログ入力ピン (VREF+, VREF-) を備えます。これらの電圧リファレンス入力は、他のアナログ入力ピンに多重化されている場合があります。アナログ入力ピンの数と、外部電圧リファレンス入力ピンのコンフィギュレーションはデバイスごとに異なります。詳細は各デバイスのデータシートを参照してください。

図 16-1: DMA 付きデバイスの ADC ブロック図





16.2 制御レジスタ

ADC モジュールは下記の 10 個の制御およびステータス レジスタを備えます。

- ADxCON1: ADCx 制御レジスタ 1(1)
- ADxCON2: ADCx 制御レジスタ 2(1)
- ADxCON3: ADCx 制御レジスタ 3(1)
- ADxCON4: ADCx 制御レジスタ 4(1,2)
- ADxCHS123: ADCx 入力チャンネル 1/2/3 選択レジスタ (1)
- ADxCHS0: ADCx 入力チャンネル 0 選択レジスタ (1)
- AD1CSSH: ADC1 入カスキャン選択レジスタ HIGH(1)
- ADxCSSL: ADCx 入カスキャン選択レジスタ LOW(1)
- AD1PCFGH: ADC1 ポート コンフィグレーション レジスタ HIGH(1,3)
- ADxPCFGL: ADCx ポート コンフィグレーション レジスタ LOW(1)

ADxCON1、ADxCON2、ADxCON3 レジスタは ADC モジュールの動作を制御します。ADxCON4 レジスタは、DMA 付きデバイスの Scatter/Gather モードで DMA バッファに保存する各アナログ入力の変換結果数を設定します。ADxCHS123 および ADxCHS0 レジスタは、S&H アンプに接続する入力ピンを選択します。ADCSSH/L レジスタは、逐次スキャンする入力を選択します。ADxPCFGH/L レジスタは、アナログ入力ピンをアナログ入力またはデジタル I/O として設定します。

16.2.1 ADC 結果バッファ

DMA 付きデバイスの場合、ADC モジュールは 1 ワードの結果バッファ (ADC1BUF0) を備えます。DMA なしデバイスの場合、ADC モジュールは結果バッファとして 16 ワードのデュアルポート RAM を備えます。16 ワードバッファの各バッファ位置は ADC1BUF0、ADC1BUF1、ADC1BUF2、..., ADC1BUFE、ADC1BUFF として参照されます。

Note: デバイスリセット後の ADC バッファレジスタの内容は未知です。

セクション 16. アナログ/デジタル コンバータ (ADC)

16

アナログ/デジタル
コンバータ (ADC)

レジスタ 16-1: ADxCON1: ADCx 制御レジスタ 1⁽¹⁾

R/W-0	U-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0
ADON	—	ADSIDL	ADDMABM ⁽³⁾	—	AD12B ⁽³⁾	FORM<1:0>	
bit 15							bit 8
R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0 HC,HS	R/C-0 HC, HS
SSRC<2:0>			—	SIMSAM	ASAM	SAMP	DONE
bit 7							bit 0
凡例: HC = ハードウェアでクリア HS = ハードウェアでセット C = クリアのみ可能ビット R = 読み出し可能ビット W = 書き込み可能ビット U = 未実装ビット、「0」として読み出し -n = POR 時の値 1 = ビットをセット 0 = ビットをクリア x = ビットは未知							

- bit 15 **ADON:** ADC 動作モードビット
1 = ADC モジュールを有効にする
0 = ADC モジュールを無効にする
- bit 14 **未実装:** 「0」として読み出し
- bit 13 **ADSIDL:** アイドルモード時停止ビット
1 = デバイスがアイドルモードに移行した時にモジュールの動作を停止する
0 = アイドルモード中もモジュールの動作を継続する
- bit 12 **ADDMABM:** DMA バッファビルド モードビット⁽³⁾
1 = 変換順にDMAバッファに書き込む(モジュールは非DMAスタンドアロンバッファに使用するアドレスと同じアドレスを DMA チャンネルに割り当てます)
0 = Scatter/Gather モードで DMA バッファに書き込む (モジュールはアナログ入力の番号と DMA バッファのサイズに基づいて Scatter/Gather アドレスを DMA チャンネルに割り当てます)
- bit 11 **未実装:** 「0」として読み出し
- bit 10 **AD12B:** 10 ビット /12 ビット動作モードビット⁽³⁾
1 = 12 ビット /1 チャンネル ADC 動作
0 = 10 ビット /4 チャンネル ADC 動作
- bit 9-8 **FORM<1:0>:** データ出力フォーマットビット
10 ビット動作の場合
11 = 符号付き小数 (DOUT = sddd dddd dd00 0000, s = 符号、d = データ)
10 = 符号なし小数 (DOUT = dddd dddd dd00 0000)
01 = 符号付き整数 (DOUT = ssss sssd dddd dddd, s = 符号、d = データ)
00 = 符号なし整数 (DOUT = 0000 00dd dddd dddd)
12 ビット動作の場合
11 = 符号付き小数 (DOUT = sddd dddd dddd 0000, s = 符号、d = データ)
10 = 符号なし小数 (DOUT = dddd dddd dddd 0000)
01 = 符号付き整数 (DOUT = ssss sddd dddd dddd, s = 符号、d = データ)
00 = 符号なし整数 (DOUT = 0000 dddd dddd dddd)
- bit 7-5 **SSRC<2:0>:** サンプルクロック源選択ビット
111 = 内部カウンタがサンプリング終了 / 変換開始をトリガする (自動変換)
110 = 予約
101 = モータ制御 PWM2 インターバルがサンプリング終了 / 変換開始をトリガする⁽²⁾
100 = GP タイマ (ADC1 では Timer5、ADC2 では Timer3) コンペアがサンプリング終了 / 変換開始をトリガする⁽³⁾
011 = モータ制御 PWM1 インターバルがサンプリング終了 / 変換開始をトリガする⁽²⁾
010 = GP タイマ (ADC1 では Timer3、ADC2 では Timer5) コンペアがサンプリング終了 / 変換開始をトリガする
001 = INT0 ピンがアクティブに遷移した時にサンプリング終了 / 変換開始をトリガする
000 = サンプリングビットがクリアされた時にサンプリング終了 / 変換開始をトリガする

Note 1: ADxCON1 および ADCx の「x」は ADC1 または ADC2 を指定します。

2: 一部のデバイスではこのクロック源を利用できません (詳細は各デバイスのデータシート参照)。

3: 一部のデバイスではこのビットを利用できません (詳細は各デバイスのデータシート参照)。

レジスタ 16-1: ADxCON1: ADCx 制御レジスタ 1⁽¹⁾ (続き)

bit 4	未実装: 「0」 として読み出し
bit 3	SIMSAM: 同時サンプリング選択ビット (CHPS<1:0> = 01 または 1x の場合にのみ適用) AD12B = 1 の場合、SIMSAM は U-0 (未実装、「0」 として読み出し) です。 1 = CH0、CH1、CH2、CH3 を同時にサンプリングする (CHPS<1:0> = 1x の場合)、または、 CH0 と CH1 を同時にサンプリングする (CHPS<1:0> = 01 の場合) 0 = 複数チャンネルを逐次サンプリングする
bit 2	ASAM: ADC サンプル自動開始ビット 1 = 変換後即座に次のサンプリングを開始する (SAMP ビットを自動的にセットする) 0 = 変換後次のサンプリングを自動的に開始しない (SAMP ビットを自動的にセットしない)
bit 1	SAMP: ADC サンプル イネーブルビット 1 = ADC S&H アンプはサンプリングを実行する 0 = ADC S&H アンプはサンプリングを待機する ASAM = 0 の場合、ソフトウェアで SAMP ビットに「1」を書き込む事によってサンプリングを開始 できます。ASAM = 1 の場合、ハードウェアが自動的に SAMP ビットをセットします。 SSRC = 000 の場合、ソフトウェアで SAMP ビットに「0」を書き込む事によってサンプリングを終 了して変換を開始できます。SSRC ≠ 000 の場合、サンプリングを終了して変換を開始するためにハー ドウェアが自動的に SAMP ビットをクリアします。
bit 0	DONE: ADC 変換ステータスビット 1 = A/D 変換サイクルが完了した 0 = A/D 変換は開始されていない、または変換を実行中 A/D 変換が完了すると、ハードウェアが自動的に DONE ビットをセットします。ソフトウェアで DONE ビットに「0」を書き込む事により、DONE ステータスをクリアできます (ソフトウェアで「1」を書 き込む事はできません)。このビットをクリアしても、実行中の動作には影響しません。ハードウェ アは次の変換の開始時に DONE ビットを自動的にクリアします。

Note 1: ADxCON1 および ADCx の「x」は ADC1 または ADC2 を指定します。

- 一部のデバイスではこのクロック源を利用できません (詳細は 各デバイスのデータシート参照)。
- 一部のデバイスではこのビットを利用できません (詳細は各デバイスのデータシート参照)。

セクション 16. アナログ/デジタル コンバータ (ADC)

16

アナログ/デジタル
コンバータ (ADC)

レジスタ 16-2: ADxCON2: ADCx 制御レジスタ 2⁽¹⁾

R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0
VCFG<2:0>			—	—	CSCNA	CHPS<1:0>	
bit 15							bit 8

R-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
BUFS	—	SMPI<3:0> ^(2,3)				BUFM	ALTS
bit 7			bit 0				

凡例:

R = 読み出し可能ビット W = 書き込み可能ビット U = 未実装ビット、「0」として読み出し
-n = POR 時の値 1 = ビットをセット 0 = ビットをクリア x = ビットは未知

bit 15-13 **VCFG<2:0>**: コンバータ電圧リファレンス コンフィグレーション ビット

	VREFH	VREFL
000	AVDD	AVss
001	外部 VREF+ ⁽⁴⁾	AVss
010	AVDD	外部 VREF- ⁽⁴⁾
011	外部 VREF+ ⁽⁴⁾	外部 VREF- ⁽⁴⁾
1xx	AVDD	AVss

bit 12-11 **未実装**: 「0」として読み出し

bit 10 **CSCNA**: 入力スキャン選択ビット

1 = サンプル A のサンプリング時に CH0+ の入力をスキャンする
0 = 入力をスキャンしない

bit 9-8 **CHPS<1:0>**: チャンネル選択ビット

AD12B = 1 の場合、CHPS<1:0> は U-0 (未実装、「0」として読み出し) です。
1x = CH0、CH1、CH2、CH3 を変換する
01 = CH0 と CH1 を変換する
00 = CH0 を変換する

bit 7 **BUFS**: バッファ書き込みステータスビット (BUFM = 1 の場合にのみ有効)

1 = ADC は現在バッファの後半部に書き込んでいる
この場合、ユーザ アプリケーションはバッファ前半部のデータにアクセスする必要があります。
0 = ADC は現在バッファの前半部に書き込んでいる
この場合、ユーザ アプリケーションはバッファ後半部のデータにアクセスする必要があります。

bit 6 **未実装**: 「0」として読み出し

Note 1: ADxCON2 および ADCx の「x」は ADC1 または ADC2 を指定します。

2: DMA 付きデバイスの場合、SMPI<3:0> ビットは「DMA アドレスのインクリメント頻度選択ビット」と呼ばれます。

3: DMA なしデバイスの場合、SMPI<3:0> ビットは「割り込み 1 回あたりのサンプル数選択ビット」と呼ばれます。

4: 一部のデバイスだけが VREF+ および VREF- ピンを備えます。詳細は各デバイスのデータシートを参照してください。

レジスタ 16-2: ADxCON2: ADCx 制御レジスタ 2⁽¹⁾ (続き)

bit 5-2 **SMPI<3:0>**: サンプリング / 変換動作ビット ^(2,3)

DMA 付きデバイスの場合

1111 =16 回のサンプリング / 変換動作が完了するごとに DMA アドレスをインクリメントする

1110 =15 回のサンプリング / 変換動作が完了するごとに DMA アドレスをインクリメントする

•

•

•

0001 =2 回のサンプリング / 変換動作が完了するごとに DMA アドレスをインクリメントする

0000 =1 回のサンプリング / 変換動作が完了するごとに DMA アドレスをインクリメントする

DMA なしデバイスの場合

1111 =16 回のサンプリング / 変換動作が完了するごとに ADC 割り込みを生成する

1110 =15 回のサンプリング / 変換動作が完了するごとに ADC 割り込みを生成する

•

•

•

0001 =2 回のサンプリング / 変換動作が完了するごとに ADC 割り込みを生成する

0000 =1 回のサンプリング / 変換動作が完了するごとに ADC 割り込みを生成する

bit 1 **BUFM**: バッファ書き込みモード選択ビット

1 = 最初の割り込みでバッファ前半部に書き込み、次の割り込みでバッファ後半部に書き込む

0 = 常に先頭アドレスからバッファの書き込みを開始する

bit 0 **ALTS**: 交互入力サンプルモード選択ビット

1 = 最初のサンプリングでサンプル A 用チャンネル入力選択を使用し、次のサンプリングでサンプル B 用チャンネル入力選択を使用する

0 = 常にサンプル A 用チャンネル入力選択を使用する

Note 1: ADxCON2 および ADCx の「x」は ADC1 または ADC2 を指定します。

2: DMA 付きデバイスの場合、SMPI<3:0> ビットは「DMA アドレスのインクリメント頻度選択ビット」と呼ばれます。

3: DMA なしデバイスの場合、SMPI<3:0> ビットは「割り込み 1 回あたりのサンプル数選択ビット」と呼ばれます。

4: 一部のデバイスだけが VREF+ および VREF- ピンを備えます。詳細は各デバイスのデータシートを参照してください。

レジスタ 16-3: ADxCON3: ADCx 制御レジスタ 3⁽¹⁾

R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ADRC	—	—	SAMC<4:0> ^(2,3)				
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ADCS<7:0>							
bit 7							bit 0

凡例:

R = 読み出し可能ビット W = 書き込み可能ビット U = 未実装ビット、「0」として読み出し
-n = POR 時の値 1 = ビットをセット 0 = ビットをクリア x = ビットは未知

bit 15 **ADRC:** ADC 変換クロック源ビット
1 = ADC の内部 RC クロックを使用する
0 = 分周したシステムクロックを使用する

bit 14-13 **未実装:** 「0」として読み出し

bit 12-8 **SAMC<4:0>:** 自動サンプリング時間ビット ^(2,3)
11111 = 31 TAD
•
•
•
00001 = 1 TAD
00000 = 0 TAD

bit 7-0 **ADCS<7:0>:** ADC 変換クロック選択ビット
11111111 = 予約
•
•
•
01000000 = 予約
00111111 = $T_{CY} \cdot (ADCS<7:0> + 1) = 64 \cdot T_{CY} = T_{AD}$
•
•
•
00000010 = $T_{CY} \cdot (ADCS<7:0> + 1) = 3 \cdot T_{CY} = T_{AD}$
00000001 = $T_{CY} \cdot (ADCS<7:0> + 1) = 2 \cdot T_{CY} = T_{AD}$
00000000 = $T_{CY} \cdot (ADCS<7:0> + 1) = 1 \cdot T_{CY} = T_{AD}$

Note 1: ADxCSSL および ADCx の「x」は ADC1 または ADC2 を指定します。

2: このビットは SSRC<2:0> ビット (ADxCON1<7:5>) = 111 の時にのみ使用されます。

3: SSRC<2:0> = 111 の場合、1 つの S&H チャンネルを使用する時または同時サンプリングを使用する時に SAMC<4:0> ビットを「00001」以上に設定する必要があります。複数の S&H チャンネルを逐次サンプリングで使用する場合、変換速度を最大限に上げるために SAMC<4:0> ビットを「00000」に設定する必要があります。

dsPIC33F/PIC24H ファミリ リファレンス マニュアル

レジスタ 16-4: ADxCON4: ADCx 制御レジスタ 4^(1,2)

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	—	—	DMABL<2:0>		
bit 7							bit 0

凡例:

R = 読み出し可能ビット W = 書き込み可能ビット U = 未実装ビット、「0」として読み出し
-n = POR 時の値 1 = ビットをセット 0 = ビットをクリア x = ビットは未知

bit 15-3 **未実装:** 「0」として読み出し

bit 2-0 **DMABL<2:0>:** アナログ入力あたりの DMA バッファ割り当てビット

111 = 各アナログ入力に 128 ワードバッファを割り当てる
110 = 各アナログ入力に 64 ワードバッファを割り当てる
101 = 各アナログ入力に 32 ワードバッファを割り当てる
100 = 各アナログ入力に 16 ワードバッファを割り当てる
011 = 各アナログ入力に 8 ワードバッファを割り当てる
010 = 各アナログ入力に 4 ワードバッファを割り当てる
001 = 各アナログ入力に 2 ワードバッファを割り当てる
000 = 各アナログ入力に 1 ワードバッファを割り当てる

Note 1: ADxCON4 および ADCx の「x」は ADC1 または ADC2 を指定します。

2: DMA なしデバイスはこのレジスタを備えません。詳細は各デバイスのデータシートを参照してください。

ADxCHS123: ADCx 入力チャンネル 1/2/3 選択レジスタ ⁽¹⁾

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	—	—	CH123NB<1:0>		CH123SB
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	—	—	CH123NA<1:0>		CH123SA
bit 7							bit 0

凡例:

R = 読み出し可能ビット W = 書き込み可能ビット U = 未実装ビット、「0」として読み出し
-n = POR 時の値 1 = ビットをセット 0 = ビットをクリア x = ビットは未知

bit 15-11 **未実装:** 「0」として読み出し

bit 10-9 **CH123NB<1:0>:** サンプル B のチャンネル 1/2/3 負極性入力選択ビット
AD12B = 1 の場合、CHxNB は U-0 (未実装、「0」として読み出し) です。
11 = CH1 負極性入力に AN9、CH2 負極性入力に AN10、CH3 負極性入力に AN11 を選択する
10 = CH1 負極性入力に AN6、CH2 負極性入力に AN7、CH3 負極性入力に AN8 を選択する
0x = CH1、CH2、CH3 負極性入力に VREFL を選択する

bit 8 **CH123SB:** サンプル B のチャンネル 1/2/3 正極性入力選択ビット
AD12B = 1 の場合、CHxSB は U-0 (未実装、「0」として読み出し) です。
1 = CH1 正極性入力に AN3、CH2 正極性入力に AN4、CH3 正極性入力に AN5 を選択する
0 = CH1 正極性入力に AN0、CH2 正極性入力に AN1、CH3 正極性入力に AN2 を選択する

bit 7-3 **未実装:** 「0」として読み出し

bit 2-1 **CH123NA<1:0>:** サンプル A のチャンネル 1/2/3 負極性入力選択ビット
AD12B = 1 の場合、CHxNA は U-0 (未実装、「0」として読み出し) です。
11 = CH1 負極性入力に AN9、CH2 負極性入力に AN10、CH3 負極性入力に AN11 を選択する
10 = CH1 負極性入力に AN6、CH2 負極性入力に AN7、CH3 負極性入力に AN8 を選択する
0x = CH1、CH2、CH3 負極性入力に VREFL を選択する

bit 0 **CH123SA:** サンプル A のチャンネル 1/2/3 正極性入力選択ビット
AD12B = 1 の場合、CHxSA は U-0 (未実装、「0」として読み出し) です。
1 = CH1 正極性入力に AN3、CH2 正極性入力に AN4、CH3 正極性入力に AN5 を選択する
0 = CH1 正極性入力に AN0、CH2 正極性入力に AN1、CH3 正極性入力に AN2 を選択する

Note 1: ADxCHS123 および ADCx の「x」は ADC1 または ADC2 を指定します。

dsPIC33F/PIC24H ファミリ リファレンス マニュアル

レジスタ 16-5: ADxCHS0: ADCx 入力チャンネル 0 選択レジスタ ⁽¹⁾

R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CH0NB	—	—	CH0SB<4:0> ⁽²⁾				
bit 15							bit 8

R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CH0NA	—	—	CH0SA<4:0> ^(2,3)				
bit 7							bit 0

凡例:

R = 読み出し可能ビット W = 書き込み可能ビット U = 未実装ビット、「0」として読み出し
 -n = POR 時の値 1 = ビットをセット 0 = ビットをクリア x = ビットは未知

bit 15 **CH0NB:** サンプル B のチャンネル 0 負極性入力選択ビット
 定義は bit 7 と同じ

bit 14-13 **未実装:** 「0」として読み出し

bit 12-8 **CH0SB<4:0>:** サンプル B のチャンネル 0 正極性入力選択ビット ⁽²⁾
 定義は bit<4:0> と同じ

bit 7 **CH0NA:** サンプル A のチャンネル 0 負極性入力選択ビット
 1 = チャンネル 0 負極性入力に AN1 を選択する
 0 = チャンネル 0 負極性入力に VREFL を選択する

bit 6-5 **未実装:** 「0」として読み出し

bit 4-0 **CH0SA<4:0>:** サンプル A のチャンネル 0 正極性入力選択ビット ^(2,3)
 11111 = チャンネル 0 正極性入力に AN31 を選択する
 11110 = チャンネル 0 正極性入力に AN30 を選択する
 •
 •
 •
 00010 = チャンネル 0 正極性入力に AN2 を選択する
 00001 = チャンネル 0 正極性入力に AN1 を選択する
 00000 = チャンネル 0 正極性入力に AN0 を選択する

- Note 1:** ADxCHS0 および ADCx の「x」は ADC1 または ADC2 を指定します。
2: ADC2 では AN16 ~ AN31 ピンを使用できません。
3: CSCNA ビット (ADxCON2<10>) = 1 の場合、これらのビットは効果を持ちません。

レジスタ 16-6: AD1CSSH: ADC1 入カスキャン選択レジスタ HIGH⁽¹⁾

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CSS31	CSS30	CSS29	CSS28	CSS27	CSS26	CSS25	CSS24
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CSS23	CSS22	CSS21	CSS20	CSS19	CSS18	CSS17	CSS16
bit 7							bit 0

凡例:

R = 読み出し可能ビット W = 書き込み可能ビット U = 未実装ビット、「0」として読み出し
-n = POR 時の値 1 = ビットをセット 0 = ビットをクリア x = ビットは未知

bit 15-0 **CSS<31:16>: ADC 入カスキャン選択ビット (2,3)**

1 = 入カスキャンに ANx を選択する
0 = 入カスキャンに ANx を選択しない

Note 1: DMA なしデバイスはこのレジスタを備えません。詳細は各デバイスのデータシートを参照してください。

2: ADC2 はアナログ入力 AN0 ~ AN15 のみをサポートします。このため ADC2 入カスキャン選択レジスタ HIGH は存在しません。

3: 最大で 16 個の入力を任意に選択してスキャンできます。

レジスタ 16-7: ADxCSSL: ADCx 入カスキャン選択レジスタ LOW⁽¹⁾

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CSS15 ⁽⁴⁾	CSS14 ⁽⁴⁾	CSS13 ⁽⁴⁾	CSS12	CSS11	CSS10	CSS9	CSS8
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CSS7	CSS6	CSS5	CSS4	CSS3	CSS2	CSS1	CSS0
bit 7							bit 0

凡例:

R = 読み出し可能ビット W = 書き込み可能ビット U = 未実装ビット、「0」として読み出し
-n = POR 時の値 1 = ビットをセット 0 = ビットをクリア x = ビットは未知

bit 15-0 **CSS<15:0>: ADC 入カスキャン選択ビット (2,3,4)**

1 = 入カスキャンに ANx を選択する
0 = 入カスキャンに ANx を選択しない

Note 1: ADxCSSL および ADCx の「x」は ADC1 または ADC2 を指定します。

2: アナログ入力数が 16 未満のデバイスでもユーザは全ての ADxCSSL ビットを選択できます。ただし、デバイス上に対応する入力が存在しないビットをスキャンに選択すると VREF- が変換されます。

3: 最大で 16 個の入力を任意に選択してスキャンできます。

4: DMA なしデバイスはこのビットを備えません。詳細は各デバイスのデータシートを参照してください。

dsPIC33F/PIC24H ファミリ リファレンス マニュアル

レジスタ 16-8: AD1PCFGH: ADC1 ポート コンフィグレーション レジスタ HIGH^(1,3)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PCFG31	PCFG30	PCFG29	PCFG28	PCFG27	PCFG26	PCFG25	PCFG24
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PCFG23	PCFG22	PCFG21	PCFG20	PCFG19	PCFG18	PCFG17	PCFG16
bit 7							bit 0

凡例:

R = 読み出し可能ビット W = 書き込み可能ビット U = 未実装ビット、「0」として読み出し
-n = POR 時の値 1 = ビットをセット 0 = ビットをクリア x = ビットは未知

bit 15-0 **PCFG<31:16>:** ADC ポート コンフィグレーション制御ビット ^(1,2)

- 1 = ポートピンをデジタルモードで使用する (ポートの入力読み出しを有効にし、ADC 入力マルチプレクサを AVss に接続する)
- 0 = ポートピンをアナログモードで使用する (ポートの入力読み出しを無効にし、ADC はピン電圧をサンプリングする)

- Note 1:** DMA なしデバイスはこのレジスタを備えません。詳細は各デバイスのデータシートを参照してください。
- 2:** アナログ入力数が 32 未満のデバイスでもユーザは全ての PCFG ビットを読み書きできます。しかし、デバイス上に対応する入力が存在しないポートの PCFG ビットは無視されます。
- 3:** ADC2 はアナログ入力 AN0 ~ AN15 のみをサポートします。このため ADC2 ポート コンフィグレーション レジスタ HIGH は存在しません。

レジスタ 16-9: ADxPCFGL: ADCx ポート コンフィグレーション レジスタ LOW⁽¹⁾

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PCFG15 ⁽⁴⁾	PCFG14 ⁽⁴⁾	PCFG13 ⁽⁴⁾	PCFG12	PCFG11	PCFG10	PCFG9	PCFG8
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PCFG7	PCFG6	PCFG5	PCFG4	PCFG3	PCFG2	PCFG1	PCFG0
bit 7							bit 0

凡例:

R = 読み出し可能ビット W = 書き込み可能ビット U = 未実装ビット、「0」として読み出し
-n = POR 時の値 1 = ビットをセット 0 = ビットをクリア x = ビットは未知

bit 15-0 **PCFG<15:0>:** ADC ポート コンフィグレーション制御ビット ^(2,3)

- 1 = ポートピンをデジタルモードで使用する (ポートの入力読み出しを有効にし、ADC 入力マルチプレクサを AVss に接続する)
- 0 = ポートピンをアナログモードで使用する (ポートの入力読み出しを無効にし、ADC はピン電圧をサンプリングする)

- Note 1:** ADxPCFGL および ADCx の「x」は ADC1 または ADC2 を指定します。
- 2:** アナログ入力数が 16 未満のデバイスでもユーザは全ての PCFG ビットを読み書きできます。しかし、デバイス上に対応する入力が存在しないポートの PCFG ビットは無視されます。
- 3:** 2 つの ADC モジュールを備えるデバイスでは、AD1PCFGL と AD2PCFGL の両方が、AN0 ~ AN15 に多重化されたポートピンのコンフィグレーションに影響します。
- 4:** DMA なしデバイスはこのビットを備えません。詳細は各デバイスのデータシートを参照してください。

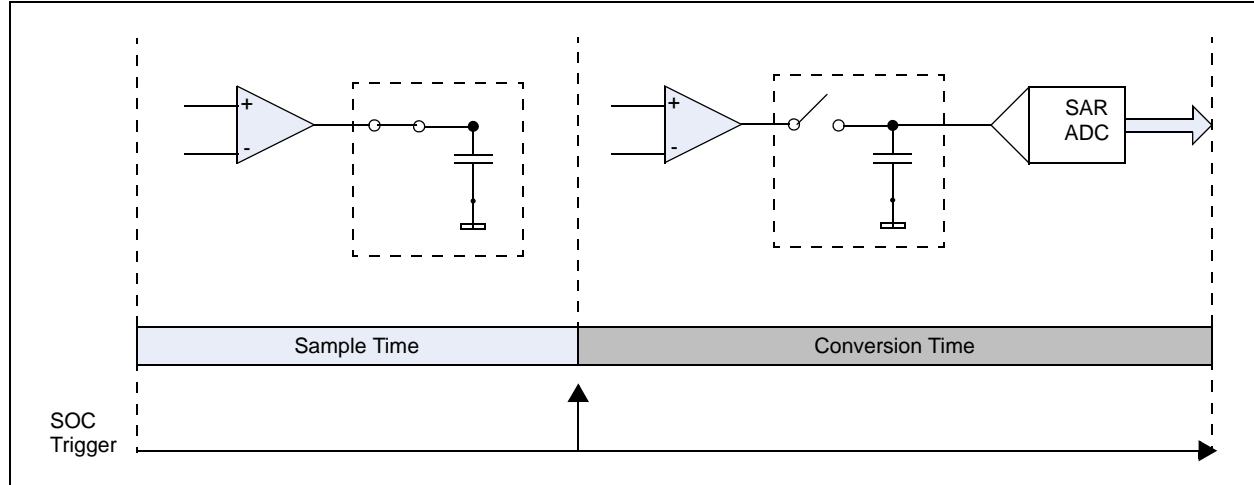
16.3 サンプルング/変換シーケンスの概要

図 16-3 に、A/D 変換が下記の 3 段階のプロセスで実行される事を示します。

1. 入力電圧信号をサンプルング コンデンサに接続する
2. サンプルング コンデンサを入力から切断する
3. コンデンサの充電電圧を等価デジタル値に変換する

2つのフェイズ(サンプルングと変換)は別々に制御されます。

図 16-3: サンプルング/変換シーケンス



16.3.1 サンプルング時間

サンプルング時間とは、選択したアナログ入力をサンプルング コンデンサに接続する時間の事です。S&H アンプが要求精度を満たす事を保証する最小サンプルング時間が存在します (16.12「A/D サンプルングの要件」参照)。

Note: ADC モジュールは、変換トリガを検出した後またはサンプルング プロセスを停止した後に変換を開始するまでに一定数の A/D クロックサイクルを必要とします。詳細は、各デバイス データシートの「電気的特性」に記載されている TPCS パラメータを参照してください。

サンプルング フェイズは、変換後に自動的に開始するか、あるいは ADC 制御レジスタ 1 (ADxCON1<1>) のサンプルング ビット (SAMP) を手動でセットする事によって開始できます。どちらの方法でサンプルング フェイズを制御するかは、ADC 制御レジスタ 1 (ADxCON1<2>) の自動サンプルング ビット (ASAM) で選択できます。表 16-1 に、ASAM ビットの設定値とサンプルング開始方法の関係を示します。

表 16-1: サンプルング開始方法の選択

ASAM	サンプルング開始方法
0	手動サンプルング
1	自動サンプルング

自動サンプルングを有効にした場合、ADC モジュールのサンプルング時間 (TSMP) は、SAMC<4:0> ビット (ADxCON3<12:8>) で定義された TAD サイクル数と等しくなります (式 16-1 参照)。

式 16-1: サンプルング時間の計算式

$$T_{SMP} = SAMC<4:0> \cdot T_{AD}$$

手動サンプルングを有効にした場合、ユーザ ソフトウェアは適正なサンプルング時間を確保するために十分な時間を提供する必要があります。

16.3.2 変換時間

変換開始 (SOC) トリガは、サンプリングを終了して A/D 変換を開始します。変換フェイズ中、サンプリング コンデンサはマルチプレクサから切断され、充電電圧が等価デジタル値に変換されます。10 ビットモードと 12 ビットモードの変換時間の計算式を式 16-2 と式 16-3 に示します。サンプリング時間と変換時間の合計が総変換時間となります。

正しく A/D 変換を行うには、最小 TAD 時間を確保できるように A/D 変換クロック (TAD) を選択する必要があります。10 ビットまたは 12 ビットモードでの最小 TAD 仕様値については、各デバイス データシートの「電気的特性」を参照してください。

式 16-2: 10 ビット ADC の変換時間

$$T_{CONV} = 12 \cdot T_{AD}$$

T_{CONV} = 変換時間
 T_{AD} = ADC クロック周期

式 16-3: 12 ビット ADC の変換時間

$$T_{CONV} = 14 \cdot T_{AD}$$

T_{CONV} = 変換時間
 T_{AD} = ADC クロック周期

各種ハードウェアから SOC を供給するか、あるいはユーザ ソフトウェアで SOC を手動制御する事ができます。変換開始 (SOC) トリガ源の選択には、ADC 制御レジスタ (ADxCON1<7:5>) の SOC トリガ源選択ビット (SSRC<2:0>) を使用します。表 16-2 に、SSRC<2:0> ビットの設定値と変換開始 (SOC) トリガ源の関係を示します。

Note: 12 ビットモードは一部のデバイスでのみ利用できます。詳細は各デバイスのデータシートを参照してください。

表 16-2: SOC トリガの選択

SSRC<2:0> ⁽¹⁾	SOC トリガ源
000	手動トリガ
001	外部割り込みトリガ (INT0)
010	タイマ割り込みトリガ
011	モータ制御用 PWM 特殊イベントトリガ
100	タイマ割り込みトリガ
111	自動トリガ

Note 1: ADC モジュールが動作している時に SSRC<2:0> 選択ビットを変更しないでください。

表 16-3 に、各種サンプリング フェイズおよび変換フェイズの選択に対応するサンプリング / 変換シーケンスを示します。

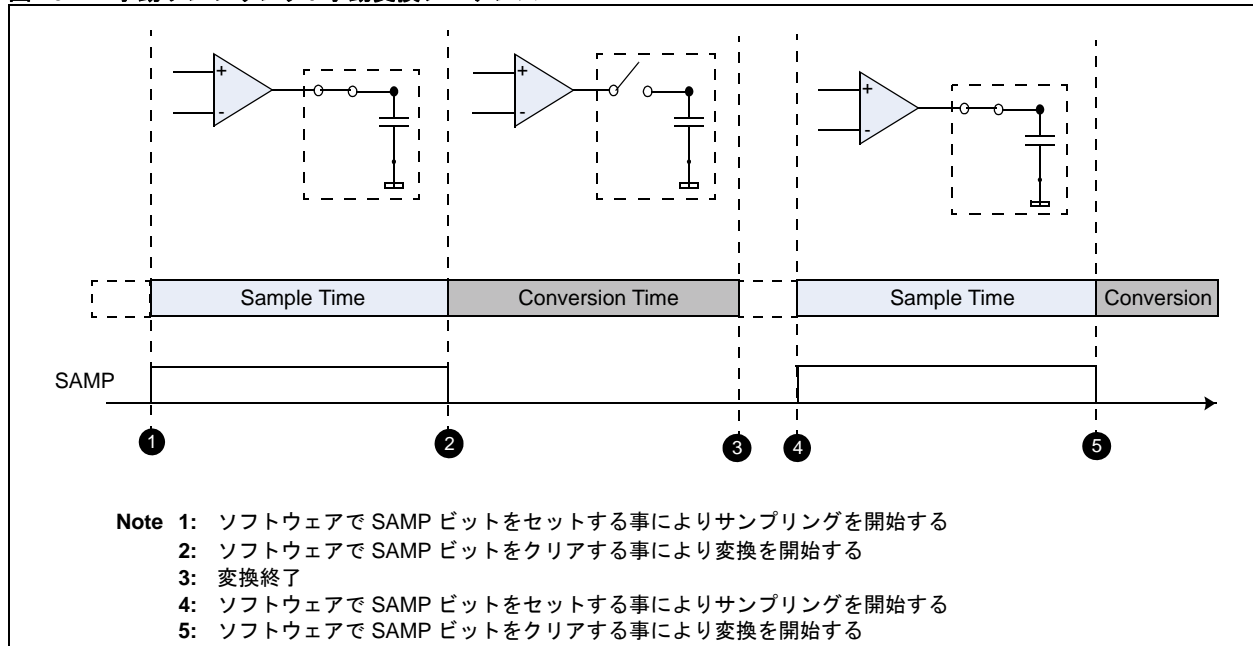
表 16-3: サンプリング/変換シーケンスの選択

ASAM	SSRC<2:0>	内容
0	000	手動サンプリング / 手動変換シーケンス
0	111	手動サンプリング / 自動変換シーケンス
0	001 010 011 100	手動サンプリング / トリガ変換シーケンス
1	000	自動サンプリング / 手動変換シーケンス
1	111	自動サンプリング / 自動変換シーケンス
1	001 010 011 100	自動サンプリング / トリガ変換シーケンス

16.3.3 手動サンプリング/手動変換シーケンス

手動サンプリング/手動変換シーケンスでは、ADC 制御レジスタ 1 (ADxCON1<1>) のサンプリング ビット (SAMP) をセットする事によってサンプリングを開始し、同ビットをクリアする事によってサンプリングを終了して変換を開始します (図 16-4 参照)。ユーザアプリケーションは、入力信号のサンプリングに適正な時間を確保できるように、SAMP ビットをセット/クリアする必要があります。例 16-1 に、手動サンプリング/手動変換用のコードシーケンスを示します。

図 16-4: 手動サンプリング/手動変換シーケンス



例 16-1: 手動サンプリング/手動変換用のコードシーケンス

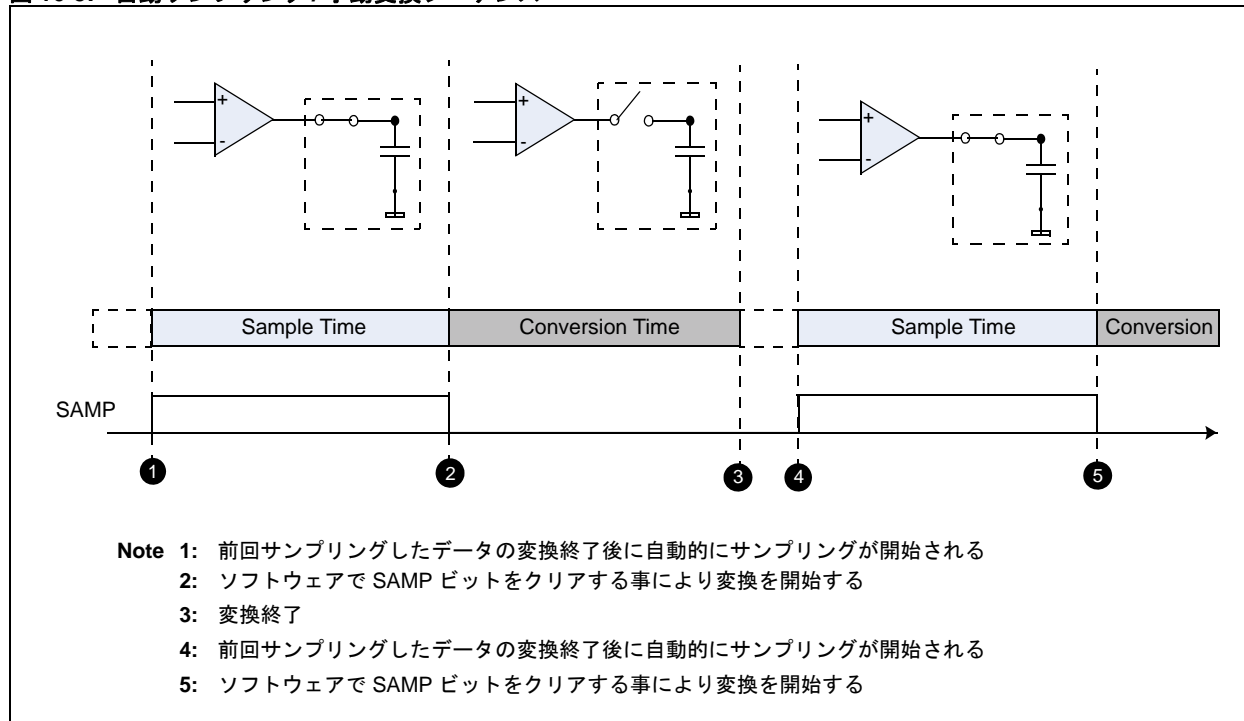
```
AD1CON1bits.SAMP = 1; // Start sampling
DelayUs(10);          // Wait for sampling time (10us)
AD1CON1bits.SAMP = 0; // Start the conversion
while (!AD1CON1bits.DONE); // Wait for the conversion to complete
ADCValue = ADC1BUF0; // Read the conversion result
```

Note: ADC モジュールの内部的な遅延により、変換開始後一定時間が経過した後にユーザソフトウェアは SAMP ビットを「0」として読み出します。一般的に、この遅延時間は 2 T_{cy} です。

16.3.4 自動サンプリング/手動変換シーケンス

自動サンプリング / 手動変換シーケンスでは、前回サンプリングしたデータの変換が終了すると自動的に次のサンプリングが開始されます。ユーザアプリケーションは、SAMP ビットをクリアする前に十分な時間をサンプリングに割り当てる必要があります。SAMP ビットをクリアすると変換が開始されます (図 16-5 参照)。

図 16-5: 自動サンプリング/手動変換シーケンス



例 16-2: 自動サンプリング/手動変換用のコードシーケンス

```
while (1) // Repeat continuously
{
    DelayNmSec(100); // Sample for 100 ms
    AD1CON1bits.SAMP = 0; // Start converting
    while (!AD1CON1bits.DONE; // Conversion done?
    AD1CON1bits.DONE = 0); // Clear conversion done status bit
    ADCValue = ADC1BUF0; // If yes, then get the ADC value
} // Repeat
```

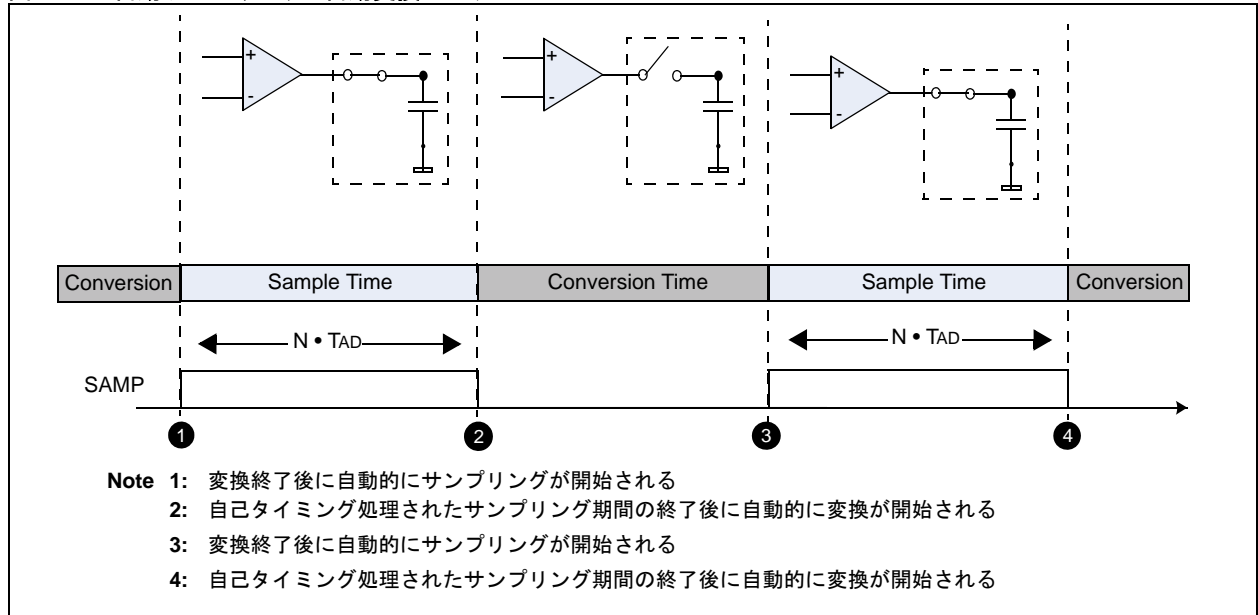
16.3.5 自動サンプリング/自動変換シーケンス

16.3.5.1 クロック同期変換トリガ

この自動変換シーケンスは、アナログ入力のサンプリング / 変換プロセスをさらに自動化します (図 16-6 参照)。サンプリング期間は自動的にタイミング処理され、サンプリング期間が終了すると自動的に変換が開始されます。ADxCON3 レジスタ (ADxCON3<12:8>) の自動サンプリング時間ビット (SAMC<4:0>) では、サンプリング期間として 0 ~ 31 の ADC クロックサイクル (TAD) 数を選択できます。最小推奨サンプリング時間 (SAMC 値) については、各デバイスデータシートの「電気的特性」を参照してください。

サンプリング終了 / 変換開始をトリガするサンプリングクロック源として内部カウンタを選択するために SSRC<2:0> ビットを「111」に設定します。

図 16-6: 自動サンプリング/自動変換シーケンス

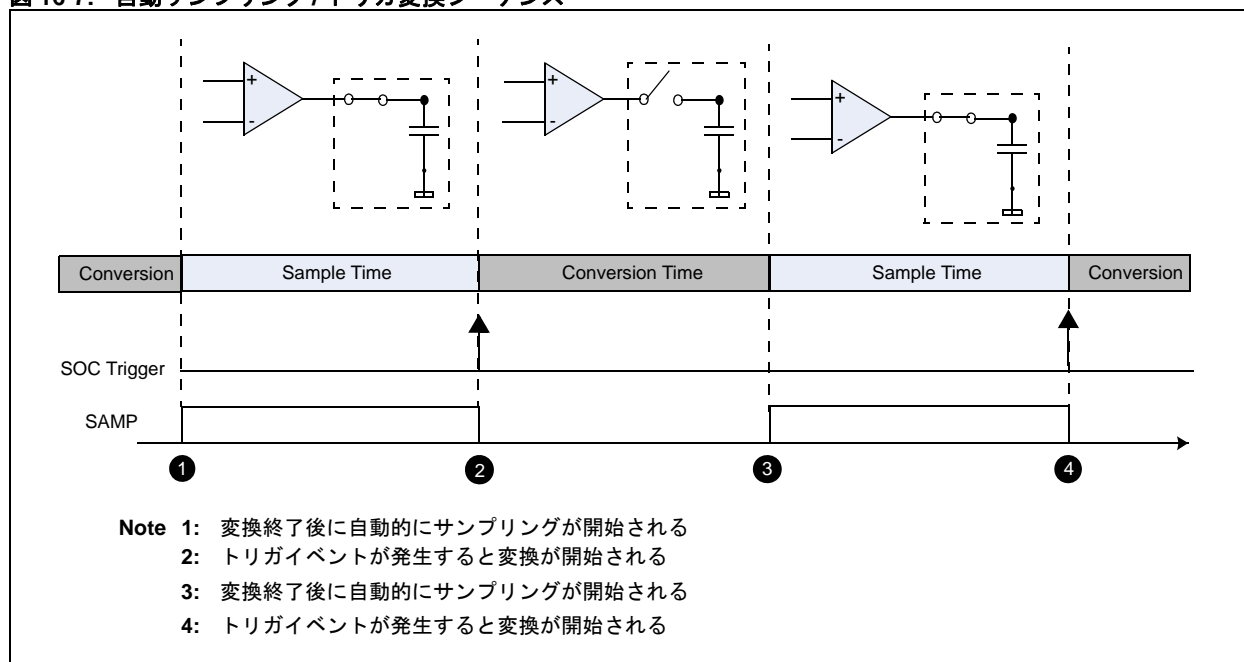


16.3.5.2 外部変換トリガ

自動サンプリング / トリガ変換シーケンスでは、変換終了後にサンプリングが自動的に開始され、選択された周辺モジュールからのトリガイイベントによって変換が開始されます (図 16-7 参照)。これにより、AD 変換を内部または外部のイベントに同期させる事ができます。外部変換トリガを選択するには、SSRC<2:0> ビットを「001」、「010」、「011」のいずれかに設定します。各種の外部変換トリガ源については 16.4.7「変換トリガ源」を参照してください。

A/D コンバータの動作中に ASAM ビットを変更しない必要があります。自動サンプリングが必要な場合、モジュールを有効化する前に ASAM ビットをセットする必要があります。A/D モジュールが安定するまでに一定の時間が必要です (各デバイス データシート内の T_{PDU} 参照)。このため、自動サンプリングを有効にした場合、ADC モジュールが安定する前の最初の ADC 結果の精度は保証されません。A/D クロックの速度によっては、最初の ADC 結果を破棄する必要があります。

図 16-7: 自動サンプリング / トリガ変換シーケンス



16.3.6 多チャンネルのサンプリング / 変換シーケンス

一般的に多チャンネル A/D コンバータは、入力マルチプレクサを使用して各入力チャンネルを逐次変換します。複数信号の同時サンプリングでは、全てのアナログ入力で厳密に同時にサンプリングが発生します (図 16-8 参照)。

特に複数チャンネル間の位相情報を必要とするアプリケーション等では同時サンプリングが必要です。逐次サンプリングは、各アナログ入力で変換を開始する直前にサンプリングを行います (図 16-8 参照)。複数入力のサンプリングにはお互い関連はありません。例えばモータ制御や電力監視アプリケーションでは、電圧と電流を計測して両者間の位相角度を評価する必要があります。

図 16-8: 同時サンプリングと逐次サンプリング

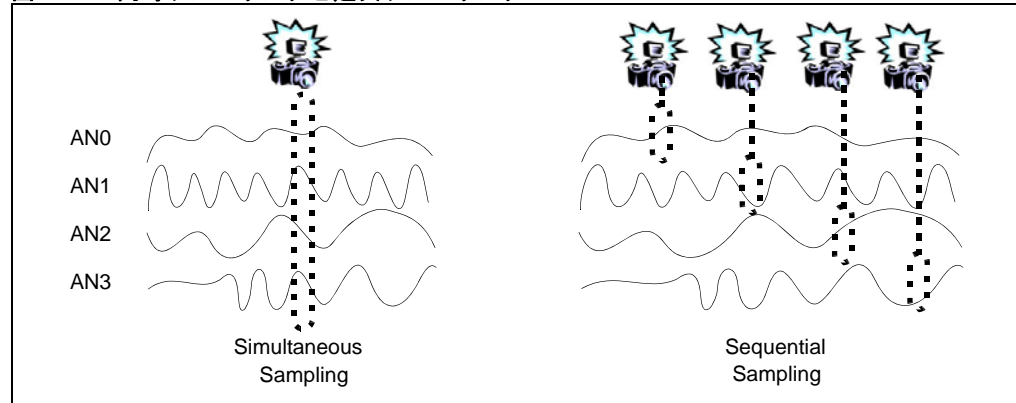


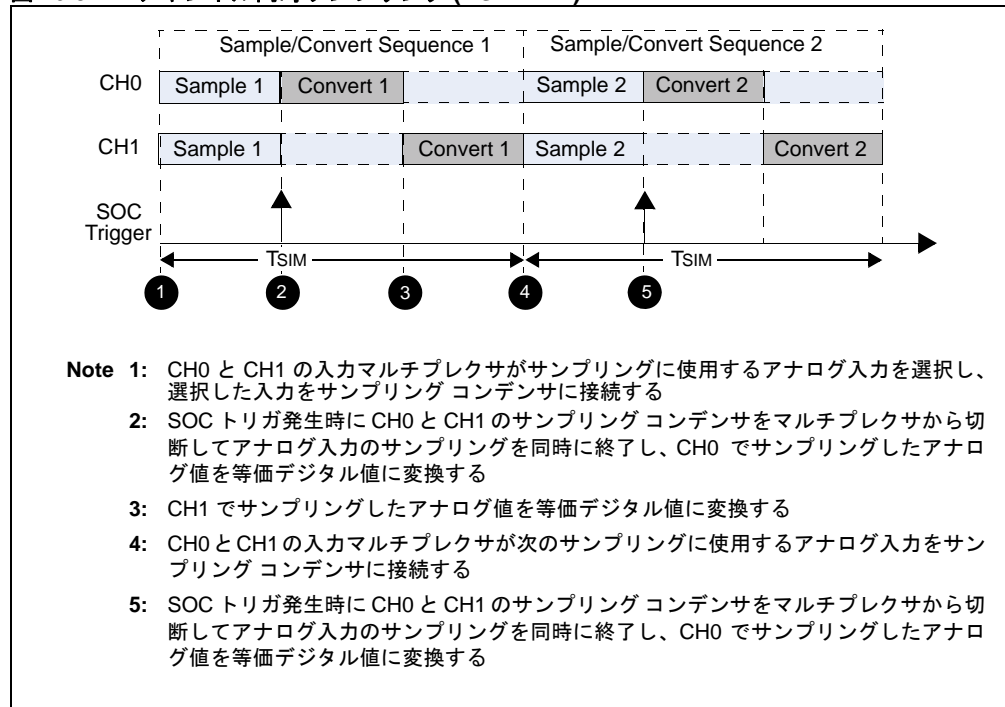
図 16-9 と図 16-10 に、ADC モジュールがサポートする同時サンプリングを示します。同時サンプリングでは、2 または 4 つの S&H チャンネルを使用して複数入力を同時にサンプリングし、その後で各チャンネルの変換を逐次実行します。

同時サンプリング モードは、ADC 制御レジスタ 1 (ADxCON1<3>) の同時サンプリング ビット (SIMSAM) で選択できます。既定値では、チャンネルは逐次サンプリング / 変換されます。表 16-4 に、SIMSAM ビットの設定値とサンプリング モードの関係を示します。逐次または同時サンプリングを行うチャンネルの選択には CHPS<1:0> ビットを使用します。

表 16-4: サンプリング モードの選択

SIMSAM	サンプリング モード
0	逐次サンプリング
1	同時サンプリング

図 16-9: 2 チャンネル同時サンプリング (ASAM = 1)



同時サンプリングを使用して複数チャンネルをサンプリング / 変換するのに要する総時間を式 16-4 に示します。

式 16-4: 複数チャンネルのサンプリング / 変換総時間 (同時サンプリングを選択した場合)

$$T_{SIM} = T_{SMP} + (M \cdot T_{CONV})$$

T_{SIM} = 同時サンプリング モードで複数チャンネルをサンプリング / 変換するのに要する総時間

T_{SMP} = サンプリング時間 (式 16-1 参照)

T_{CONV} = 変換時間 (式 16-2 参照)

M = チャンネル数 (CHPS<1:0> ビットで選択)

図 16-10: 4 チャンネル同時サンプリング

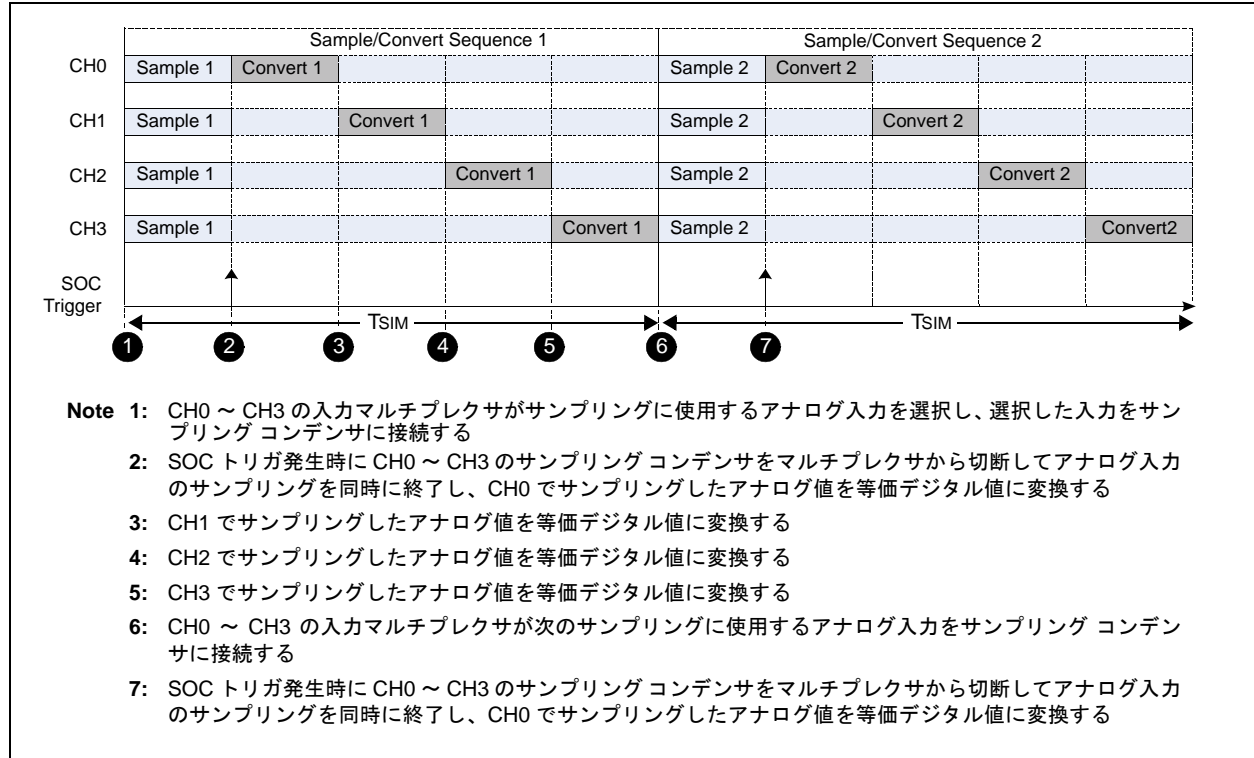


図 16-11 と図 16-12 に、既定値で選択される逐次サンプリングを示します。逐次サンプリングでは、複数チャンネルを逐次サンプリング / 変換します。

逐次サンプリングを使用して複数チャンネルをサンプリング / 変換するのに要する総時間を式 16-5 に示します。

式 16-5: 複数チャンネルのサンプリング / 変換総時間 (逐次サンプリングを選択した場合)

$T_{SMP} < T_{CONV}$ の場合

$$T_{SEQ} = M \cdot T_{CONV} \quad (M > 1 \text{ の場合})$$

$$T_{SEQ} = T_{SMP} + T_{CONV} \quad (M = 1 \text{ の場合})$$

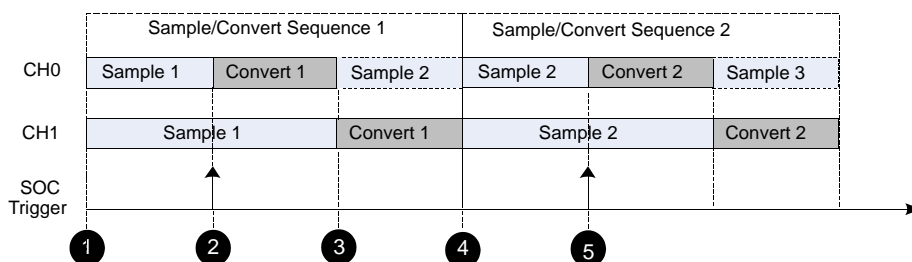
T_{SEQ} = 逐次サンプリング モードで複数チャンネルをサンプリング / 変換するのに要する総時間

T_{CONV} = 変換時間 (式 16-2 参照)

T_{SMP} = サンプリング時間 (式 16-1 参照)

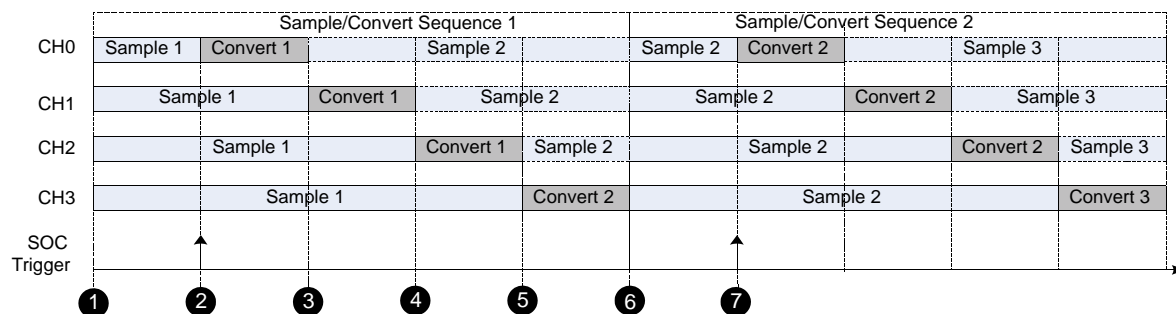
M = チャンネル数 (CHPS<1:0> ビットで選択)

図 16-11: 2 チャンネル逐次サンプリング (ASAM = 1)



- Note**
- 1: CH0 と CH1 の入力マルチプレクサがサンプリングに使用するアナログ入力を選択し、選択した入力をサンプリング コンデンサに接続する
 - 2: SOC トリガ発生時に CH0 のサンプリング コンデンサをマルチプレクサから切断して入力電圧を一定に保持し、CH0 でサンプリングしたアナログ値を等価デジタル値に変換する
 - 3: 変換後に CH0 のマルチプレクサ出力をサンプリング コンデンサに接続すると共に、CH1 のサンプリング コンデンサをマルチプレクサから切断して入力電圧を一定に保持し、CH1 でサンプリングしたアナログ値を等価デジタル値に変換する
 - 4: 変換後に CH1 のマルチプレクサ出力をサンプリング コンデンサに接続し、CH0 と CH1 の入力マルチプレクサが次のサンプリングに使用するアナログ入力を選択する
 - 5: SOC トリガ発生時に CH0 のサンプリング コンデンサをマルチプレクサから切断して入力電圧を一定に保持し、CH0 でサンプリングしたアナログ値を等価デジタル値に変換する

図 16-12: 4 チャンネル逐次サンプリング



- Note**
- 1: CH0 ~ CH3 の入力マルチプレクサがサンプリングに使用するアナログ入力を選択し、選択した入力をサンプリング コンデンサに接続する
 - 2: SOC トリガ発生時に CH0 のサンプリング コンデンサをマルチプレクサから切断して入力電圧を一定に保持し、CH0 でサンプリングしたアナログ値を等価デジタル値に変換する
 - 3: 変換後に CH0 のマルチプレクサ出力をサンプリング コンデンサに接続すると共に、CH1 のサンプリング コンデンサをマルチプレクサから切断して入力電圧を一定に保持し、CH1 でサンプリングしたアナログ値を等価デジタル値に変換する
 - 4: 変換後に CH1 のマルチプレクサ出力をサンプリング コンデンサに接続すると共に、CH2 のサンプリング コンデンサをマルチプレクサから切断して入力電圧を一定に保持し、CH2 でサンプリングしたアナログ値を等価デジタル値に変換する
 - 5: 変換後に CH2 のマルチプレクサ出力をサンプリング コンデンサに接続すると共に、CH3 のサンプリング コンデンサをマルチプレクサから切断して入力電圧を一定に保持し、CH3 でサンプリングしたアナログ値を等価デジタル値に変換する
 - 6: 変換後に CH3 のマルチプレクサ出力をサンプリング コンデンサに接続し、CH0 ~ CH3 の入力マルチプレクサが次のサンプリングに使用するアナログ入力を選択する
 - 7: SOC トリガ発生時に CH0 のサンプリング コンデンサをマルチプレクサから切断して入力電圧を一定に保持し、CH0 でサンプリングしたアナログ値を等価デジタル値に変換する

16.4 ADC のコンフィグレーション

16.4.1 ADC 動作モードの選択

ADC 制御レジスタ 1 (ADxCON1<10>) の 12 ビット動作モードビット (AD12B) の設定により、ADC モジュールを 10 ビット /4 チャンネル ADC (既定値設定) または 12 ビット /1 チャンネル ADC として動作させる事ができます。表 16-5 に、ADC12B ビットの設定値と ADC 動作モードの関係を示します。

- Note 1:** ADC12B ビットを変更する場合、あらかじめ ADC モジュールを無効化する必要があります。
- 2:** 12 ビットモードは一部のデバイスでのみ利用できます。詳細は各デバイスのデータシートを参照してください。

表 16-5: ADC 動作モード

AD12B	動作モード
0	10 ビット /4 チャンネル ADC
1	12 ビット /1 チャンネル ADC

16.4.2 ADC チャンネルの選択

10 ビットモード (AD12B = 0) では、ユーザ アプリケーションで ADC 制御レジスタ (ADxCON2<9:8>) のチャンネル選択ビット (CHPS<1:0>) を設定する事により、1 チャンネル (CH0)、2 チャンネル (CH0 と CH1)、4 チャンネル (CH0 ~ CH3) モードを選択できます。12 ビットモードを選択した場合、ユーザ アプリケーションは CH0 のみを使用できます。表 16-6 に、CHPS<1:0> ビットの設定値とチャンネル選択の関係を示します。

表 16-6: 10 ビット ADC チャンネルの選択

CHPS<1:0>	チャンネル選択
00	CH0
01	2 チャンネル (CH0 と CH1)
1x	4 チャンネル (CH0 ~ CH3)

16.4.3 電圧リファレンスの選択

A/D 変換用電圧リファレンスの選択には、ADC 制御レジスタ (ADxCON2<15:13>) の電圧リファレンス コンフィグレーション ビット (VCFG<2:0>) を使用します。ADC モジュールへ供給する電圧リファレンス HIGH (VREFH) と電圧リファレンス LOW (VREFL) は、内部の AVDD/AVSS 電源レールまたは外部 VREF+/VREF- 入力ピンから供給できます。少ピンデバイスでは、外部電圧リファレンスピンが AN0 および AN1 入力と多重化されている事があります。そのような場合でも、ADC モジュールは VREF+/VREF- 入力と多重化されたアナログ入力ピンを使用して A/D 変換を実行できます。外部リファレンスピンへの供給電圧は仕様を満たす必要があります。詳細は各デバイス データシートの「電気的特性」を参照してください。また、VREF+/VREF- ピンを利用できるかどうかについても、各デバイスのデータシート参照してください。

表 16-7: 電圧リファレンスの選択

VCFG<2:0>	VREFH	VREFL
000	AVDD	AVSS
001	VREF+	AVSS
010	AVDD	VREF-
011	VREF+	VREF-
1xx	AVDD	AVSS

16.4.4 ADC クロックの選択

ADC モジュールのクロック源には、命令サイクルクロック (TCY) または専用の内部 RC クロックを使用できます (図 16-13 参照)。命令サイクルクロックを使用する場合、クロック分周器で命令サイクルクロックを分周する事によって低いクロック周波数を選択できます。ADC 制御レジスタ (ADxCON3<7:0>) の ADC 変換クロック選択ビット (ADCS<7:0>) では、クロック分周比を 1:1 ~ 1:64 の範囲で 64 段階に選択できます。

A/D 変換を正しく行うには、75 ns 以上の ADC クロック周期 (TAD) が必要です。

式 16-6 に、ADC クロック周期 (TAD) を ADCS 制御ビットと命令サイクルクロック周期 (TCY) の関数として示します。

式 16-6: ADC クロック周期

ADRC = 0 の場合

$$ADC \text{ クロック周期} (TAD) = TCY \cdot (ADCS + 1)$$

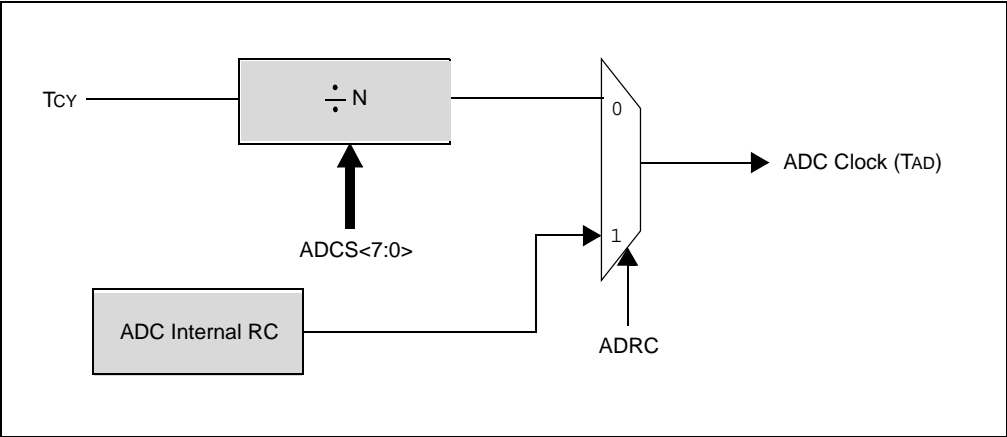
ADRC = 1 の場合

$$ADC \text{ クロック周期} (TAD) = TADRC$$

ADC モジュールは A/D 変換用の内部 RC クロック源を備えます。この内部 RC クロック源を使用すると、デバイスのスリープモード中に A/D 変換を実行できます。内部 RC オシレータは、ADC 制御レジスタ 3 (ADxCON1<15>) の ADC 変換クロック源ビット (ADRC) をセットする事により選択できます。ADRC ビットをセットした場合、ADCS<7:0> ビットは ADC の動作に影響しません。

Note: ADRC 周波数の仕様値は各デバイスのデータシートを参照してください。

図 16-13: ADC クロックの生成



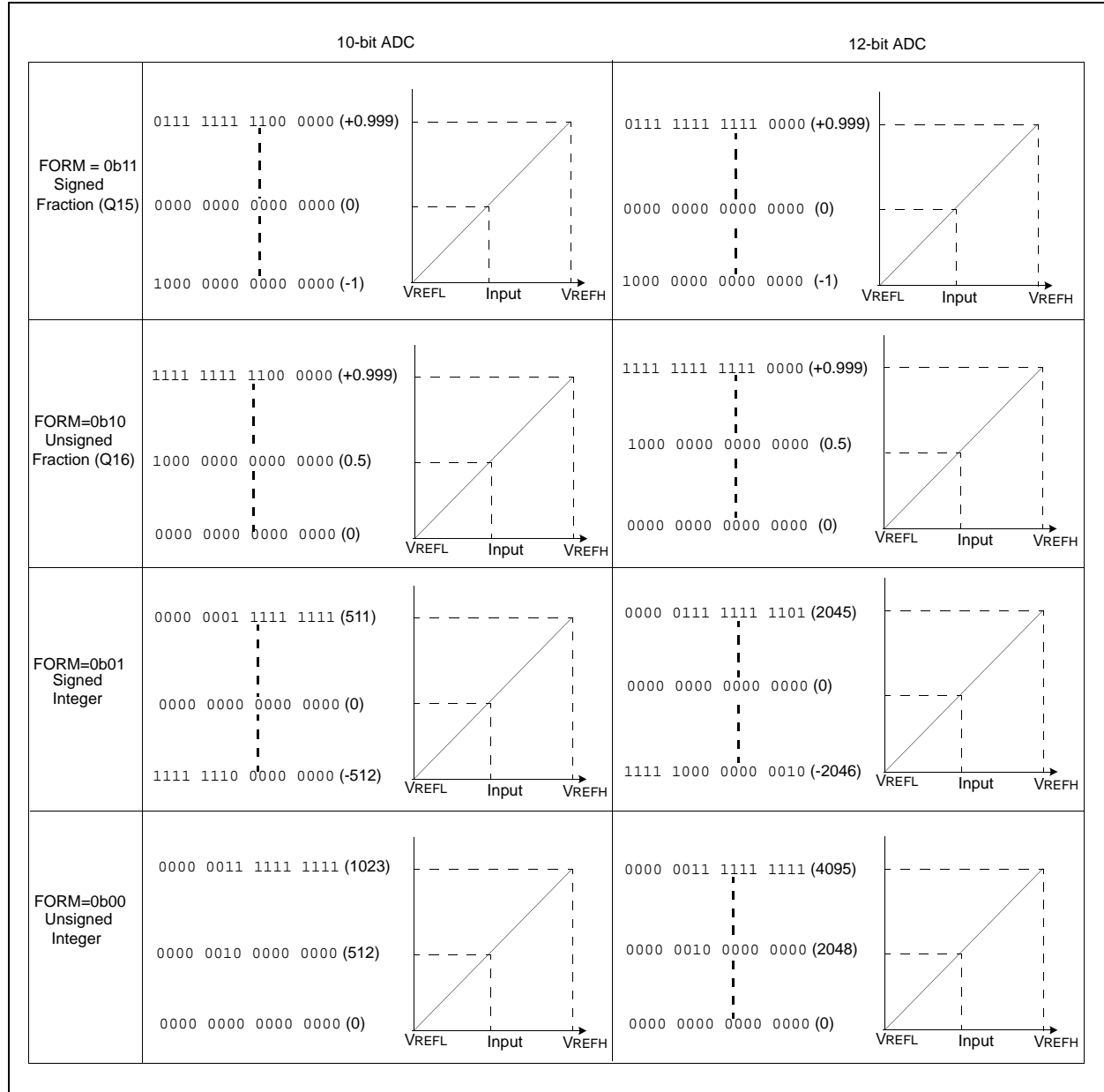
16.4.5 出力データ フォーマットの選択

ADC 結果には 4 種類の数値フォーマットを使用できます (図 16-14 参照)。出力データ フォーマットの選択には、ADC 制御レジスタ (ADxCON1<9:8>) のデータ出力フォーマット ビット (FORM<1:0>) を使用します。表 16-8 に、FORM<1:0> ビットの設定値とデータ フォーマット の関係を示します。

表 16-8: データ フォーマットの選択

FORM<1:0>	データ フォーマット
11	符号付き小数フォーマット
10	符号なし小数フォーマット
01	符号付き整数フォーマット
00	符号なし整数フォーマット

図 16-14: ADC 出力フォーマット



16.4.6 サンプリング / 変換動作 (SMPI) ビット

ADC 制御レジスタ 2 (ADxCON2<5:2>) の割り込みあたりサンプル数制御ビット (SMPI<3:0>) の機能は、DMA 付きデバイスと DMA なしデバイスで全く異なります。

DMA なしデバイスでは、SMPI<3:0> ビットは「割り込みあたりサンプル数選択ビット」と呼ばれます。DMA 付きデバイスでは、SMPI<3:0> ビットは「DMA アドレスのインクリメント頻度選択ビット」と呼ばれます。

16.4.6.1 DMA なしデバイスの SMPI ビットの設定

DMA なしデバイスでは、SMPI<3:0> ビットを設定する事によって、毎回または特定回数のサンプリング / 変換シーケンスの後で割り込みを生成する事ができます。割り込みから次の割り込みまでのサンプリング / 変換シーケンスの回数は 1 ~ 16 の範囲で選択できます。この間の変換結果の総数は、サンプリング チャンネル数 (CHPS<1:0> ビットで設定) と SMPI<3:0> ビット値の積として計算できます。各種サンプリング モードにおける SMPI 値については、16.5「ADC 割り込みの生成」を参照してください。

16.4.6.2 DMA 付きデバイスの SMPI ビットの設定

DMA 付きデバイスで複数の変換結果をバッファリングする必要がある場合、ADC モジュールと共に DMA を使用して変換結果を DMA バッファに保存する必要があります。この場合、SMPI<3:0> ビットでは DMA RAM バッファポインタをインクリメントする頻度を選択します。DMA RAM バッファポインタのインクリメント数は、DMABL<2:0> ビットが指定する 1 入力あたりの DMA RAM バッファ長以下である必要があります。SMPI<3:0> ビットの設定に関係なく、毎回の変換終了後に ADC 割り込みが生成されます。

同時または逐次サンプリング モードで 1、2、4 チャンネルモードのいずれかを有効にし、CH0 チャンネルのスキャンを無効にする場合、SMPI<3:0> ビットを「000」に設定して毎回のサンプリングで DMA アドレスポインタをインクリメントします。

交互入力選択モードを使用する同時または逐次サンプリング モードで 1、2、4 チャンネルモードのいずれかを有効にし、CH0 チャンネルのスキャンを無効にする場合、SMPI<3:0> ビットを「001」に設定して 2 回のサンプリングごとに DMA アドレスポインタをインクリメントします。

チャンネル スキャンを使用し、交互入力選択モードを無効にする場合、SMPI<3:0> ビットをスキャンする入力の数より 1 つ少なく設定する必要があります (SMPI<3:0> = N - 1)。

16.4.7 変換トリガ源

サンプリング終了 / 変換開始を他のイベントに同期させたい事がしばしばあります。ADC モジュールは下記のいずれかを変換トリガ源として使用できます。

- 外部割り込みトリガ (INT0 のみ)
- タイマ割り込みトリガ
- モータ制御用 PWM 特殊イベントトリガ (dsPIC33F モータ制御用デバイスのみ)

16.4.7.1 外部割り込みトリガ (INT0 のみ)

SSRC<2:0> = 001 の場合、A/D 変換は INT0 ピンがアクティブ状態に移移するとトリガされます。INT0 ピンは、立ち上がりエッジ入力または立ち下がりエッジ入力として設定できます。

16.4.7.2 タイマ割り込みトリガ

この ADC モジュール トリガモードを選択するには、SSRC<2:0> ビットを「010」に設定します。TMR3 (ADC1 用) と TMR5 (ADC2 用) を使用して、16 ビット タイマカウンタ レジスタ (TMRx) と 16 ビット タイマ周期レジスタ (PRx) が一致した時に A/D 変換の開始をトリガできます。32 ビットタイマを使用して A/D 変換の開始をトリガする事もできます。SSRC<2:0> = 100 の場合、タイマがスワップされます (TMR5 を ADC1 に使用し、TMR3 を ADC2 に使用)。

16.4.7.3 モータ制御用 PWM 特殊イベントトリガ (dsPIC33F モータ制御用デバイス専用)

PWM モジュールが備えるイベントトリガを使用すると、A/D 変換を PWM タイムベースに同期させることができます。SSRC<2:0> = 011 の場合、PWM 周期内の任意のユーザ設定可能タイミングで A/D サンプリング / 変換をトリガできます。特殊イベントトリガを使用すると、A/D 変換結果を収集してからデューティ サイクル値を更新するまでの遅延時間を最小限に抑えることができます。

次の変換トリガが発生する前に ADC モジュールが確実に入力のサンプリングを完了できるようにするために、アプリケーションは ASAM ビットをセットする必要があります。

16.4.8 アナログ ポートピンの設定

アナログ/デジタルピン コンフィグレーション レジスタ (ADxPCFGL) は、アナログ入力として使用するデバイスピンの入力条件を指定します。これらのレジスタは、パラレル I/O ポートモジュール内のデータ方向レジスタ (TRISx) と共に、ADC ピンの動作を制御します。

PCFGn ビット (ADxPCFGL<n>) をクリアすると、対応するピンはアナログ入力として設定されます。リセット時に ADxPCFGL レジスタはクリアされるため、ADC 入力ピンはリセット時に既定値のアナログ入力ピンとして設定されます。

ピンがアナログ入力として設定された場合、対応するポートの I/O デジタル入力バッファは無効化されます (電流を消費しません)。

ポートピンをアナログ入力として使用する場合、対応する TRIS ビットはセットされている必要があります (そのポートを入力として指定するため)。A/D 入力に関連付けた I/O ピン出力として設定 (対応する TRIS ビットをクリア) した場合、そのポートのデジタル出力レベル (VOH または VOL) が変換されます。TRIS ビットはデバイスリセット時に全てセットされます。

PCFGn ビットをセットすると、対応するピンはデジタル I/O として設定されます。この場合、アナログ マルチプレクサへの入力は AVss に接続されます。

- Note 1:** ADC ポートレジスタの読み出し時に、アナログ入力として設定された全てのピンは「0」として読み出されます。
- 2:** デジタル入力として定義されたピンへアナログ電圧レベルが加えられると、入力バッファにデバイス仕様値を超える電流消費を生じる可能性があります。

16.4.9 ADC モジュールの起動

ADON ビット (ADxCON1<15>) を「1」にセットすると、そのモジュールは動作モードとなり、電力の供給を受けて完全に機能します。

ADON ビットを「0」にクリアするとモジュールは無効化されます。この場合、消費電流を抑えるために回路のデジタル部とアナログ部の動作は停止します。

停止モードから動作モードに復帰する場合、ユーザ アプリケーションはアナログ段が安定するまで待機する必要があります。安定化時間の詳細は各デバイス データシートの「電氣的特性」を参照してください。

- Note:** ADON = 1 (ADC モジュールが動作中) の時、SSRC<2:0>、SIMSAM、ASAM、CHPS<1:0>、SMPI<3:0>、BUFM、ALTS ビットと ADCON3 および ADCSSL レジスタへの書き込みを回避する必要があります。このような書き込みは予期せぬ結果を招きます。

16.4.10 ADC モジュールの停止

ADON ビットをクリアすると ADC モジュールは無効化されます (全てのスキャン、サンプリング、変換プロセスが停止します)。この状態でも ADC モジュールは電流を消費します。PMD レジスタの ADxMD ビットをセットすると、ADC モジュールが無効化され、かつ ADC クロック源が停止するため、デバイスの消費電流を低減できます。ADxMD ビットをセットした後にクリアすると、ADC モジュール レジスタは既定値状態にリセットされる事に注意してください。さらに、ADC 入力ピンに多重化されたデジタルピンを使用している場合、それらのピンの機能はリセット時にアナログ機能に戻されます。これらのピンは、ADxMD ビットがセットされている間 (クリアされるまで) はデジタルピンとして機能します。この場合 ADxPCFG ビットは効果を持ちません。

Note: A/D 変換実行中に ADON ビットをクリアすると、実行中の変換は中止されます。不完全な変換結果は ADC バッファに書き込まれません。
--

16.5 ADC 割り込みの生成

DMA が有効な場合、SMPI<3:0> ビット (ADxCON2<5:2>) は、DMA アドレス / ポインタのインクリメント 1 回あたりのチャンネル (CH0/CH1/CH2/CH3) あたりサンプリング / 変換数を決定します。

DMA バッファに変換順モードで書き込むように ADC モジュールを設定した場合、SMPI<3:0> ビットは効果を持ちません。

DMA 転送を有効にする場合、SMPI<3:0> ビットをクリアする必要があります (チャンネル スキャンまたは交互サンプリング使用時を除く)。SMPI<3:0> の設定要件の詳細は **16.7「DMA 付きデバイスにおける変換結果のバッファリングの指定」** を参照してください。

SIMSAM ビット (ADxCON1<3>) で逐次サンプリングを指定した場合、CHPS<1:0> ビット (ADxCON2<9:8>) で選択したチャンネル数に関係なく、ADC モジュールは各変換で 1 回サンプリングしてバッファにデータを保存します。使用する DMA チャンネルに対して DMAxCNT レジスタで指定した値は、バッファ内のデータサンプル数に一致します。

DMA 付きデバイスでは毎回の変換後に割り込みが発生し、これにより DONE ビットがセットされます (DONE ビットは割り込みフラグ (ADxIF) の状態を反映するため)。

DMA なしデバイスでは、ADC モジュールは変換が完了するたびに変換結果を ADC 結果バッファに書き込みます。ADC 結果バッファは 16 ワードの配列を持ち、SFR 空間を介してアクセスされます。ユーザ アプリケーションは、AD 変換結果が生成されるたびに毎回読み出す事ができます。しかしこれは CPU 時間を消費しすぎます。一般的には、コードを簡略化するために、モジュールが一定数の結果をバッファに書き込んだ時に割り込みを生成します。ADC モジュールは 16 個の結果のバッファリングをサポートします。従って、割り込み 1 回あたりの変換回数は 16 以下に制限されます。

ADC 割り込み 1 回あたりの変換回数 (1 ~ 16) は、下記のパラメータにより決まります。

- 選択した S&H チャンネルの数
- サンプリング方式 (逐次サンプリングか同時サンプリングか)
- 割り込みあたりサンプリング / 変換数ビット (SMPI<3:0>) の設定

表 16-9 に、各種モード設定に対応する ADC 割り込み 1 回あたりの変換数を示します。

表 16-9: 各種サンプリング モードにおける割り込み 1 回あたりのサンプル数

CHPS<1:0>	SIMSAM	SMPI<3:0>	割り込みあたり 変換数	内容
00	x	N-1	N	1 チャンネルモード
01	0	N-1	N	2 チャンネル逐次サンプリングモード
1x	0	N-1	N	4 チャンネル逐次サンプリングモード
01	1	N-1	2 • N	2 チャンネル同時サンプリングモード
1x	1	N-1	4 • N	4 チャンネル同時サンプリングモード

Note 1: 2 チャンネル同時サンプリング モードでは、SMPI<3:0> ビットを 7 以下に設定する必要があります。

2: 4 チャンネル同時サンプリング モードでは、SMPI<3:0> ビットを 3 以下に設定する必要があります。

ADC 割り込み発生時に DONE ビット (ADxCON1<0>) がセットされて、サンプリング / 変換シーケンスが完了した事を示します。このビットは、次のサンプリング / 変換シーケンスの開始時にハードウェアで自動的にクリアされます。

DMA なしデバイスでは、割り込みの生成は SMPI<3:0> および CHPS ビットに基づくため、DONE ビットは毎回の変換後にセットされず、割り込みフラグ (ADxIF) がセットされた時にセットされます。

16.5.1 バッファ書き込みモード

ADC 制御レジスタ 2 (ADxCON2<1>) のバッファ書き込みモードビット (BUFM) を「1」にセットした場合、16 ワード結果バッファは 2 つの 8 ワードグループ (下位グループ: ADC1BUF0 ~ ADC1BUF7) と上位グループ: ADC1BUF8 ~ ADC1BUFF) に分割されます。ADC 割り込みが発生するたびに、変換結果は 2 つの 8 ワードバッファの一方に交互に書き込まれます。BUFM ビットをセットした場合の各バッファのサイズは 8 ワードです。従って、割り込み 1 回あたりの変換数は 8 以下に制限されます。

BUFM ビットを「0」にクリアした場合、全ての変換シーケンスで 16 ワードバッファの全体が使用されます。バッファを分割すべきかどうかの判断は、割り込み後のバッファ内容の転送に利用できる時間によって決まります。この判断はアプリケーションごとに異なります。

アプリケーションが 1 チャンネルのサンプリング/変換にかかる時間以内にフル状態のバッファを高速に読み出せる場合、BUFM ビットを「0」に設定して 1 回の割り込みあたり最大 16 回の変換を実行できます。アプリケーションは、バッファの先頭が上書きされるまでに 1 回のサンプリング/変換時間を使用できます。プロセッサがこのサンプリング/変換時間以内にバッファを読み出す事ができない場合、BUFM ビットを「1」にセットする必要があります。例えば、8 回の変換ごとに ADC 割り込みを生成する場合、プロセッサはバッファから 8 つの変換結果を転送するために次の割り込みまでの全時間を使用できます。

16.5.2 バッファ書き込み状態

BUFM 制御ビットを使用して変換結果バッファを分割した場合、BUFS ステータスビット (ADxCON2<7>) は ADC モジュールが現在どちらのバッファに書き込んでいるのかを示します。BUFS = 0 の場合、ADC モジュールは下位グループに書き込んでいます。この場合ユーザアプリケーションは上位グループから変換結果を読み出す必要があります。BUFS = 1 の場合、ADC モジュールは上位グループに書き込んでいます。この場合ユーザアプリケーションは下位グループから変換結果を読み出す必要があります。

16.6 変換するアナログ入力の選択

変換する入力を柔軟に選択するために、ADC モジュールは下記の機能を備えます。

- 固定入力選択
- 交互入力選択
- チャンネル スキャン (CH0 のみ)

16.6.1 固定入力選択

10 ビット ADC コンフィグレーションでは、最大 4 つの S&H チャンネル (CH0 ~ CH3) を使用できますが、12 ビット ADC コンフィグレーションでは 1 つの S&H チャンネル (CH0) しか使用できません。これらの S&H チャンネルは、アナログ マルチプレクサを介してアナログ入力ピンに接続されます。

ALTS = 0 の場合、アナログ入力の選択には CH0SA<4:0>、CH0NA、CH123SA、CH123NA<1:0> ビットを使用します。

表 16-10: アナログ入力の選択

		MUXA	
		制御ビット	アナログ入力
CH0	+ve	CH0SA<4:0>	AN0 ~ AN31
	-ve	CH0NA	VREF-、AN1
CH1	+ve	CH123SA	AN0、AN3
	-ve	CH123NA<1:0>	AN6、AN9、VREF-
CH2	+ve	CH123SA	AN1、AN4
	-ve	CH123NA<1:0>	AN7、AN10、VREF-
CH3	+ve	CH123SA	AN2、AN5
	-ve	CH123NA<1:0>	AN8、AN11、VREF-

Note: デバイスによって利用できる入力が異なります。

同時および逐次サンプリングモードでは、CHPS ビットと SIMSAM ビットを設定する事によって 4 チャンネルの全てを有効にできます。

DMA 付きデバイスでは、SMPI<3:0> ビットを「0000」に設定して DMA アドレスポインタを毎回のサンプリングでインクリメントします。

例 16-3 に、ADC 入力を 4 チャンネル ADC コンフィグレーション向けに設定するコードシーケンスを示します。

例 16-3: ADC 入力の設定用コードシーケンス

```
// Initialize MUXA Input Selection
AD1CHS0bits.CH0SA = 3; // Select AN3 for CH0 +ve input
AD1CHS0bits.CH0NA = 0; // Select VREF- for CH0 -ve input

AD1CHS123bits.CH123SA=0;// Select AN0 for CH1 +ve input
                        // Select AN1 for CH2+ve input
                        // Select AN2 for CH3 +ve input
AD1CHS123bits.CH124NA=0;// Select VREF- for CH1/CH2/CH3 -ve inputs
```

16.6.2 交互入力選択モード

交互入力選択モードでは、MUXA および MUXB 制御ビットが変換用チャンネルを交互に選択します。ADC はまず MUXA 入力選択を使用してスイープを行い、次に MUXB 入力選択を使用してスイープを行い、次は再び MUXA 入力選択を使用して同じ動作を繰り返します。交互入力選択モードは、ADC 制御レジスタ 2 (ADxCON2<0>) の交互サンプリング ビット (ALTS) で有効にできます。

アナログ入力マルチプレクサは、AD1CHS123 および AD1CHS0 レジスタによって制御されます。変換を行う入力ソースの選択には、2 組の制御ビット MUXA (CHySA/CHyNA) と MUXB (CHySB/CHyNB) を使用します。MUXB 制御ビットは、交互入力選択モードにのみ使用されます。

表 16-11: アナログ入力の選択

		MUXA		MUXB	
		制御ビット	アナログ入力	制御ビット	アナログ入力
CH0	+ve	CH0SA<4:0>	AN0 ~ AN31	CH0SB<4:0>	AN0 ~ AN31
	-ve	CH0NA	VREF-, AN1	CH0NB	VREF-, AN1
CH1	+ve	CH123SA	AN0、AN3	CH123SB	AN0、AN3
	-ve	CH123NA<1:0>	AN6、AN9、VREF-	CH123NB<1:0>	AN6、AN9、VREF-
CH2	+ve	CH123SA	AN1、AN4	CH123SB	AN1、AN4
	-ve	CH123NA<1:0>	AN7、AN10、VREF-	CH123NB<1:0>	AN7、AN10、VREF-
CH3	+ve	CH123SA	AN2、AN5	CH123SB	AN2、AN5
	-ve	CH123NA<1:0>	AN8、AN11、VREF-	CH123NB<1:0>	AN8、AN11、VREF-

Note: デバイスによって利用できる入力異なります。

DMA なしデバイスで交互入力選択モードを使用する場合、偶数回のサンプリング / 変換シーケンスごとに ADC 割り込みが発生するように割り込みあたりサンプリング / 変換数ビット (SMPI<3:0>) を設定する必要があります。表 16-12 に、各種 ADC コンフィグレーションでの交互入力選択モードに使用できる SMPI 値を示します。

表 16-12: 交互入力選択モードに使用できる SMPI 値

CHPS<1:0>	SIMSAM	SMPI<3:0> (10 進数)	割り込みあたり 変換回数	内容
00	x	1,3,5,7,9,11,13,15	2,4,6,8,10,12,14,16	1 チャンネルモード
01	0	3,7,11,15	4,8,12,16	2 チャンネル逐次 サンプリングモード
1x	0	7,15	8,16	4 チャンネル逐次 サンプリングモード
01	1	1,3,5,7	4,8,12,16	2 チャンネル同時 サンプリングモード
1x	1	1,3	8,16	4 チャンネル同時 サンプリングモード

例 16-4 に、DMA なしデバイスの 4 チャンネル同時サンプリング コンフィグレーションで ADC モジュールを交互入力選択モードに設定するコードシーケンスを示します。図 16-15 に、ADC モジュールの動作シーケンスを示します。

Note: ADC 割り込みが発生すると、ADC 内部ロジックが初期化されて変換シーケンスが最初から再開されます。

例 16-4: ADC を 4 チャンネル同時サンプリングで交互入力選択モードに設定するコードシーケンス (DMA なしデバイス)

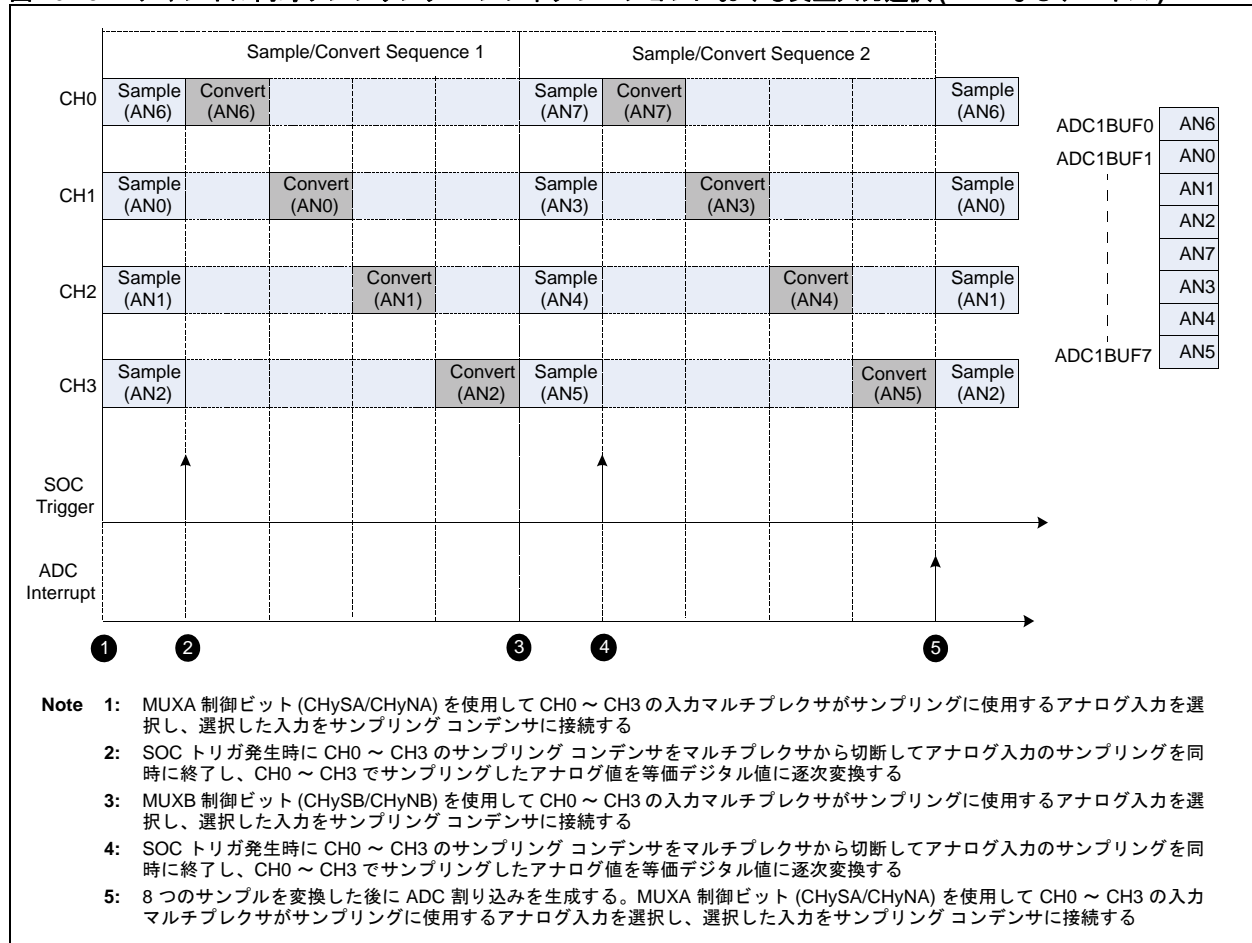
```
AD1CON1bits.AD12B = 0; // Select 10-bit mode
AD1CON2bits.CHPS = 3; // Select 4-channel mode
AD1CON1bits.SIMSAM = 1; // Enable Simultaneous Sampling
AD1CON2bits.ALTS = 1; // Enable Alternate Input Selection
AD1CON2bits.SMPI = 1; // Select 8 conversion between interrupt
AD1CON1bits.ASAM = 1; // Enable Automatic Sampling
AD1CON1bits.SSRC = 2; // Timer3 generates SOC trigger

// Initialize MUXA Input Selection
AD1CHS0bits.CH0SA = 6; // Select AN6 for CH0 +ve input
AD1CHS0bits.CH0NA = 0; // Select VREF- for CH0 -ve input
AD1CHS123bits.CH123SA = 0; // Select CH1 +ve = AN0, CH2 +ve = AN1, CH3 +ve = AN2
AD1CHS123bits.CH123NA = 0; // Select VREF- for CH1/CH2/CH3 -ve inputs

// Initialize MUXB Input Selection
AD1CHS0bits.CH0SB = 7; // Select AN7 for CH0 +ve input
AD1CHS0bits.CH0NB = 0; // Select VREF- for CH0 -ve input

AD1CHS123bits.CH123SB = 1; // Select CH1 +ve = AN3, CH2 +ve = AN4, CH3 +ve = AN5
```

図 16-15: 4 チャンネル同時サンプリング コンフィグレーションにおける交互入力選択 (DMA なしデバイス)



例 16-5 に、DMA なしデバイスの 2 チャンネル逐次サンプリングコンフィグレーションで ADC モジュールを交互入力選択モードに設定するコードシーケンスをします。

例 16-5: ADC を 2 チャンネル逐次サンプリングで交互入力選択モードに設定するコードシーケンス (DMA なしデバイス)

```
AD1CON1bits.AD12B=0; // Select 10-bit mode
AD1CON2bits.CHPS=1; // Select 2-channel mode
AD1CON2bits.SMPI = 3; // Select 4 conversion between interrupt
AD1CON1bits.ASAM = 1; // Enable Automatic Sampling
AD1CON2bits.ALTS = 1; // Enable Alternate Input Selection
AD1CON1bits.SIMSAM = 0; // Enable Sequential Sampling
AD1CON1bits.SSRC = 2; // Timer3 generates SOC trigger

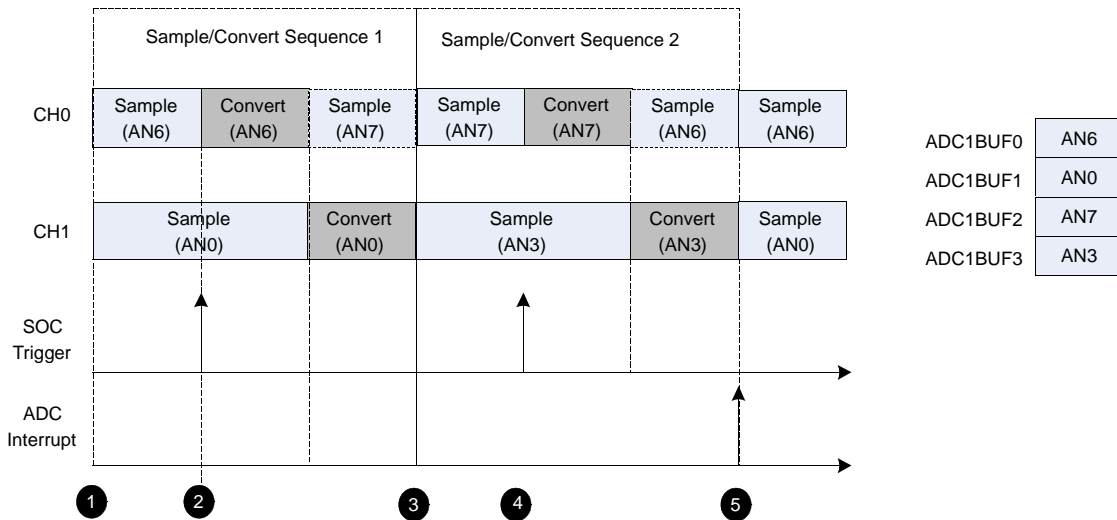
// Initialize MUXA Input Selection
AD1CHS0bits.CH0SA = 6; // Select AN6 for CH0 +ve input
AD1CHS0bits.CH0NA = 0; // Select VREF- for CH0 -ve input

AD1CHS123bits.CH123SA=0; // Select AN0 for CH1 +ve input
AD1CHS123bits.CH123NA=0; // Select Vref- for CH1 -ve inputs

// Initialize MUXB Input Selection
AD1CHS0bits.CH0SB = 7; // Select AN7 for CH0 +ve input
AD1CHS0bits.CH0NB = 0; // Select VREF- for CH0 -ve input

AD1CHS123bits.CH123SB=1; // Select AN3 for CH1 +ve input
AD1CHS123bits.CH124NB=0; // Select VREF- for CH1 -ve inputs
```

図 16-16: 2 チャンネル逐次サンプリングコンフィグレーションにおける交互入力選択 (DMA なしデバイス)



- Note**
- 1: MUXA 制御ビット (CHySA/CHyNA) を使用して CH0 と CH1 の入力マルチプレクサがサンプリングに使用するアナログ入力を選択し、選択した入力をサンプリング コンデンサに接続する
 - 2: SOC トリガ発生時に CH0/CH1 入力を逐次サンプリングして等価デジタル値に変換する
 - 3: MUXB 制御ビット (CHySB/CHyNB) を使用して CH0 と CH1 の入力マルチプレクサがサンプリングに使用するアナログ入力を選択し、選択した入力をサンプリング コンデンサに接続する
 - 4: SOC トリガ発生時に CH0/CH1 入力を逐次サンプリングして等価デジタル値に変換する
 - 5: 4 つのサンプルを変換した後、ADC 割り込みを生成する。MUXA 制御ビット (CHySA/CHyNA) を使用して CH0 と CH1 の入力マルチプレクサがサンプリングに使用するアナログ入力を選択し、選択した入力をサンプリング コンデンサに接続する

DMA 付きデバイスで交互入力選択モードを有効にする場合、SMPI<3:0> を「001」に設定する事によって 2 回のサンプリングごとに DMA アドレスポインタをインクリメントします。

図 16-17: 4 チャンネル同時サンプリングにおける交互入力選択 (DMA 付きデバイス)

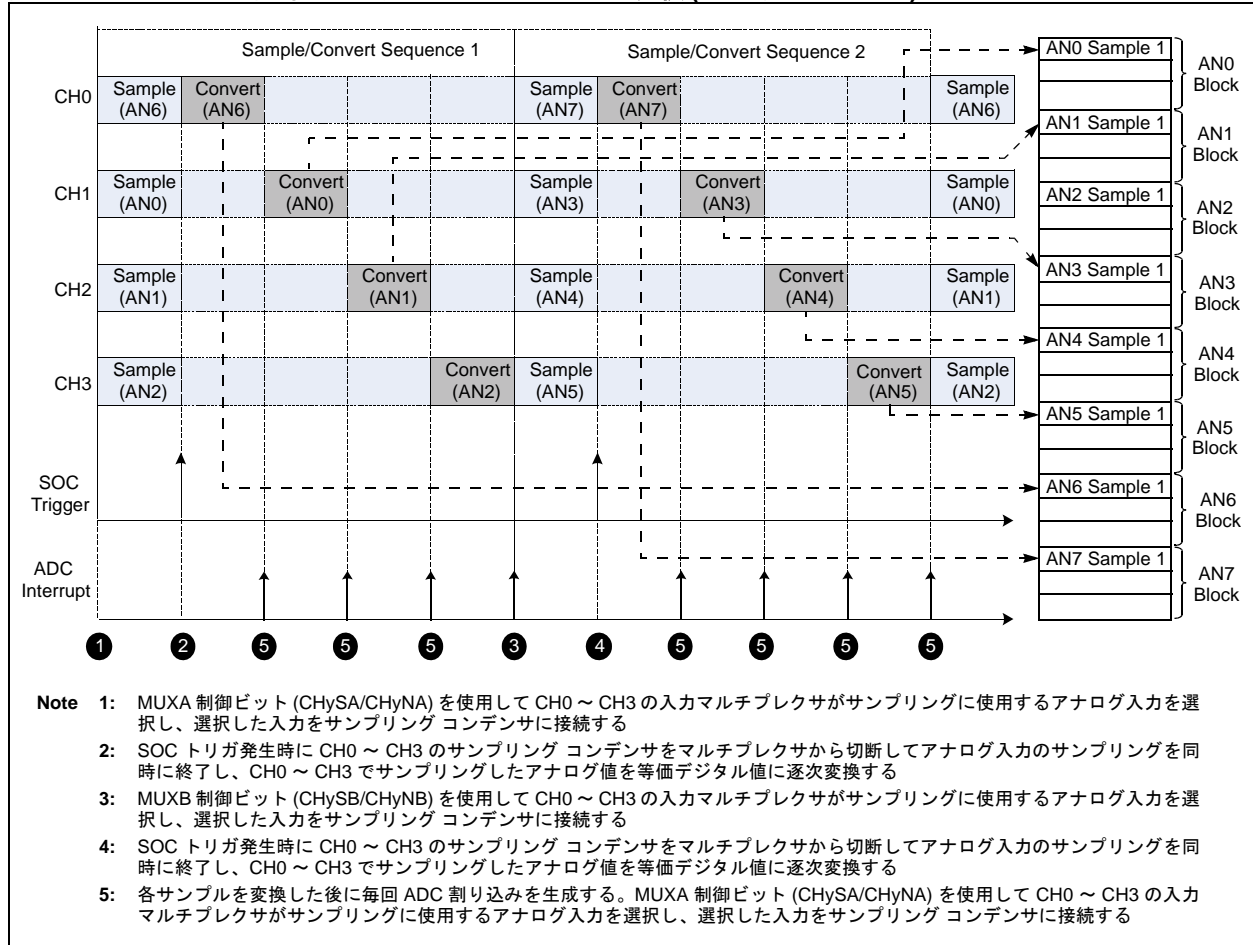
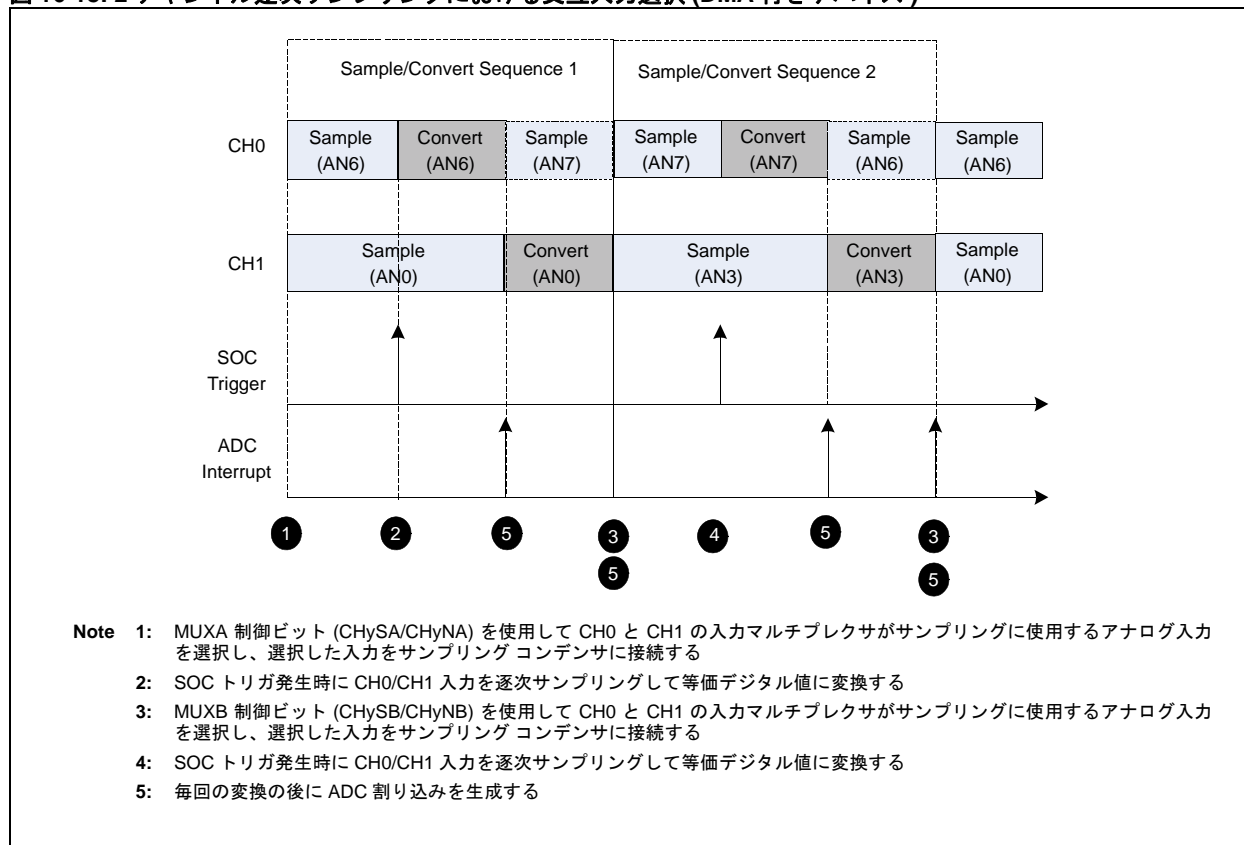


図 16-18: 2 チャンネル逐次サンプリングにおける交互入力選択 (DMA 付きデバイス)



16.6.3 チャンネル スキャン

ADC モジュールは、CH0 (S&H チャンネル「0」) を使用するチャンネル スキャンモードをサポートします。スキャンする入力数はソフトウェアで選択できます。アナログ入力 AN0 ~ AN31 (DMA なしデバイスの場合 AN0 ~ AN12) から任意のサブセットを変換用に選択できます。選択した入力は昇順に変換されます。例えばAN4、AN1、AN3を選択した場合、AN1-AN3-AN4の順に変換されます。チャンネル選択レジスタ (AD1CSSL) を設定する事により、変換シーケンスに含めるアナログ入力を選択できます。チャンネル選択レジスタのビットを「1」にセットすると、そのビットに対応するアナログ入力チャンネルは変換シーケンスに含まれます。ADC 制御レジスタ 2 (ADxCON2<10>) のチャンネル スキャンビット (OSCNA) をセットすると、チャンネルスキャン モードが有効になります。チャンネル スキャンモードでは、MUXA ソフトウェア制御は無視され、ADC モジュールは有効化されたチャンネルを使用してシーケンスを実行します。

DMA なしデバイスでは、毎回のサンプリング / 変換シーケンスで 1 つのアナログ入力がスキャンされます。選択された全てのチャンネルをスキャンした後に、ADC 割り込みを生成する必要があります。N 個の入力をチャンネル スキャン用に有効にした場合、N 回のサンプリング / 変換シーケンスごとに割り込みを生成する必要があります。表 16-13 に、各種 ADC コンフィグレーションで CH0 チャンネルを使用して N 個のアナログ入力をスキャンする場合の SMPI 値を示します。

Note: 最大で 16 個の ADC 入力を 1 度にスキャンするように設定できます。

表 16-13: チャンネル スキャンモードにおける割り込み 1 回あたりの変換回数
(DMA なしデバイス)

CHPS<1:0> >	SIMSAM	SMPI<3:0> (10 進数)	割り込みあたり 変換回数	内容
00	x	N-1	N	1 チャンネルモード
01	0	2N-1	2N	2 チャンネル逐次サンプリングモード
1x	0	4N-1	4N	4 チャンネル逐次サンプリングモード
01	1	N-1	2N	2 チャンネル同時サンプリングモード
1x	1	N-1	4N	4 チャンネル同時サンプリングモード

例 16-6 に、DMA なしデバイスで CH0 を使用して 4 つのアナログ入力をスキャンするコードシーケンスを示します。図 16-19 に、ADC モジュールの動作シーケンスを示します。

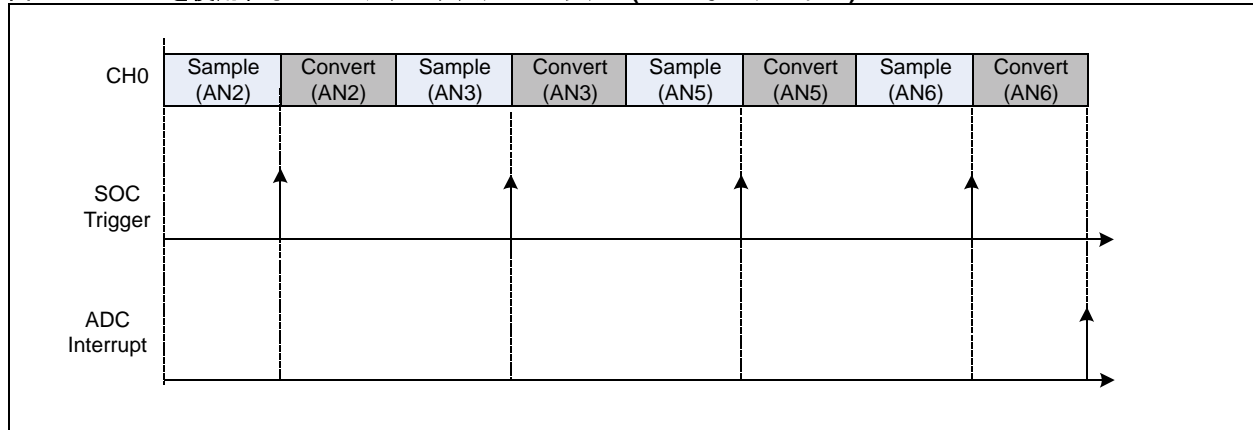
Note: ADC 割り込み発生時に ADC ロジックが初期化されて変換シーケンスが最初から再開されます。

例 16-6: CH0 を使用して 4 つのアナログ入力をスキャンするコードシーケンス
(DMA なしデバイス、10 ビット/12 ビット ADC)

```
AD1CON1bits.AD12B=1;    // Select 12-bit mode, 1-channel mode
AD1CON2bits.SMPI = 3;    // Select 4 conversions between interrupt
AD1CHS0bits.ASAM = 1;    // Enable Automatic Sampling
AD1CON2bits.CSCNA = 1;   // Enable Channel Scanning

// Initialize Channel Scan Selection
AD1CSSLbits.CSS2=1;      // Enable AN2 for scan
AD1CSSLbits.CSS3=1;      // Enable AN3 for scan
AD1CSSLbits.CSS5=1;      // Enable AN5 for scan
AD1CSSLbits.CSS6=1;      // Enable AN6 for scan
```

図 16-19: CH0 を使用する 4 つのアナログ入力のスキャン (DMA なしデバイス)



例 16-7 に、DMA なしデバイスの 2 チャンネル交互入力選択コンフィグレーションで CH0 を使用して 2 つのアナログ入力をスキャンするコードシーケンスを示します。図 16-20 に、ADC モジュールの動作シーケンスを示します。

例 16-7: 交互入力選択によるチャンネル スキャンのコードシーケンス (DMA なしデバイス)

```
AD1CON1bits.AD12B = 0; // Select 10-bit mode
AD1CON2bits.CHPS = 1; // Select 2-channel mode
AD1CON1bits.SIMSAM = 0; // Enable Sequential Sampling
AD1CON2bits.ALTS = 1; // Enable Alternate Input Selection
AD1CON2bits.CSCNA = 1; // Enable Channel Scanning
AD1CON2bits.SMPI = 7; // Select 8 conversion between interrupt
AD1CON1bits.ASAM = 1; // Enable Automatic Sampling

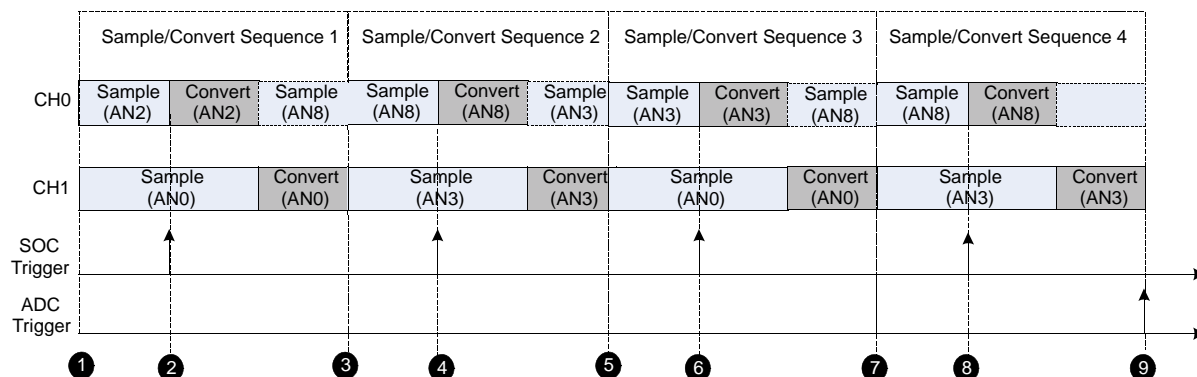
// Initialize Channel Scan Selection
AD1CSSLbits.CSS2 = 1; // Enable AN2 for scan
AD1CSSLbits.CSS3 = 1; // Enable AN3 for scan

// Initialize MUXA Input Selection
AD1CHS123bits.CH123SA = 0; // Select AN0 for CH1 +ve input
AD1CHS123bits.CH123NA = 0; // Select Vref- for CH1 -ve inputs

// Initialize MUXB Input Selection
AD1CHS0bits.CH0SB = 8; // Select AN8 for CH0 +ve input
AD1CHS0bits.CH0NB = 0; // Select VREF- for CH0 -ve inputs

AD1CHS123bits.CH123SB = 0; // Select AN4 for CH1 +ve input
AD1CHS123bits.CH124NB = 0; // Select VREF- for CH1 -ve inputs
```

図 16-20: 交互入力選択によるチャンネル スキャン (DMA なしデバイス)



- Note**
- 1: CH0 の入力マルチプレクサはチャンネル スキャンロジックが内部生成する制御ビットを使用し、CH1 の入力マルチプレクサは MUXA 制御ビット (CHySA/CHyNA) を使用してサンプリングに使用するアナログ入力を選択し、選択したアナログ入力をサンプリング コンデンサに接続する
 - 2: SOC トリガ発生時に CH0/CH1 入力を逐次サンプリングして等価デジタル値に変換する
 - 3: CH0 と CH1 の入力マルチプレクサは MUXB 制御ビット (CHySB/CHyNB) を使用してサンプリングに使用するアナログ入力を選択し、選択した入力をサンプリング コンデンサに接続する
 - 4: SOC トリガ発生時に CH0/CH1 入力を逐次サンプリングして等価デジタル値に変換する
 - 5: CH0 の入力マルチプレクサはチャンネル スキャンロジックが内部生成する制御ビットを使用し、CH1 の入力マルチプレクサは MUXA 制御ビット (CHySA/CHyNA) を使用してサンプリングに使用するアナログ入力を選択し、選択したアナログ入力をサンプリング コンデンサに接続する
 - 6: SOC トリガ発生時に CH0/CH1 入力を逐次サンプリングして等価デジタル値に変換する
 - 7: CH0 と CH1 の入力マルチプレクサは MUXB 制御ビット (CHySB/CHyNB) を使用してサンプリングに使用するアナログ入力を選択し、選択した入力をサンプリング コンデンサに接続する
 - 8: SOC トリガ発生時に CH0/CH1 入力を逐次サンプリングして等価デジタル値に変換する
 - 9: 8 つのサンプルを変換した後に ADC 割り込みを生成する

DMA 付きデバイスで CH0 だけを有効 (ALTS = 0) にしてチャンネル スキャンを使用する場合、SMPI<3:0> ビットをスキャンする入力数 (N) より 1 つ少ない値に設定する必要があります (SMPI<3:0> = N - 1)。

図 16-21: CH0 を使用する 4 つのアナログ入力のスキャン (DMA 付きデバイス)

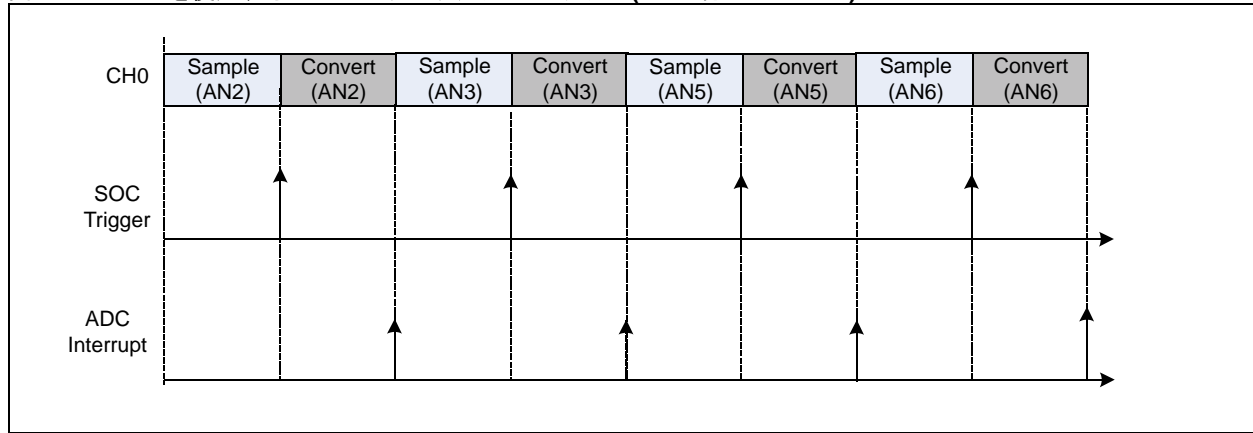
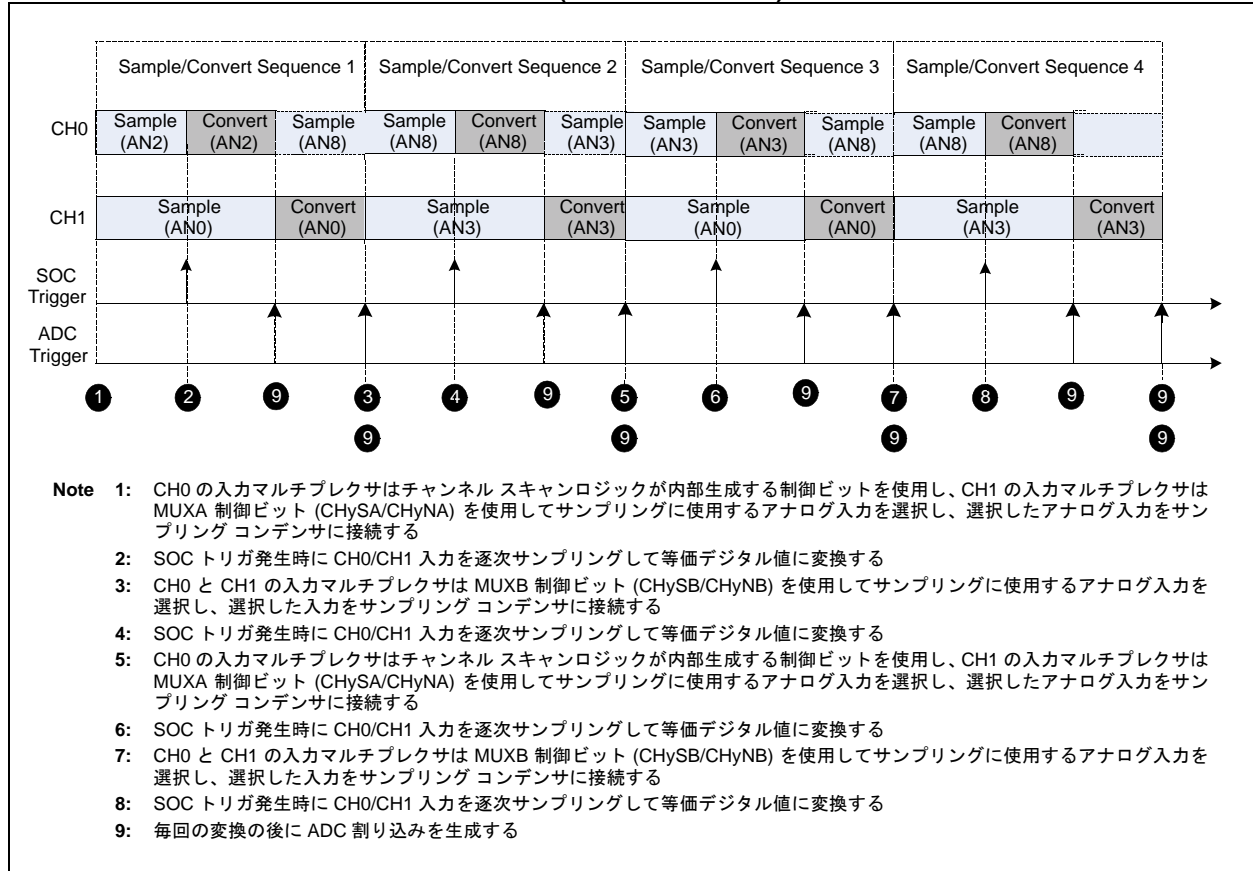


図 16-22: 交互入力選択によるチャンネル スキャン (DMA 付きデバイス)



16.7 DMA 付きデバイスにおける変換結果のバッファリングの指定

ADC モジュールは、A/D 変換結果の保存用に 1 ワード読み出し専用デュアルポート レジスタ (ADCxBUF0) を備えます。割り込み生成までに複数の変換結果をバッファリングする必要がある場合、DMA データ転送を使用できます。両方の ADC チャンネル (ADC1 と ADC2) が DMA データ転送をトリガできます。どちらの ADC チャンネルを DMA IRQ 要因として選択したかに応じて、割り込みモジュールの割り込みフラグステータス レジスタ (IFS0 または IFS1) 内の A/D 変換完了割り込みフラグ ステータスビット (AD1IF または AD2IF) がサンプリング / 変換シーケンスの結果としてセットされた時に DMA 転送が発生します。

各 A/D 変換の結果は ADCxBUF0 レジスタに保存されます。ADC モジュールに対して DMA チャンネルを有効にしない場合、ユーザ アプリケーションは結果バッファが次の変換結果によって上書きされる前に毎回の結果を読み出す必要があります。DMA を有効にすると、ADCxBUF0 レジスタから DMA RAM 空間内のユーザ定義バッファへ複数の変換結果を自動的に転送できます。従って、アプリケーションは最小限のソフトウェア オーバーヘッドで複数の変換結果を処理できます。

Note: ADC バッファからのデータ転送用に DMA チャンネルを設定する方法と、アプリケーションがデータにアクセスできるように書き込み先 DMA バッファ領域を定義する方法については、**セクション 22. 「ダイレクト メモリアクセス (DMA)」** (DS70182) を参照してください。割り込みレジスタの詳細は**セクション 6. 「割り込み」** (DS70184) を参照してください。

ADCx 制御レジスタ 1 (ADxCON1<12>) の DMA バッファ ビルドモード ビット (ADDMABM) では、ADC に使用する DMA RAM バッファ領域に変換結果を書き込む方法を設定します。このビットをセット (ADDMABM = 1) すると、DMA バッファは変換順に書き込まれます。ADC モジュールは、非 DMA スタンドアロン バッファに使用するアドレスと同じアドレスを DMA チャンネルに割り当てます。ADDMABM ビットをクリアすると、DMA バッファは Scatter/Gather モードで書き込まれます。ADC モジュールは、アナログ入力の番号と DMA バッファのサイズに基づいて、DMA チャンネルに Scatter/Gather アドレスを割り当てます。

SIMSAM ビットで同時サンプリングを指定した場合、バッファ内のデータサンプル数は CHPS<1:0> ビットの影響を受けます。バッファ内のデータサンプルのエントリ数は、サンプリングあたりのチャンネル数 (CH/S) とサンプリング回数の積として計算できます。オーバーランによるバッファ内データの喪失を防ぐために、DMAxCNT レジスタを適切なバッファサイズに設定する必要があります。

ADC モジュールで複数チャンネルを同時サンプリングし、かつ CH0 がチャンネル スキャンモードである場合、ADC モジュールが CH0 で先頭の ADC 入力から変換を再開するまでの変換数は 16 に限定されます。このモードで ADC を動作させる場合、バッファ オーバーランによるデータの喪失を避けるために、DMAxCNT レジスタを 15 に設定する必要があります。

SMPI<3:0> ビットの設定によって ADC 割り込みを無効化する事はできません。割り込みを無効化するには、ADxIE アナログ モジュール割り込みイネーブルビットをクリアする必要があります。

16.7.1 Scatter/Gather モードでの DMA の使用

ADDMABM ビットが「0」の時に Scatter/Gather モードが有効になります。このモードでは、DMA チャンネルを周辺モジュール間接アドレッシング用に設定する必要があります。DMA バッファは、AN0 ~ AN31 の中で利用可能な全てのアナログ入力に対応する連続した複数のメモリブロックに分割されます。ADC モジュールは、各アナログ入力の各変換結果を、ユーザが定義した DMA バッファ領域内の対応するブロックへ自動的に転送します。同一アナログ入力の一連のサンプリング結果は、その入力に割り当てられたブロックに逐次保存されます。

DMABL<2:0> ビット (ADxCON4<2:0>) では、各アナログ入力に割り当てた DMA バッファに格納するサンプル数を指定します。

ADC モジュールは内部ポインタを使用して各ブロック内のバッファ位置にアクセスします。このポインタは ADC モジュールを有効化した時に「0」にリセットされます。この内部ポインタの値は、DMABL<2:0> ビットで定義した値に達すると「0」にリセットされます。これにより、あるアナログ入力の変換結果が別のアナログ入力の変換結果を上書きしてしまう事態が避けられます。SMPI<3:0> ビットは、内部ポインタをインクリメントする頻度 (何回のサンプリング / 変換ごとにインクリメントするか) を指定します。

チャンネル スキャンまたは交互サンプリングが不要な場合、SMPI<3:0> ビットをクリアしてチャンネルごとの毎回のサンプリングでポインタをインクリメントするように設定する必要があります。従って、理論的には、サンプリングを行うアナログ入力に割り当てたブロックを DMA バッファ内のどの位置に配置してもかまいません。

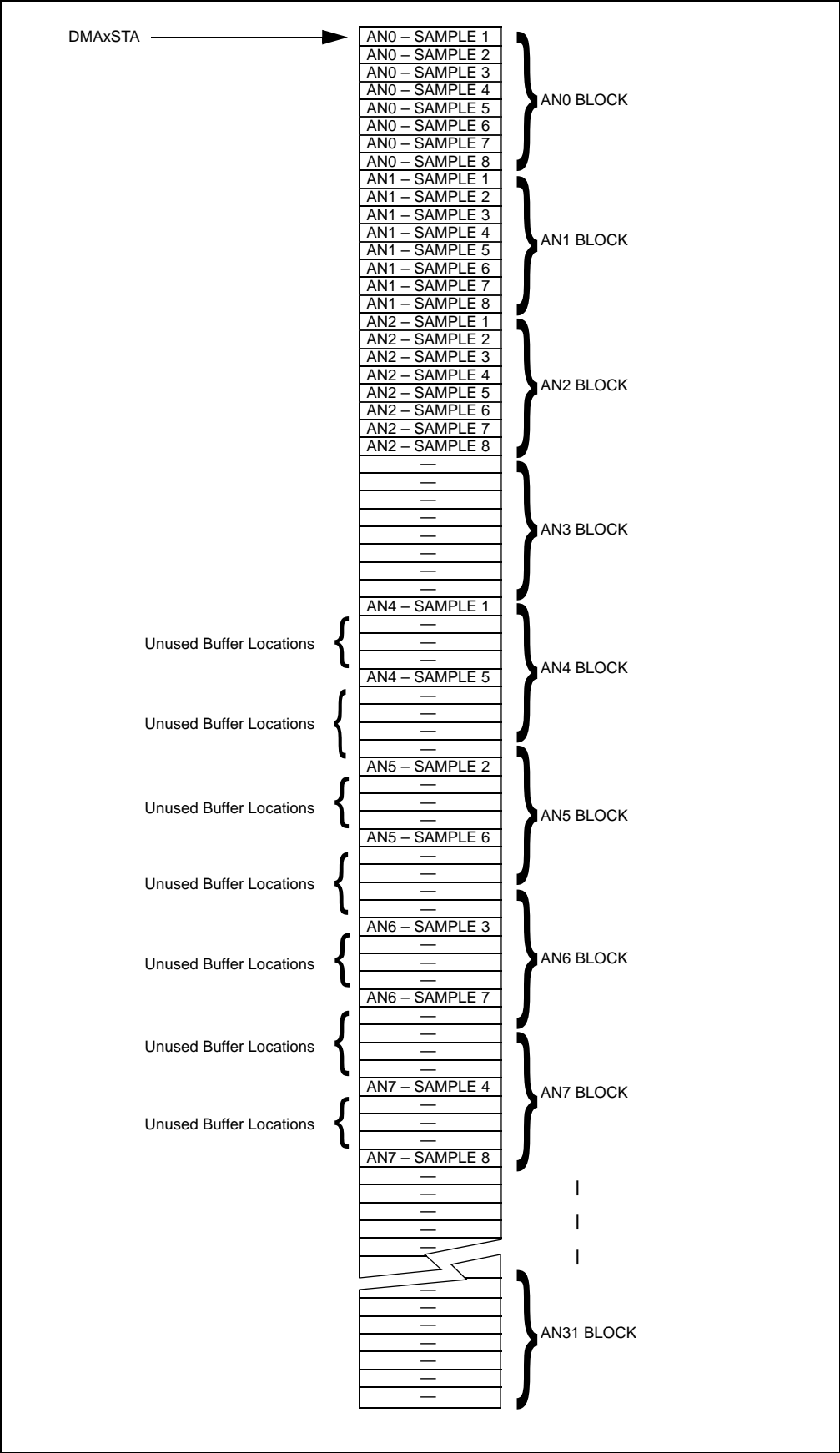
図 16-23 に示す例では、AN0、AN1、AN2 入力の変換結果を保存する各メモリブロックを、ブロック間に未使用領域を置かず連続して配置しています。しかし CH0 がスキャンする 4 つのアナログ入力 (AN4、AN5、AN6、AN7) では、AN5 ブロックの先頭、AN6 ブロックの先頭から 2 つ、AN7 ブロックの先頭から 3 つのバッファ位置が未使用となるため、DMA バッファ内のデータ配置は効率的ではありません。

スキャンを使用する場合、同時サンプリングを実行しない (SIMSAM = 0) ため、SMPI<3:0> にはスキャンする入力の数より 1 つ少ない値を設定する必要があります。例えば CHPS<1:0> = 00 (1 つの S&H チャンネルだけを使用する) かつ AD1CSSL = 0xFFFF (AN0 ~ AN15 の全てをスキャンする) の場合、SMPI<3:0> を「1111」(16 回のサンプリング / 変換シーケンスごとに内部ポインタをインクリメントする) に設定します。これにより、アナログ入力に割り当てたブロックの未使用領域がスキャンされる事を回避できます。

同様に、ALTS = 1 (アナログ入力の交互選択を使用する) の場合、SMPI<3:0> を「0001」(2 回のサンプリング / 変換ごとに内部ポインタをインクリメントする) に設定します。

Note: ADC モジュールは、生成されたバッファアドレスのリミットチェックを実行しません。従ってユーザは、例えば、使用する DMAxSTA または DMAxSTB レジスタの最下位ビット (LSb) が「0」である事を確認する必要があります。また、使用する可能性のあるアナログ入力の数と DMABL<2:0> で指定するバッファサイズの積が DMA バッファの全長を超えない事も必要です。

図 16-23: Scatter/Gather モードにおける DMA バッファ



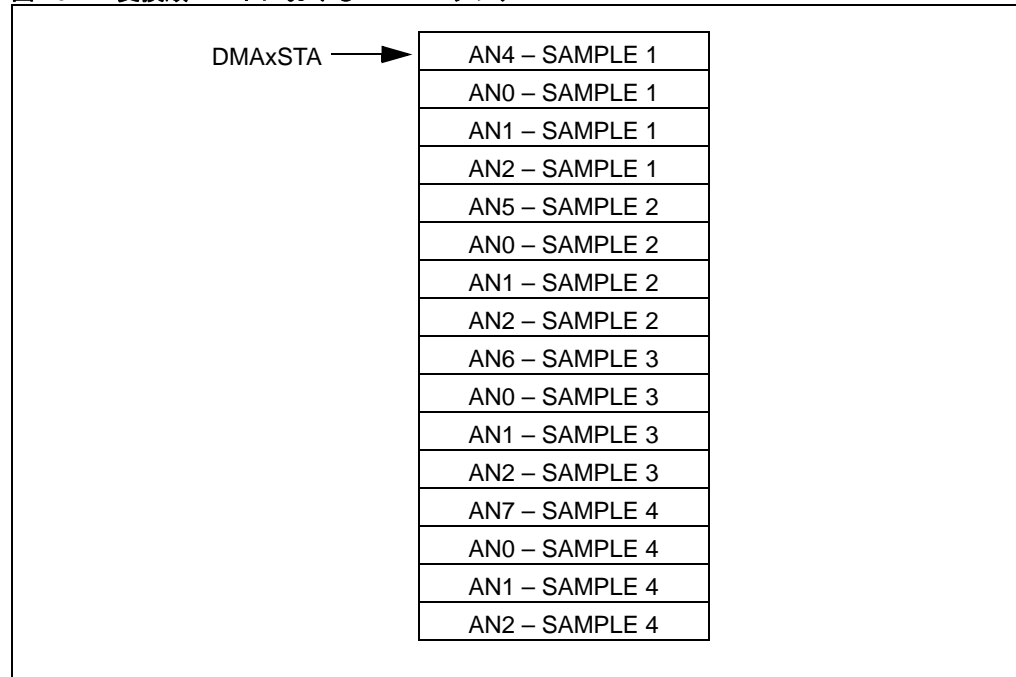
16.7.2 変換順モードでの DMA の使用

ADDMABM ビット (ADCON1<12>) = 1 の時に変換順モードが有効になります。このモードでは、DMA チャンネルをレジスタ間接または周辺モジュール間接アドレッシング向けに設定できます。全ての変換結果は、ADC モジュールによる変換実行順にユーザが指定した DMA バッファに保存されます。このモードでは、バッファはアナログ入力別のブロックに分割されません。複数入力からの変換結果は、指定されたバッファ書き込みモードに従って配置されます。

このコンフィグレーションでは、バッファポインタを常に 1 ワードごとにインクリメントする必要があります。従って SMPI<3:0> ビット (ADxCON2<5:2>) をクリアする必要があります。また、DMABL<2:0> ビット (ADxCON4<2:0>) は無視されます。

図 16-24 に、図 16-23 と同じコンフィグレーションで変換順モードを使用する場合の例を示します。この例では、DMAxCNT レジスタの設定により、16 個の変換結果が得られた後に DMA 割り込みが発生します。

図 16-24: 変換順モードにおける DMA バッファ



16.8 ADC のコンフィグレーション例

A/D 変換を実行するには下記の手順に従う必要があります。

1. 10 ビットまたは 12 ビットモードを選択する (ADxCON1<10>)
2. アナログ入力の予測電圧レンジに適合する電圧リファレンス源を選択する (ADxCON2<15:13>)
3. 必要なデータ収集レートに適合するアナログ変換クロック (プロセッサクロックを何分周するか) を選択する (ADxCON3<7:0>).
4. ポートピンをアナログ入力として選択する (ADxPCFGH<15:0> と ADxPCFGL<15:0>).
5. S&Hチャンネルへの入力の割り当てを指定する (ADxCHS0<15:0>とADxCHS123<15:0>)
6. 使用するS&Hチャンネルの数を指定する (ADxCON2<9:8>、ADxPCFGH<15:0>、ADxPCFGL<15:0>)
7. サンプリング方法を指定する (ADxCON1<3>、ADxCSSH<15:0>、ADxCSSL<15:0>)
8. 手動サンプリングまたは自動サンプリングを選択する
9. 変換トリガとサンプリング時間を選択する
10. 変換結果をバッファに保存する方法を選択する (ADxCON1<9:8>)
11. 割り込み頻度または DMA バッファポインタのインクリメント頻度を選択する (ADxCON2<9:5>)
12. DMA バッファに保存する各 ADC モジュール入力のサンプル数を選択する (ADxCON4<2:0>)
13. データ フォーマットを選択する
14. ADC 割り込みを設定する (任意)
 - ADxIF ビットをクリアする
 - 割り込み優先度を選択する (ADxIP<2:0>)
 - ADxIE ビットをセットする
15. DMA チャンネルを設定する (任意)
16. ADC モジュールを有効にする (ADxCON1<15>)

これらの設定手順における各種オプションについて以下の各セクションで説明します。

16.9 1.1 Msps 用 ADC コンフィグレーション

デバイスが 40 MIPS で動作する場合、スループット 1.1 Msps/ 分解能 10 ビットでサンプリングするように ADC モジュールを設定できます。

AD12B ビット (ADxCON1<10>) を「0」にクリアする事により、ADC モジュールを 10 ビット動作に設定します。ASAM ビット (ADxCON1<3>) を「1」にセットする事により、変換終了後にサンプリングを自動的に開始します。SSRC<2:0> ビット (ADxCON1<7:5>) を「111」に設定する事により、サンプリング終了 / 変換開始をトリガするサンプルクロック源として内部カウンタを選択します。ADRC ビット (ADxCON3<15>) を「0」にクリアする事により、ADC 変換クロックとしてシステムクロックを選択します。自動サンプリング時間ビットを 12 TAD よりも低く設定します。ADCS<7:0> ビット (ADxCON3<7:0>) を「00000011」に設定する事により、ADC 変換クロックを 75 ns に設定します (式 16-7 参照)。

式 16-7: 40 MIPS 動作時の ADC 変換クロック

$$T_{AD} = T_{CY} \times (ADCS<7:0> + 1) = (1/40M) \times 3 = 75 \text{ ns (13.3 MHz)}$$

最大 16 MIPS のデバイスでも、CPU が 13.3 MIPS で動作すれば 1.1 Msps の ADC 速度を達成できます。式 16-8 に従って ADC 変換クロックを 75 ns に設定します。

式 16-8: 13.3 MIPS 動作時の ADC 変換クロック

$$T_{AD} = T_{CY} \times (ADCS<7:0> + 1) = (1/13.3M) \times 3 = 75 \text{ ns (13.3 MHz)}$$

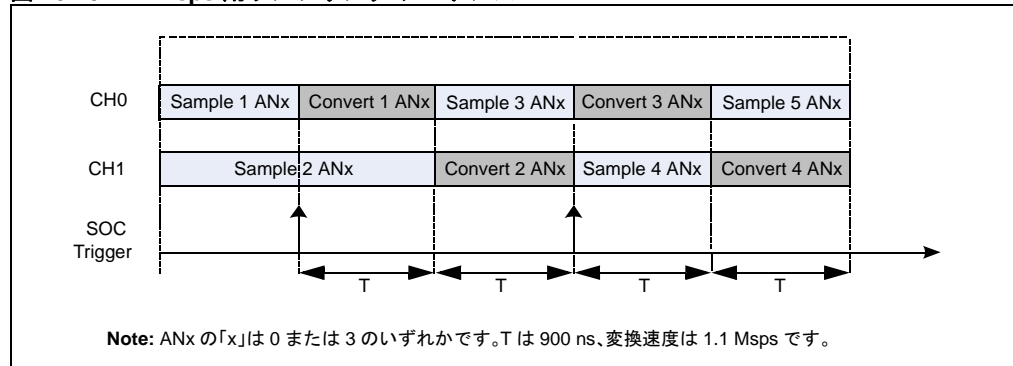
ADC モジュールを 10 ビット動作に設定するため、ADC の変換時間は式 16-9 から 12 TAD となります。

式 16-9: ADC 変換時間

$$T_{CONV} = 12 \times T_{AD} = 900 \text{ ns (1.1 MHz)}$$

ADC チャンネル CH0 と CH1 を選択 (CHPS<1:0> = 01) し、アナログ入力 AN0 または AN3 (常にいずれか1つだけ) を逐次モード (SIMSAM = 0) で変換するように設定します。図 16-25 に、ADC モジュールの動作シーケンスを示します。

図 16-25: 1.1 Msps 用サンプリング シーケンス



DMA 付きデバイスでは、DMA チャンネルをピンポンモードに設定して ADC から DMA RAM へ変換結果を転送できます。ADC と DMA のコンフィグレーションコードを例 16-8 に示します。

DMA なしデバイスでも ADC のコンフィグレーションは同じです。サンプルは 1.1 Msps の速度で ADC1BUF0 ~ ADC1BUFF に転送されます。BUFS ビットをセットする事により、2 つに分割したバッファに交互にアクセスしながらデータを処理できます。

例 16-8: 1.1 Msps 用 ADC コンフィグレーション コード

```
void initAdcl(void)
{
    AD1CON1bits.FORM = 3; // Data Output Format:Signed Fraction (Q15 format)
    AD1CON1bits.SSRC = 7; // Internal Counter (SAMC) ends sampling and starts conversion
    AD1CON1bits.ASAM = 1; // ADC Sample Control:Sampling begins immediately after
                          // conversion
    AD1CON1bits.AD12B = 0; // 10-bit ADC operation
    AD1CON2bits.SIMSAM = 0; // Sequential sampling of channels

    AD1CON2bits.CHPS = 1; // Converts channels CH0/CH1

    AD1CON3bits.ADRS = 0; // ADC Clock is derived from Systems Clock
    AD1CON3bits.SAMC = 0; // Auto Sample Time = 0 * TAD
    AD1CON3bits.ADCS = 2; // ADC Conversion Clock TAD = Tcy * (ADCS + 1) = (1/40M) * 3 =
                          // 75 ns (13.3 MHz)
                          // ADC Conversion Time for 10-bit Tconv = 12 * TAD = 900 ns (1.1 MHz)

    AD1CON1bits.ADDMABM = 1; // DMA buffers are built in conversion order mode
    AD1CON2bits.SMPI = 0;    // SMPI must be 0

    //AD1CHS0/AD1CHS123:A/D Input Select Register
    AD1CHS0bits.CH0SA = 0;    // MUXA +ve input selection (AIN0) for CH0
    AD1CHS0bits.CH0NA = 0;    // MUXA -ve input selection (VREF-) for CH0

    AD1CHS123bits.CH123SA = 0; // MUXA +ve input selection (AIN0) for CH1
    AD1CHS123bits.CH123NA = 0; // MUXA -ve input selection (VREF-) for CH1

    //AD1PCFGH/AD1PCFGL:Port Configuration Register
    AD1PCFGL = 0xFFFF;
    AD1PCFGH = 0xFFFF;
    AD1PCFGLbits.PCFG0 = 0; // AN0 as Analog Input
    IFS0bits.AD1IF = 0;      // Clear the A/D interrupt flag bit
    IEC0bits.AD1IE = 0;      // Do Not Enable A/D interrupt
    AD1CON1bits.ADON = 1;    // Turn on the A/D converter
}

void initDma0(void)
{
    DMA0CONbits.AMODE = 0;    // Configure DMA for Register indirect with post increment
    DMA0CONbits.MODE = 2;     // Configure DMA for Continuous Ping-Pong mode

    DMA0PAD = (int)&ADC1BUF0;
    DMA0CNT = (NUMSAMP-1);

    DMA0REQ = 13;

    DMA0STA = __builtin_dmaoffset(BufferA);
    DMA0STB = __builtin_dmaoffset(BufferB);

    IFS0bits.DMA0IF = 0;      //Clear the DMA interrupt flag bit
    IEC0bits.DMA0IE = 1;      //Set the DMA interrupt enable bit

    DMA0CONbits.CHEN = 1;
}
```

16.10 DMA なしデバイスのサンプリング / 変換シーケンス例

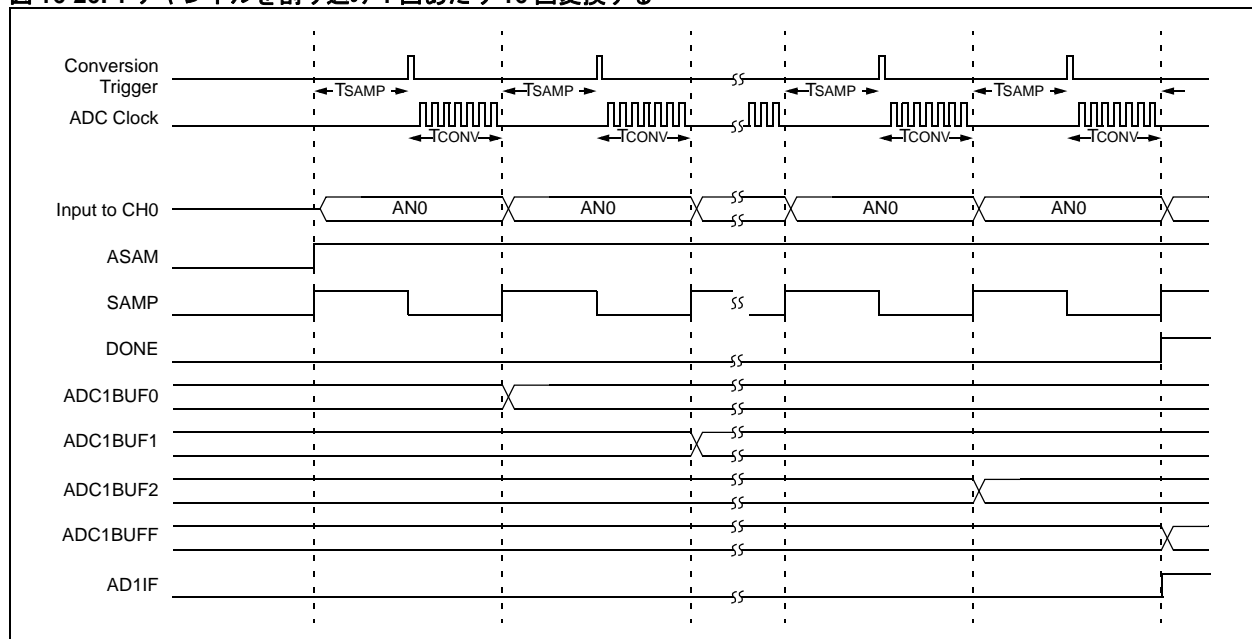
以下のコンフィグレーション例では、各種のサンプリングおよびバッファリング設定での A/D 動作を説明します。各例では、ASAM ビットをセットする事により、自動的にサンプリングを開始します。また、変換トリガがサンプリング終了 / 変換開始をトリガします。

16.10.1 1 チャンネルを複数回サンプリング / 変換する

図 16-26 と表 16-14 に、ADC の基本的なコンフィグレーションを示します。この場合、1 つの S&H チャンネル (CH0) を使用して 1 つの ADC 入力 (AN0) をサンプリング / 変換します。結果は ADC バッファ (ADC1BUF0 ~ ADC1BUFF) に保存されます。このプロセスを 16 回繰り返してバッファがフルになると、ADC モジュールが割り込みを生成します。以上のプロセスを繰り返し実行します。

CHPS ビットで S&H CH0 だけを有効にします。ALTS をクリアする事により、MUXA 入力だけを有効にします。CH0SA ビットと CH0NA ビットで CH0+ 入力に入力チャンネル AN0、CH- 入力に VREF- を選択します。その他の入力選択ビットは全て使用しません。

図 16-26: 1 チャンネルを割り込み 1 回あたり 16 回変換する



dsPIC33F/PIC24H ファミリ リファレンス マニュアル

表 16-14: 1 チャンネルを ADC 割り込み 1 回あたり 16 回変換する
制御ビット

シーケンス選択	
SMPI<3:0> = 1111	サンプリング 16 回ごとに割り込む
CHPS<1:0> = 00	チャンネル CH0
SIMSAM = n/a	1 チャンネル サンプリングには適用せず
BUFM = 0	16 ワード単一結果バッファ
ALTS = 0	常に MUXA 入力選択を使用する

MUXA 入力選択	
CH0SA<3:0> = 0000	CH0+ 入力に AN0 を選択する
CH0NA = 0	CH0- 入力に VREF- を選択する
CSCNA = 0	入カスキャンしない
CSSL<15:0> = n/a	スキャン入力選択を使用しない
CH123SA = n/a	チャンネル CH1/CH2/CH3+ 入力を使用しない
CH123NA<1:0> = n/a	チャンネル CH1/CH2/CH3 - 入力を使用しない

MUXB 入力選択	
CH0SB<3:0> = n/a	チャンネル CH0+ 入力を使用しない
CH0NB = n/a	チャンネル CH0- 入力を使用しない
CH123SB = n/a	チャンネル CH1/CH2/CH3 + 入力を使用しない
CH123NA<1:0> = n/a	チャンネル CH1/CH2/CH3 - 入力を使用しない

初回 ADC 割り込み時の ADC バッファの状態	
ADC1BUF0	AN0 サンプル 1
ADC1BUF1	AN0 サンプル 2
ADC1BUF2	AN0 サンプル 3
ADC1BUF3	AN0 サンプル 4
ADC1BUF4	AN0 サンプル 5
ADC1BUF5	AN0 サンプル 6
ADC1BUF6	AN0 サンプル 7
ADC1BUF7	AN0 サンプル 8
ADC1BUF8	AN0 サンプル 9
ADC1BUF9	AN0 サンプル 10
ADC1BUFA	AN0 サンプル 11
ADC1BUFB	AN0 サンプル 12
ADC1BUFC	AN0 サンプル 13
ADC1BUFD	AN0 サンプル 14
ADC1BUFE	AN0 サンプル 15
ADC1BUFF	AN0 サンプル 16

動作シーケンス

MUXA 入力をサンプリングする: AN0 → CH0
CH0 を変換して ADC1BUF0 に書き込む
MUXA 入力をサンプリングする: AN0 → CH0
CH0 を変換して ADC1BUF1 に書き込む
MUXA 入力をサンプリングする: AN0 → CH0
CH0 を変換して ADC1BUF2 に書き込む
MUXA 入力をサンプリングする: AN0 → CH0
CH0 を変換して ADC1BUF3 に書き込む
MUXA 入力をサンプリングする: AN0 → CH0
CH0 を変換して ADC1BUF4 に書き込む
MUXA 入力をサンプリングする: AN0 → CH0
CH0 を変換して ADC1BUF5 に書き込む
MUXA 入力をサンプリングする: AN0 → CH0
CH0 を変換して ADC1BUF6 に書き込む
MUXA 入力をサンプリングする: AN0 → CH0
CH0 を変換して ADC1BUF7 に書き込む
MUXA 入力をサンプリングする: AN0 → CH0
CH0 を変換して ADC1BUF8 に書き込む
MUXA 入力をサンプリングする: AN0 → CH0
CH0 を変換して ADC1BUF9 に書き込む
MUXA 入力をサンプリングする: AN0 → CH0
CH0 を変換して ADC1BUFA に書き込む
MUXA 入力をサンプリングする: AN0 → CH0
CH0 を変換して ADC1BUFB に書き込む
MUXA 入力をサンプリングする: AN0 → CH0
CH0 を変換して ADC1BUFC に書き込む
MUXA 入力をサンプリングする: AN0 → CH0
CH0 を変換して ADC1BUFD に書き込む
MUXA 入力をサンプリングする: AN0 → CH0
CH0 を変換して ADC1BUFE に書き込む
MUXA 入力をサンプリングする: AN0 → CH0
CH0 を変換して ADC1BUFF に書き込む
ADC 割り込み
以上を繰り返す

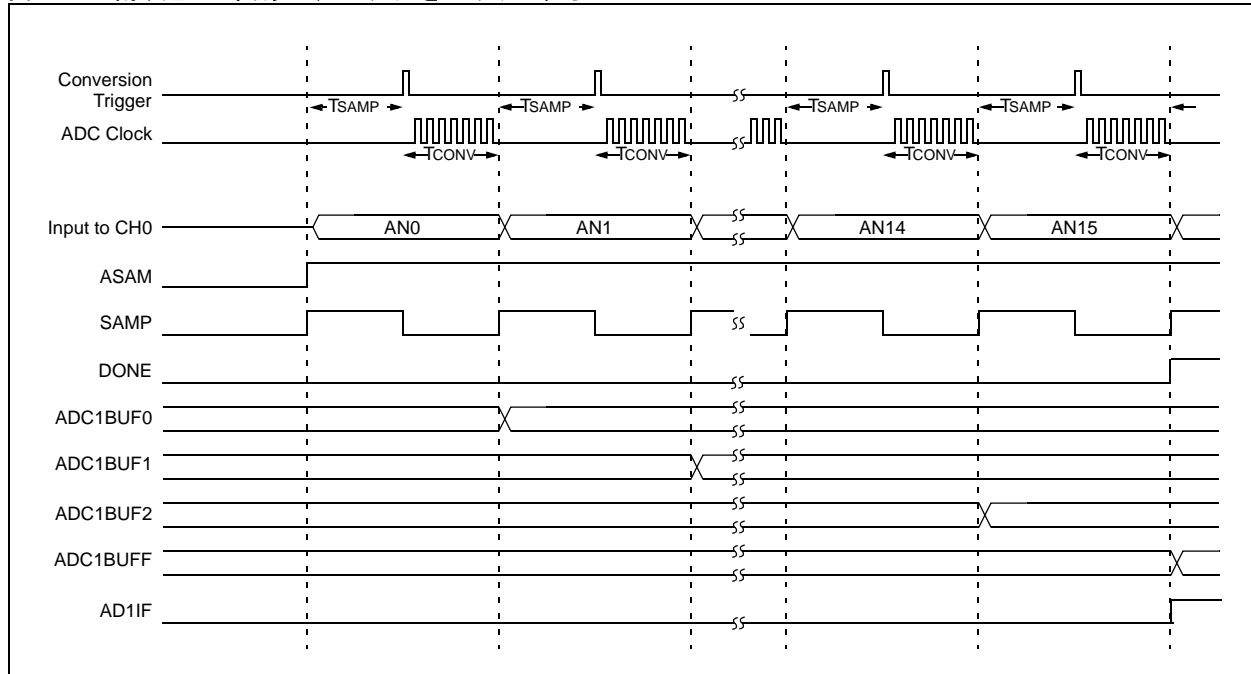
2 回目 ADC 割り込み時の ADC バッファの状態	
	AN0 サンプル 17
	AN0 サンプル 18
	AN0 サンプル 19
	AN0 サンプル 20
	AN0 サンプル 21
	AN0 サンプル 22
	AN0 サンプル 23
	AN0 サンプル 24
	AN0 サンプル 25
	AN0 サンプル 26
	AN0 サンプル 27
	AN0 サンプル 28
	AN0 サンプル 29
	AN0 サンプル 30
	AN0 サンプル 31
	AN0 サンプル 32

16.10.2 全アナログ入力をスキャンしながら A/D 変換する

図 16-27 と表 16-15 に、利用可能な全アナログ入力チャンネルを 1 つの S&H チャンネル (CH0) でサンプリング / 変換する場合のセットアップ例を示します。ADC 制御レジスタ 2 (ADxCON2<10>) のスキャン入力選択ビット (CSCNA) をセットする事により、CH0 正極性入力への ADC 入力をスキャンします。その他の条件は 16.10.1 「1 チャンネルを複数回サンプリング / 変換する」と同じです。

CH0 は最初に AN0 入力をサンプリング / 変換し、次に AN1 入力をサンプリング / 変換します。この入カスキャンプロセスは、バッファがフルになるまで 16 回繰り返されます。結果は ADC バッファ (ADC1BUFA ~ ADC1BUFF) に保存されます。その後 ADC モジュールが割り込みを生成します。以上のプロセスを繰り返し実行します。

図 16-27: 割り込み 1 回あたり 16 入力をスキャンする



dsPIC33F/PIC24H ファミリ リファレンス マニュアル

表 16-15: ADC 割り込み 1 回あたり 16 入力をスキャンする
制御ビット

シーケンス選択	
SMPI<3:0> = 1111	サンプリング 16 回ごとに割り込む
CHPS<1:0> = 00	チャンネル CH0
SIMSAM = n/a	1 チャンネル サンプリングには適用せず
BUFM = 0	16 ワード単一結果バッファ
ALTS = 0	常に MUXA 入力選択を使用する

MUXA 入力選択	
CH0SA<3:0> = n/a	CSCNA によるオーバーライド
CH0NA = 0	CH0- 入力に VREF- を選択する
CSCNA = 1	CH0+ 入力をスキャンする
CSSL<15:0> = 1111 1111 1111 1111	AN0 ~ AN15 をスキャンする
CH123SA = n/a	チャンネル CH1/CH2/CH3+ 入力を使用しない
CH123NA<1:0> = n/a	チャンネル CH1/CH2/CH3- 入力を使用しない

MUXB 入力選択	
CH0SB<3:0> = n/a	チャンネル CH0+ 入力を使用しない
CH0NB = n/a	チャンネル CH0- 入力を使用しない
CH123SB = n/a	チャンネル CH1/CH2/CH3+ 入力を使用しない
CH123NB<1:0> = n/a	チャンネル CH1/CH2/CH3- 入力を使用しない

動作シーケンス

MUXA 入力をサンプリングする : AN0 → CH0
CH0 を変換して ADC1BUF0 に書き込む
MUXA 入力をサンプリングする : AN1 → CH0
CH0 を変換して ADC1BUF1 に書き込む
MUXA 入力をサンプリングする : AN2 → CH0
CH0 を変換して ADC1BUF2 に書き込む
MUXA 入力をサンプリングする : AN3 → CH0
CH0 を変換して ADC1BUF3 に書き込む
MUXA 入力をサンプリングする : AN4 → CH0
CH0 を変換して ADC1BUF4 に書き込む
MUXA 入力をサンプリングする : AN5 → CH0
CH0 を変換して ADC1BUF5 に書き込む
MUXA 入力をサンプリングする : AN6 → CH0
CH0 を変換して ADC1BUF6 に書き込む
MUXA 入力をサンプリングする : AN7 → CH0
CH0 を変換して ADC1BUF7 に書き込む
MUXA 入力をサンプリングする : AN8 → CH0
CH0 を変換して ADC1BUF8 に書き込む
MUXA 入力をサンプリングする : AN9 → CH0
CH0 を変換して ADC1BUF9 に書き込む
MUXA 入力をサンプリングする : AN10 → CH0
CH0 を変換して ADC1BUFA に書き込む
MUXA 入力をサンプリングする : AN11 → CH0
CH0 を変換して ADC1BUFB に書き込む
MUXA 入力をサンプリングする : AN12 → CH0
CH0 を変換して ADC1BUFC に書き込む
MUXA 入力をサンプリングする : AN13 → CH0
CH0 を変換して ADC1BUFD に書き込む
MUXA 入力をサンプリングする : AN14 → CH0
CH0 を変換して ADC1BUFE に書き込む
MUXA 入力をサンプリングする : AN15 → CH0
CH0 を変換して ADC1BUFF に書き込む
ADC 割り込み
以上を繰り返す

初回 ADC 割り込み時の ADC バッファの状態

ADC1BUF0	AN0 サンプル 1
ADC1BUF1	AN1 サンプル 2
ADC1BUF2	AN2 サンプル 3
ADC1BUF3	AN3 サンプル 4
ADC1BUF4	AN4 サンプル 5
ADC1BUF5	AN5 サンプル 6
ADC1BUF6	AN6 サンプル 7
ADC1BUF7	AN7 サンプル 8
ADC1BUF8	AN8 サンプル 9
ADC1BUF9	AN9 サンプル 10
ADC1BUFA	AN10 サンプル 11
ADC1BUFB	AN11 サンプル 12
ADC1BUFC	AN12 サンプル 13
ADC1BUFD	AN13 サンプル 14
ADC1BUFE	AN14 サンプル 15
ADC1BUFF	AN15 サンプル 16

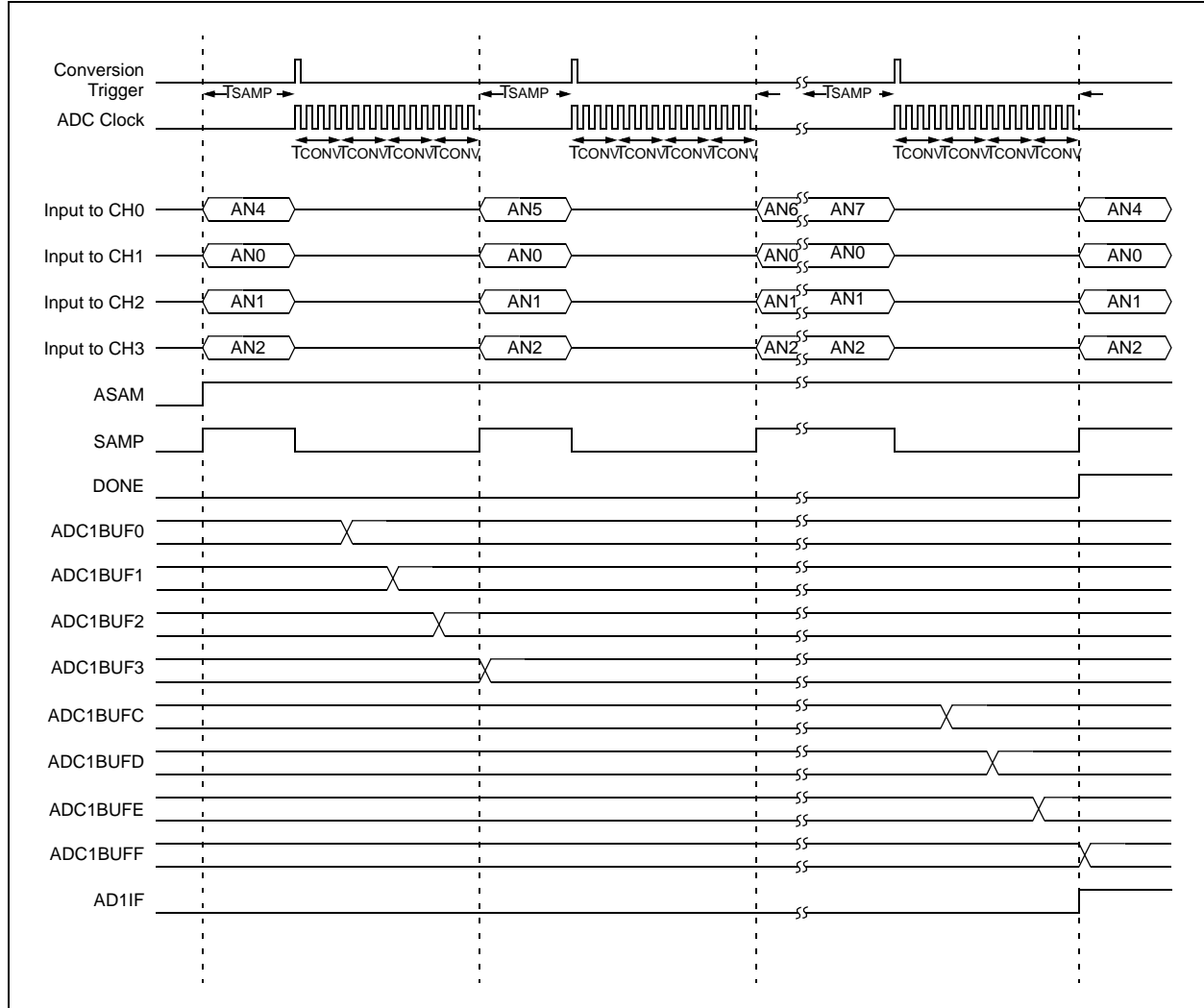
2 回目 ADC 割り込み時の ADC バッファの状態

AN0 サンプル 17
AN1 サンプル 18
AN2 サンプル 19
AN3 サンプル 20
AN4 サンプル 21
AN5 サンプル 22
AN6 サンプル 23
AN7 サンプル 24
AN8 サンプル 25
AN9 サンプル 26
AN10 サンプル 27
AN11 サンプル 28
AN12 サンプル 29
AN13 サンプル 30
AN14 サンプル 31
AN15 サンプル 32

16.10.3 3 入力を高頻度でサンプリングし、別の 4 入力をスキャンする

図 16-28 と表 16-16 に、S&H チャンネル CH 1、CH2、CH3 を使用して 3 つの入力を高頻度でサンプリングし、S&H チャンネル CH0 を使用して他の 4 つの入力をスキャンして低頻度でサンプリングするように ADC モジュールを設定する方法を示します。この場合、MUXA 入力のみを使用し、4 チャンネル全てを同時にサンプリングします。4 つの入力 (AN4、AN5、AN6、AN7) を CH0 でスキャンし、AN0、AN1、AN2 をそれぞれ CH1、CH2、CH3 への固定入力とします。つまり、16 回のサンプリング中に AN0、AN1、AN2 をそれぞれ 4 回サンプリングし、AN4、AN5、AN6、AN7 をそれぞれ 1 回だけサンプリングします。

図 16-28: 1 回の割り込みあたり 3 入力を 4 回、4 入力を 1 回変換する



dsPIC33F/PIC24H ファミリ リファレンス マニュアル

表 16-16: 1 回の ADC 割り込みあたり 3 入力を 4 回、4 入力を 1 回変換する
制御ビット

SMPI<3:0> = 0011 サンプリング 4 回ごとに割り込む
CHPS<1:0> = 1X CH0/CH1/CH2/CH3 をサンプリングする
SIMSAM = 1 全てのチャンネルを同時サンプリングする
BUFM = 0 16 ワード単一結果バッファ
ALTS = 0 常に MUXA 入力選択を使用する

MUXA 入力選択

CH0SA<3:0> = n/a CSCNA によるオーバーライド
CH0NA = 0 CH0- 入力に VREF- を選択する
CSCNA = 1 CH0+ 入力をスキャンする
CSSL<15:0> = 0000 0000 1111 0000 AN4、AN5、AN6、AN7 をスキャンする
CH123SA = 0 CH1+ = AN0、CH2+ = AN1、CH3+ = AN2
CH123NA<1:0> = 0X CH1-、CH2-、CH3- = VREF-

MUXB 入力選択

CH0SB<3:0> = n/a チャンネル CH0+ 入力を使用しない
CH0NB = n/a チャンネル CH0- 入力を使用しない
CH123SB = n/a チャンネル CH1/CH2/CH3+ 入力を使用しない
CH123NB<1:0> = n/a チャンネル CH1/CH2/CH3- 入力を使用しない

初回 ADC 割り込み時の ADC バッファの状態

ADC1BUF0	AN4 サンプル 1
ADC1BUF1	AN0 サンプル 1
ADC1BUF2	AN1 サンプル 1
ADC1BUF3	AN2 サンプル 1
ADC1BUF4	AN5 サンプル 1
ADC1BUF5	AN0 サンプル 2
ADC1BUF6	AN1 サンプル 2
ADC1BUF7	AN2 サンプル 2
ADC1BUF8	AN6 サンプル 1
ADC1BUF9	AN0 サンプル 3
ADC1BUFA	AN1 サンプル 3
ADC1BUFB	AN2 サンプル 3
ADC1BUFC	AN7 サンプル 1
ADC1BUFD	AN0 サンプル 4
ADC1BUFE	AN1 サンプル 4
ADC1BUFF	AN2 サンプル 4

動作シーケンス

MUXA 入力をサンプリングする: AN4 → CH0、AN0 → CH1、AN1 → CH2、AN2 → CH3
CH0 を変換して ADC1BUF0 に書き込む
CH1 を変換して ADC1BUF1 に書き込む
CH2 を変換して ADC1BUF2 に書き込む
CH3 を変換して ADC1BUF3 に書き込む
MUXA 入力をサンプリングする: AN5 → CH0、AN0 → CH1、AN1 → CH2、AN2 → CH3
CH0 を変換して ADC1BUF4 に書き込む
CH1 を変換して ADC1BUF5 に書き込む
CH2 を変換して ADC1BUF6 に書き込む
CH3 を変換して ADC1BUF7 に書き込む
MUXA 入力をサンプリングする: AN6 → CH0、AN0 → CH1、AN1 → CH2、AN2 → CH3
CH0 を変換して ADC1BUF8 に書き込む
CH1 を変換して ADC1BUF9 に書き込む
CH2 を変換して ADC1BUF10 に書き込む
CH3 を変換して ADC1BUF11 に書き込む
MUXA 入力をサンプリングする: AN7 → CH0、AN0 → CH1、AN1 → CH2、AN2 → CH3
CH0 を変換して ADC1BUFC に書き込む
CH1 を変換して ADC1BUFD に書き込む
CH2 を変換して ADC1BUFE に書き込む
CH3 を変換して ADC1BUFF に書き込む
ADC 割り込み
以上を繰り返す

2 回目 ADC 割り込み時の ADC バッファの状態

AN4 サンプル 2
AN0 サンプル 5
AN1 サンプル 5
AN2 サンプル 5
AN5 サンプル 2
AN0 サンプル 6
AN1 サンプル 6
AN2 サンプル 6
AN6 サンプル 2
AN0 サンプル 7
AN1 サンプル 7
AN2 サンプル 7
AN7 サンプル 2
AN0 サンプル 8
AN1 サンプル 8
AN2 サンプル 8

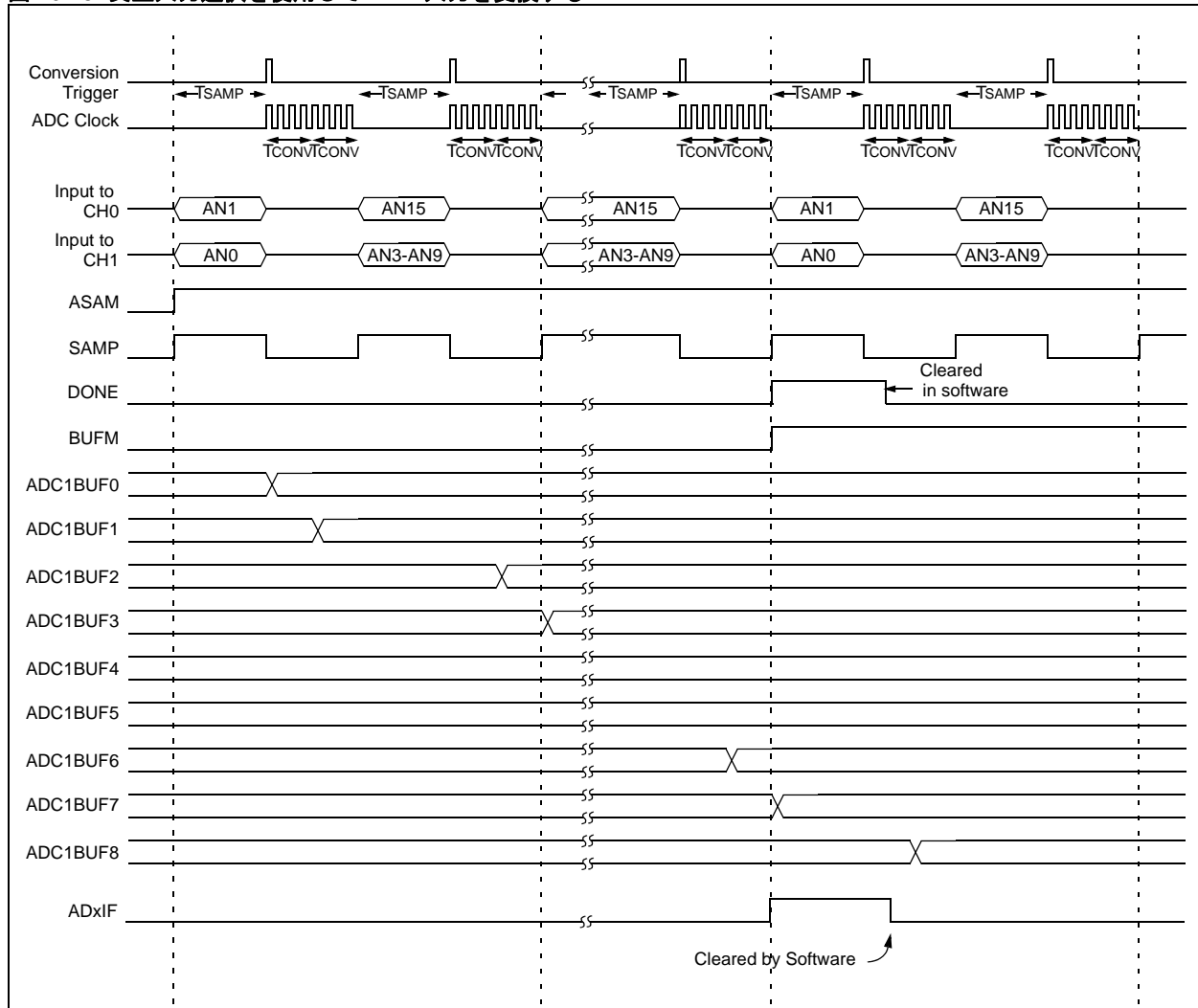
Note: この同時サンプリング事例では、1 回のサンプリングと 4 回の変換が 1 回のサンプリング / 変換シーケンスとして扱われます。従って SMPI<3:0> = 0011(サンプリング 4 回ごとに割り込む) に設定する事により、16 個の変換結果を ADC1BUF0 ~ ADC1BUFF にバッファリングした後に ADC 割り込みを生成します。

16.10.4 MUXA/MUXB 交互入力選択を使用する

図 16-29 と表 16-17 に、MUXA と MUXB に割り当てた入力を交互にサンプリングする例を示します。この例では、2 チャンネルの同時サンプリングを有効にします。ALTS ビット (ADCxCON2<0>) をセットする事により、交互入力選択を有効にします。最初のサンプリングには CH0SA、CH0NA、CH123SA、CH123NA ビットで指定した MUXA 入力を使用します。次のサンプリングには CH0SB、CH0NB、CH123SB、CH123NB ビットで指定した MUXB 入力を使用します。この例では、MUXB 入力選択で 1 つの S&H (CH1) に 2 つのアナログ入力を差動入力として選択する事により (AN3-AN9) をサンプリングします。

4 つの S&H チャンネルを交互入力選択なしで使用した場合の変換数は、この例のように 2 チャンネルを交互入力選択して使用した場合の変換数と同じです。しかし CH1、CH2、CH3 チャンネルのアナログ入力選択自由度が限られているため、この例の方が 4 チャンネルを使用するよりも柔軟に入力を選択できます。

図 16-29: 交互入力選択を使用して 2 x 2 入力を変換する



dsPIC33F/PIC24H ファミリ リファレンス マニュアル

表 16-17: 交互入力選択を使用して 2 x 2 入力を変換する
制御ビット

シーケンス選択	
SMPI<3:0> = 0011	サンプリング 4 回ごとに割り込む
CHPS<1:0> = 01	チャンネル CH0 と CH1 をサンプリングする
SIMSAM = 1	全てのチャンネルを同時にサンプリングする
BUFM = 1	2 x 8 ワードの結果バッファ
ALTS = 1	MUXA/MUXB 入力の交互選択

MUXA 入力選択	
CH0SA<3:0> = 0001	CH0+ 入力に AN1 を選択する
CH0NA = 0	CH0- 入力に VREF- を選択する
CSCNA = 0	入カスキャンしない
CSSL<15:0> = n/a	スキャン入力選択を使用しない
CH123SA = 0	CH1+ = AN0、CH2+ = AN1、CH3+ = AN2
CH123NA<1:0> = 0x	CH1-、CH2-、CH3- = VREF-

MUXB 入力選択	
CH0SB<3:0> = 1111	CH0+ 入力に AN15 を選択する
CH0NB = 0	CH0- 入力に VREF- を選択する
CH123SB = 1	CH1+ = AN3、CH2+ = AN4、CH3+ = AN5
CH123NB<1:0> = 11	CH1- = AN9、CH2- = AN10、CH3- = AN11

初回 ADC 割り込み時の
ADC バッファの状態

ADC1BUF0	AN1 サンプル 1
ADC1BUF1	AN0 サンプル 1
ADC1BUF2	AN15 サンプル 2
ADC1BUF3	(AN3-AN9) サンプル 2
ADC1BUF4	AN1 サンプル 3
ADC1BUF5	AN0 サンプル 3
ADC1BUF6	AN15 サンプル 4
ADC1BUF7	(AN3-AN9) サンプル 4
ADC1BUF8	
ADC1BUF9	
ADC1BUFA	
ADC1BUFB	
ADC1BUFC	
ADC1BUFD	
ADC1BUFE	
ADC1BUFF	

動作シーケンス

MUXA 入力をサンプリング: AN1 → CH0, AN0 → CH1
CH0 を変換して ADC1BUF0 に書き込む
CH1 を変換して ADC1BUF1 に書き込む
MUXB 入力をサンプリング: AN15→CH0,(AN3-AN9)→CH1
CH0 を変換して ADC1BUF2 に書き込む
CH1 を変換して ADC1BUF3 に書き込む
MUXA 入力をサンプリング: AN1 → CH0, AN0 → CH1
CH0 を変換して ADC1BUF4 に書き込む
CH1 を変換して ADC1BUF5 に書き込む
MUXB 入力をサンプリング: AN15→CH0,(AN3-AN9)→CH1
CH0 を変換して ADC1BUF6 に書き込む
CH1 を変換して ADC1BUF7 に書き込む
割り込み; バッファを変更する
MUXA 入力をサンプリング: AN1 → CH0, AN0 → CH1
CH0 を変換して ADC1BUF8 に書き込む
CH1 を変換して ADC1BUF9 に書き込む
MUXB 入力をサンプリング: AN15→CH0,(AN3-AN9)→CH1
CH0 を変換して ADC1BUFA に書き込む
CH1 を変換して ADC1BUFB に書き込む
MUXA 入力をサンプリング: AN1 → CH0, AN0 → CH1
CH0 を変換して ADC1BUFC に書き込む
CH1 を変換して ADC1BUFD に書き込む
MUXB 入力をサンプリング: AN15→CH0,(AN3-AN9)→CH1
CH0 を変換して ADC1BUFE に書き込む
CH1 を変換して ADC1BUFF に書き込む
ADC 割り込み; バッファを変更する
以上を繰り返す

2 回目 ADC 割り込み時の
ADC バッファの状態

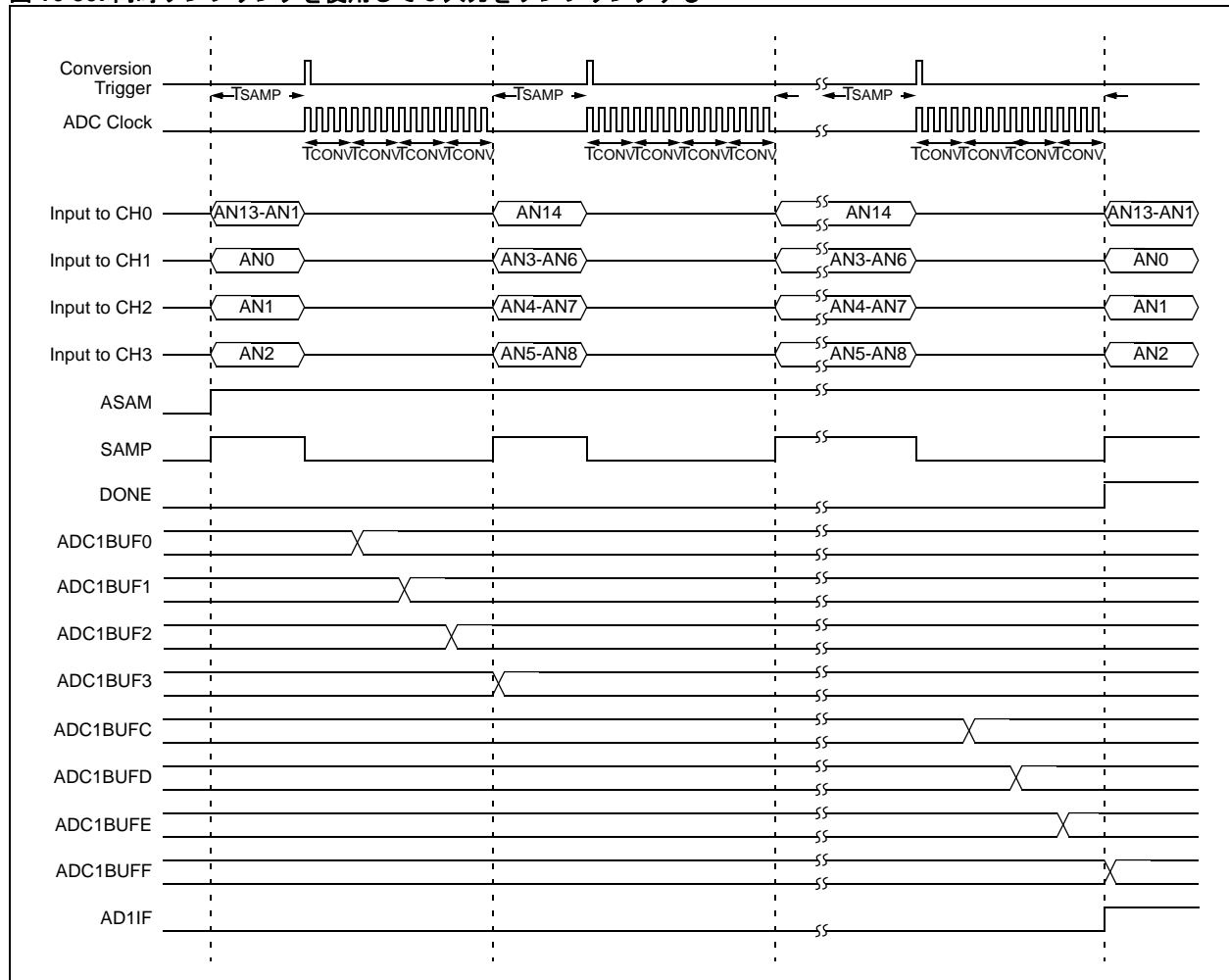
	AN1 サンプル 5
	AN0 サンプル 5
	AN15 サンプル 6
	(AN3-AN9) サンプル 6
	AN1 サンプル 7
	AN0 サンプル 7
	AN15 サンプル 8
	(AN3-AN9) サンプル 8

16.10.5 同時サンプリングを使用して 8 入力をサンプリングする

この例と次の例の設定は似ていますが、この例では同時サンプリング (SIMSAM = 1) を使用するのに対して次の例では逐次サンプリング (SIMSAM = 0) を使用するという点で異なります。両例では、交互入力選択を使用し、S&H に差動入力を指定します。

図 16-30 と表 16-18 に、同時サンプリングの例を示します。同時サンプリングで複数チャンネルを変換する場合、ADC モジュールは全てのチャンネルをサンプリングしてから変換シーケンスを実行します。この例では ASAM ビットをセットするため、変換完了後自動的にサンプリングが開始されます。

図 16-30: 同時サンプリングを使用して 8 入力をサンプリングする



dsPIC33F/PIC24H ファミリ リファレンス マニュアル

表 16-18: 同時サンプリングを使用して 8 入力をサンプリングする
制御ビット

シーケンス選択	
SMPI<3:0> = 0001	サンプリング 2 回ごとに割り込む
CHPS<1:0> = 1X	CH0/CH1/CH2/CH3 をサンプリングする
SIMSAM = 1	全てのチャンネルを同時にサンプリングする
BUFM = 0	16 ワード単一結果バッファ
ALTS = 1	MUXA/MUXB 入力の交互選択

MUXA 入力選択	
CH0SA<3:0> = 1101	CH0+ 入力に AN13 を選択する
CH0NA = 1	CH0- 入力に AN1 を選択する
CSCNA = 0	入カスキャンしない
CSSL<15:0> = n/a	スキャン入力選択を使用しない
CH123SA = 0	CH1+ = AN0、CH2+ = AN1、CH3+ = AN2
CH123NA<1:0> = 0X	CH1-、CH2-、CH3- = VREF-

MUXB 入力選択	
CH0SB<3:0> = 1110	CH0+ 入力に AN14 を選択する
CH0NB = 0	CH0- 入力に VREF- を選択する
CH123SB = 1	CH1+ = AN3、CH2+ = AN4、CH3+ = AN5
CH123NB<1:0> = 10	CH1- = AN6、CH2- = AN7、CH3- = AN8

初回 ADC 割り込み時の ADC バッファの状態	
ADC1BUF0	(AN13-AN1) サンプル 1
ADC1BUF1	AN0 サンプル 1
ADC1BUF2	AN1 サンプル 1
ADC1BUF3	AN2 サンプル 1
ADC1BUF4	AN14 サンプル 1
ADC1BUF5	(AN3-AN6) サンプル 1
ADC1BUF6	(AN4-AN7) サンプル 1
ADC1BUF7	(AN5-AN8) サンプル 1
ADC1BUF8	(AN13-AN1) サンプル 2
ADC1BUF9	AN0 サンプル 2
ADC1BUFA	AN1 サンプル 2
ADC1BUFB	AN2 サンプル 2
ADC1BUFC	AN14 サンプル 2
ADC1BUFD	(AN3-AN6) サンプル 2
ADC1BUFE	(AN4-AN7) サンプル 2
ADC1BUFF	(AN5-AN8) サンプル 2

動作シーケンス	
MUXA 入力をサンプリングする:	
(AN13-AN1)→CH0、AN0→CH1、AN1→CH2、AN2→CH3	
CH0 を変換して ADC1BUF0 に書き込む	
CH1 を変換して ADC1BUF1 に書き込む	
CH2 を変換して ADC1BUF2 に書き込む	
CH3 を変換して ADC1BUF3 に書き込む	
MUXB 入力をサンプリングする:	
AN14 → CH0、	
(AN3-AN6)→CH1、(AN4-AN7)→CH2、(AN5-AN8)→CH3	
CH0 を変換して ADC1BUF4 に書き込む	
CH1 を変換して ADC1BUF5 に書き込む	
CH2 を変換して ADC1BUF6 に書き込む	
CH3 を変換して ADC1BUF7 に書き込む	
MUXA 入力をサンプリングする:	
(AN13-AN1)→CH0、AN0→CH1、AN1→CH2、AN2→CH3	
CH0 を変換して ADC1BUF8 に書き込む	
CH1 を変換して ADC1BUF9 に書き込む	
CH2 を変換して ADC1BUFA に書き込む	
CH3 を変換して ADC1BUFB に書き込む	
MUXB 入力をサンプリングする:	
AN14 → CH0、	
(AN3-AN6)→CH1、(AN4-AN7)→CH2、(AN5-AN8)→CH3	
CH0 を変換して ADC1BUFC に書き込む	
CH1 を変換して ADC1BUFD に書き込む	
CH2 を変換して ADC1BUFE に書き込む	
CH3 を変換して ADC1BUFF に書き込む	
ADC 割り込み	
以上を繰り返す	

2 回目 ADC 割り込み時の ADC バッファの状態	
(AN13-AN1) サンプル 3	
AN0 サンプル 3	
AN1 サンプル 3	
AN2 サンプル 3	
AN14 サンプル 3	
(AN3-AN6) サンプル 3	
(AN4-AN7) サンプル 3	
(AN5-AN8) サンプル 3	
(AN13-AN1) サンプル 4	
AN0 サンプル 4	
AN1 サンプル 4	
AN2 サンプル 4	
AN14 サンプル 4	
(AN3-AN6) サンプル 4	
(AN4-AN7) サンプル 4	
(AN5-AN8) サンプル 4	

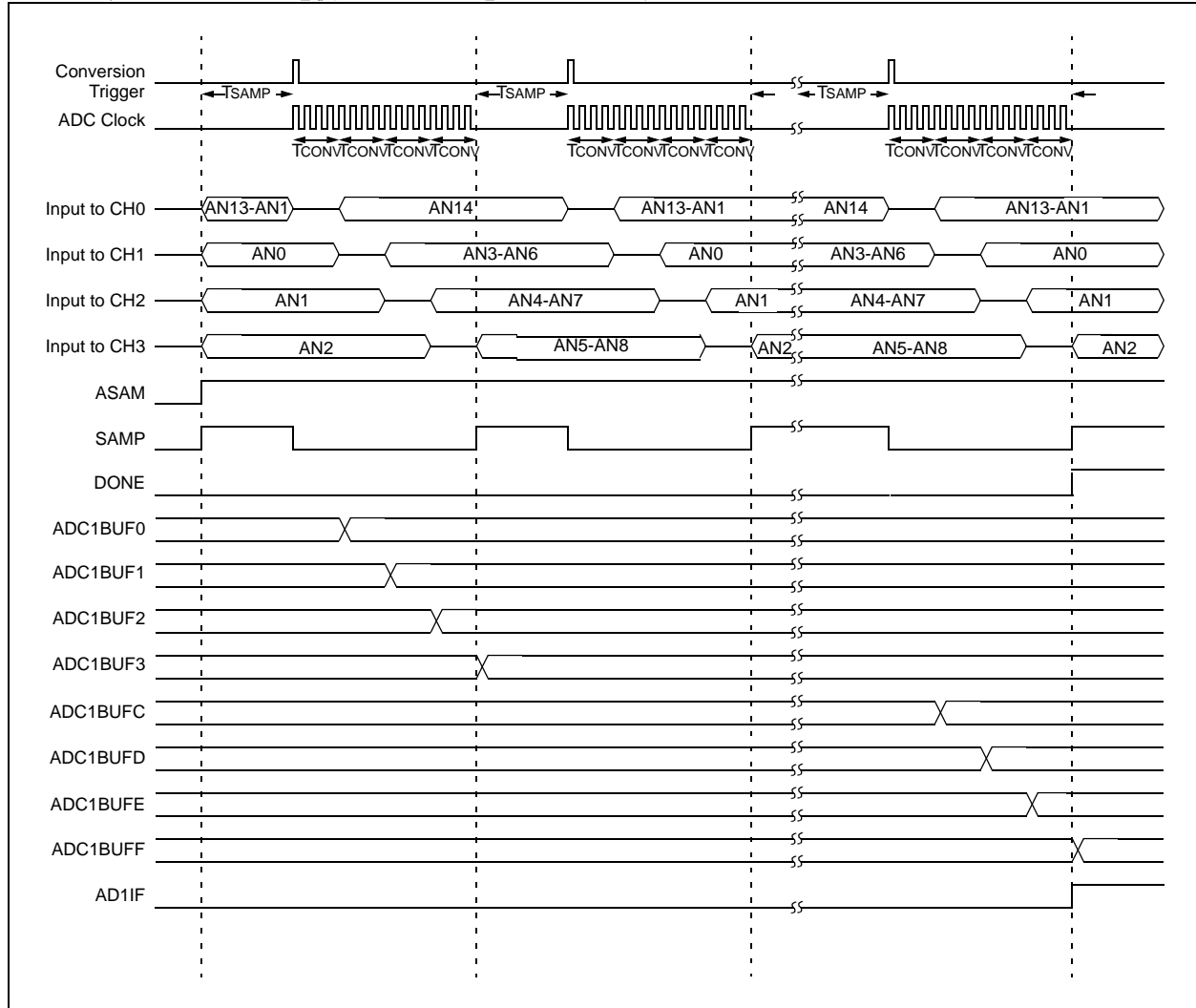
16.10.6 逐次サンプリングを使用して 8 入力をサンプリングする

図 16-31 と表 16-19 に、逐次サンプリングの例を示します。逐次サンプリングで複数チャンネルを変換する場合、ADC モジュールは可能な限り早期に 1 チャンネルのサンプリングを開始し、続いてそのチャンネルの変換を実行します。この例では ASAM ビットをセットするため、各チャンネルのサンプリングは、そのチャンネルの変換が完了した後に自動的に開始されます。

ASAM ビットをクリアした場合、変換が完了しても SAMP ビットがセットされるまでサンプリングは開始されません。

複数チャンネルを使用する場合、逐次サンプリングでは他チャンネルで変換実行中に 1 チャンネルをサンプリングできるため、同時サンプリングよりも長いサンプリング時間が得られます。

図 16-31: 逐次サンプリングを使用して 8 入力をサンプリングする



dsPIC33F/PIC24H ファミリ リファレンス マニュアル

表 16-19: 逐次サンプリングを使用して 8 入力をサンプリングする
制御ビット

シーケンス選択	
SMPI<3:0> = 1111	サンプリング 16 回ごとに割り込む
CHPS<1:0> = 1X	CH0/CH1/CH2/CH3 をサンプリングする
SIMSAM = 0	全てのチャンネルを逐次サンプリングする
BUFM = 0	16 ワード単一結果バッファ
ALTS = 1	MUXA/MUXB 入力の交互選択

MUXA 入力選択	
CH0SA<4:0> = 01101	CH0+ 入力に AN13 を選択する
CH0NA = 1	CH0- 入力に AN1 を選択する
CSCNA = 0	入カスキャンしない
CSSL<15:0> = n/a	スキャン入力選択を使用しない
CH123SA = 0	CH1+ = AN0、CH2+ = AN1、CH3+ = AN2
CH123NA<1:0> = 0X	CH1-、CH2-、CH3- = VREF-

MUXB 入力選択	
CH0SB<4:0> = 01110	CH0+ 入力に AN14 を選択する
CH0NB = 0	CH0- 入力に VREF- を選択する
CH123SB = 1	CH1+ = AN3、CH2+ = AN4、CH3+ = AN5
CH123NB<1:0> = 10	CH1- = AN6、CH2- = AN7、CH3- = AN8

動作シーケンス

サンプリング : (AN13-AN1) → CH0
CH0 を変換して ADC1BUF0 に書き込む
サンプリング : AN0 → CH1
CH1 を変換して ADC1BUF1 に書き込む
サンプリング : AN1 → CH2
CH2 を変換して ADC1BUF2 に書き込む
サンプリング : AN2 → CH3
CH3 を変換して ADC1BUF3 に書き込む
サンプリング : AN14 → CH0
CH0 を変換して ADC1BUF4 に書き込む
サンプリング : (AN3-AN6) → CH1
CH1 を変換して ADC1BUF5 に書き込む
サンプリング : (AN4-AN7) → CH2
CH2 を変換して ADC1BUF6 に書き込む
サンプリング : (AN5-AN8) → CH3
CH3 を変換して ADC1BUF7 に書き込む
サンプリング : (AN13-AN1) → CH0
CH0 を変換して ADC1BUF8 に書き込む
サンプリング : AN0 → CH1
CH1 を変換して ADC1BUF9 に書き込む
サンプリング : AN1 → CH2
CH2 を変換して ADC1BUFA に書き込む
サンプリング : AN2 → CH3
CH3 を変換して ADC1BUFB に書き込む
サンプリング : AN14 → CH0
CH0 を変換して ADC1BUFC に書き込む
サンプリング : (AN3-AN6) → CH1
CH1 を変換して ADC1BUFD に書き込む
サンプリング : (AN4-AN7) → CH2
CH2 を変換して ADC1BUFE に書き込む
サンプリング : (AN5-AN8) → CH3
CH3 を変換して ADC1BUFF に書き込む
ADC 割り込み
以上を繰り返す

初回 ADC 割り込み時の ADC バッファの状態

ADC1BUF0	(AN13-AN1) サンプル 1
ADC1BUF1	AN0 サンプル 1
ADC1BUF2	AN1 サンプル 1
ADC1BUF3	AN2 サンプル 1
ADC1BUF4	AN14 サンプル 1
ADC1BUF5	(AN3-AN6) サンプル 1
ADC1BUF6	(AN4-AN7) サンプル 1
ADC1BUF7	(AN5-AN8) サンプル 1
ADC1BUF8	(AN13-AN1) サンプル 2
ADC1BUF9	AN0 サンプル 2
ADC1BUFA	AN1 サンプル 2
ADC1BUFB	AN2 サンプル 2
ADC1BUFC	AN14 サンプル 2
ADC1BUFD	(AN3-AN6) サンプル 2
ADC1BUFE	(AN4-AN7) サンプル 2
ADC1BUFF	(AN5-AN8) サンプル 2

2 回目 ADC 割り込み時の ADC バッファの状態

(AN13-AN1) サンプル 3
AN0 サンプル 3
AN1 サンプル 3
AN2 サンプル 3
AN14 サンプル 3
(AN3-AN6) サンプル 3
(AN4-AN7) サンプル 3
(AN5-AN8) サンプル 3
(AN13-AN1) サンプル 4
AN0 サンプル 4
AN1 サンプル 4
AN2 サンプル 4
AN14 サンプル 4
(AN3-AN6) サンプル 4
(AN4-AN7) サンプル 4
(AN5-AN8) サンプル 4

16.11 DMA 付きデバイスのサンプリング/変換シーケンス例

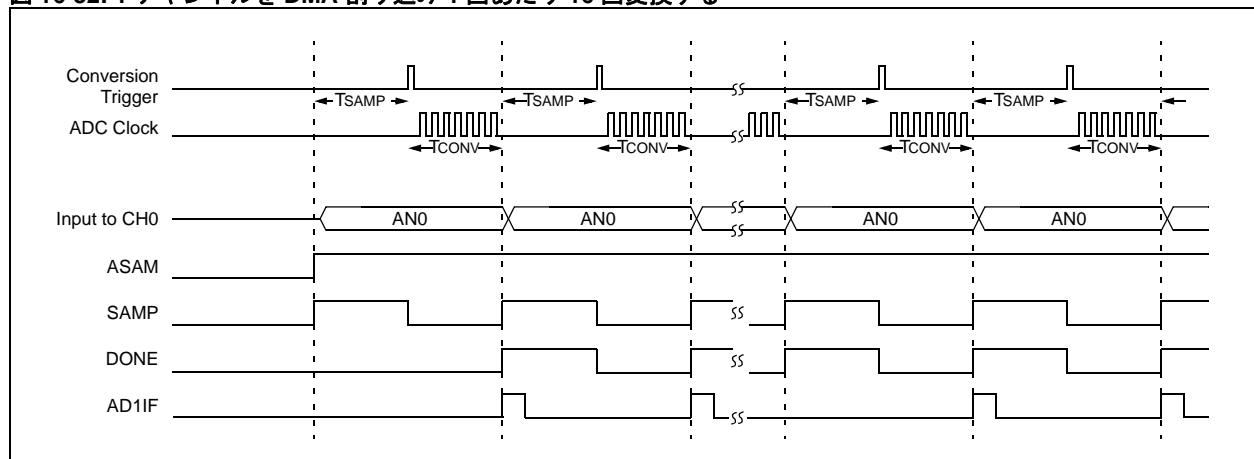
以下のコンフィグレーション例では、各種のサンプリングおよびバッファリング設定での A/D 動作を説明します。各例では、ASAM ビットをセットする事により、自動的にサンプリングを開始します。変換トリガによりサンプリングが終了して変換が開始されます。

16.11.1 1 チャンネルを複数回サンプリング/変換する

図 16-32 と表 16-20 に、ADC の基本的なコンフィグレーションを示します。この場合、1 つの S&H チャンネル (CH0) を使用して 1 つの ADC 入力 (AN0) をサンプリング/変換します。結果はユーザが定義した DMA RAM バッファに保存されます。このプロセスを 16 回繰り返してバッファがフルになると、DMA モジュールが割り込みを生成します。以上のプロセスを繰り返し実行します。

CHPS<1:0> ビットで S&H CH0 だけを有効にします。ALTS をクリアする事により、MUXA 入力だけを有効にします。CH0SA ビットと CH0NA ビットで CH0+ 入力に入力チャンネル AN0、CH- 入力に VREF- を選択します。その他の入力選択ビットは全て使用しません。

図 16-32: 1 チャンネルを DMA 割り込み 1 回あたり 16 回変換する

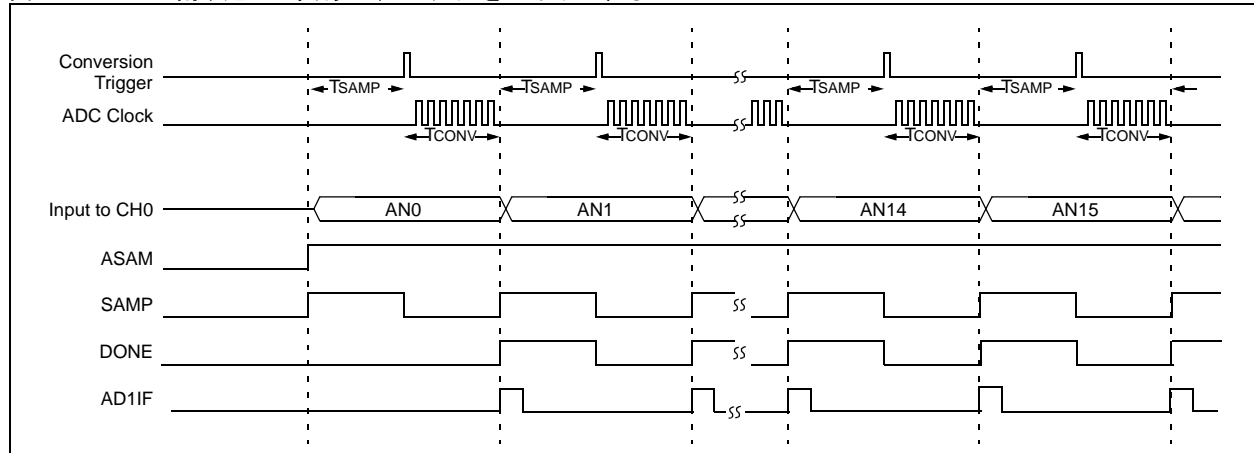


16.11.2 全アナログ入力をスキャンしながら A/D 変換する

図 16-33 と表 16-21 に、利用可能な全アナログ入力チャンネルを 1 つの S&H チャンネル (CH0) でサンプリング / 変換する場合のセットアップ例を示します。ADC 制御レジスタ 2 (ADxCON2<10>) のスキャン入力選択ビット (CSCNA) をセットする事により、CH0 正極性入力への ADC 入力をスキャンします。その他の条件は 16.10.1 「1 チャンネルを複数回サンプリング / 変換する」と同じです。

まず、CH0 で AN0 入力がサンプリング / 変換されます。結果はユーザが定義した DMA バッファに保存されます。次に AN1 入力をサンプリング / 変換します。この入力スキャンプロセスは、バッファがフルまでに 16 回繰り返されます。その後 DMA モジュールは割り込みを生成します。以上のプロセスを繰り返し実行します。

図 16-33: DMA 割り込み 1 回あたり 16 入力をスキャンする



16.11.3 MUXA/MUXB 交互入力選択を使用する

図 16-34 と表 16-22 に、MUXA と MUXB に割り当てた入力を交互にサンプリングする例を示します。この例では、2 チャンネルの同時サンプリングを有効にしています。ALTS ビット (ADCxCON2<0>) をセットする事により、交互入力選択を有効にします。最初のサンプリングには CH0SA、CH0NA、CH123SA、CH123NA ビットで指定した MUXA 入力を使用します。次のサンプリングには CH0SB、CH0NB、CH123SB、CH123NB ビットで指定した MUXB 入力を使用します。この例では、MUXB 入力選択で 1 つの S&H (CH1) に 2 つのアナログ入力を差動入力として選択する事により (AN3-AN9) をサンプリングします。

4 つの S&H チャンネルを交互入力選択なしで使用した場合の変換数は、この例のように 2 チャンネルを交互入力選択して使用した場合の変換数と同じです。しかし CH1、CH2、CH3 チャンネルのアナログ入力選択自由度が限られているため、この例の方が 4 チャンネルを使用するよりも柔軟に入力を選択できます。

図 16-34: 交互入力選択を使用して 2 x 2 入力を変換する

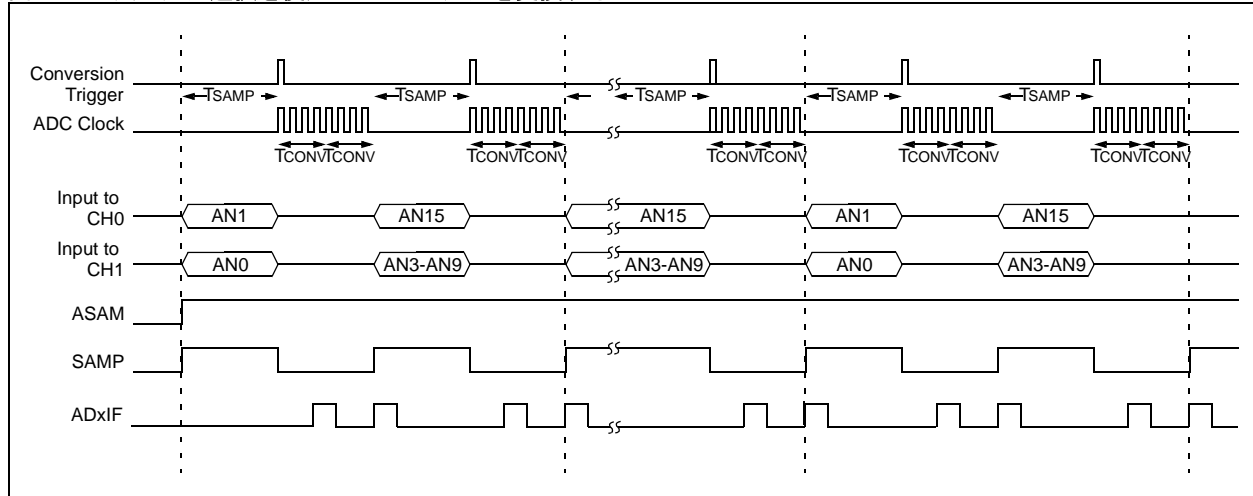


表 16-22: 交互入力選択を使用して 2 x 2 入力を変換する

制御ビット		動作シーケンス	
シーケンス選択		DMA 割り込み	
SMPI<3:0> = 0001 サンプリング / 変換動作 2 回ごとに DMA アドレスをインクリメントする		MUXA 入力をサンプル : AN1→CH0, AN0→CH1	
CHPS<1:0> = 01 チャンネル CH0/CH1 をサンプリングする		CH0 を変換する	
SIMSAM = 1 全てのチャンネルを同時にサンプリング		CH1 を変換する	
ADDMABM = 1 変換順に DMA バッファに書き込む		MUXB 入力をサンプル : AN15→CH0, (AN3-AN9)→CH1	
ALTS = 1 MUXA/MUXB 入力の交互選択		CH0 を変換する	
MUXA 入力選択		CH1 を変換する	
CH0SA<3:0> = 0001 CH0+ 入力に AN1 を選択する		MUXA 入力をサンプル : AN1 ® CH0, AN0 → CH1	
CH0NA = 0 CH0- 入力に VREF- を選択する		CH0 を変換する	
CSCNA = 0 入力をスキャンしない		CH1 を変換する	
CSSL<15:0> = n/a スキャン入力選択を使用しない		MUXB 入力をサンプル : AN15→CH0, (AN3-AN9)→CH1	
CH123SA = 0 CH1+ = AN0, CH2+ = AN1, CH3+ = AN2		CH0 を変換する	
CH123NA<1:0> = 0x CH1-、CH2-、CH3- = VREF-		CH1 を変換する	
MUXB 入力選択		MUXA 入力をサンプル : AN1 → CH0, AN0 → CH1	
CH0SB<3:0> = 1111 CH0+ 入力に AN15 を選択する		CH0 を変換する	
CH0NB = 0 CH0- 入力に VREF- を選択する		CH1 を変換する	
CH123SB = 1 CH1+ = AN3, CH2+ = AN4, CH3+ = AN5		MUXB 入力をサンプル : AN15→CH0, (AN3-AN9)→CH1	
CH123NB<1:0> = 11 CH1- = AN9, CH2- = AN10, CH3- = AN11		CH0 を変換する	
		CH1 を変換する	
		DMA 割り込み	
		以上を繰り返す	

初回の DMA 割り込み時の
DMA バッファの状態

AN1 サンプル 1
AN0 サンプル 1
AN15 サンプル 1
(AN3-AN9) サンプル 1

2 回目の DMA 割り込み時の
DMA バッファの状態

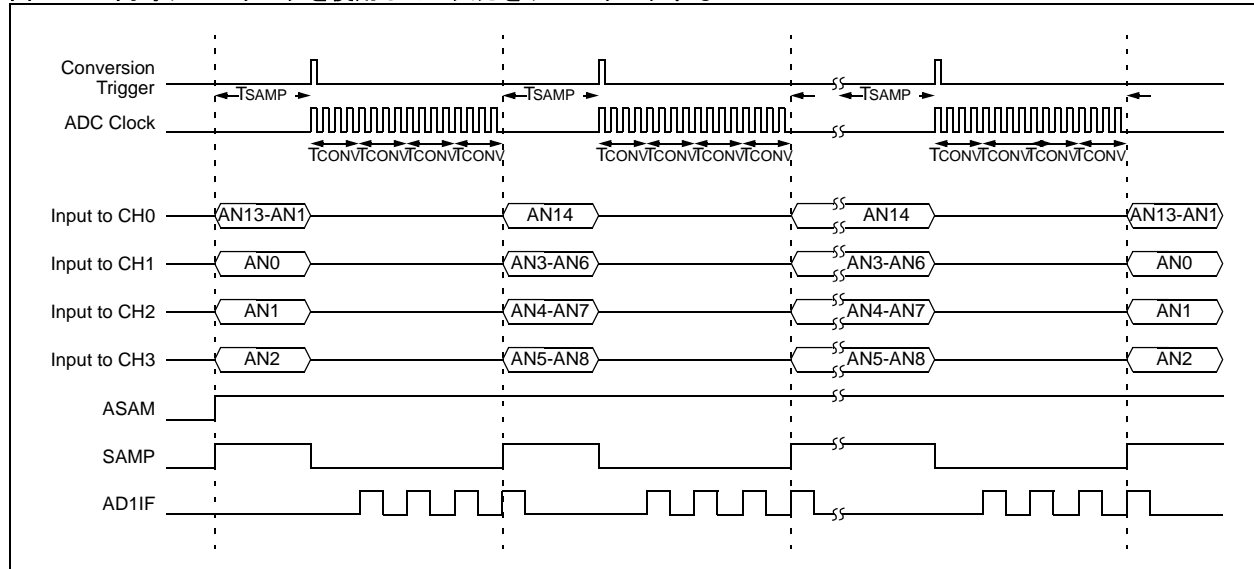
AN1 サンプル 3
AN0 サンプル 3
AN15 サンプル 3
(AN3-AN9) サンプル 3

16.11.4 同時サンプリングを使用して 8 入力をサンプリングする

この例と次の例の設定は似ていますが、この例では同時サンプリング (SIMSAM = 1) を使用するのに対して次の例では逐次サンプリング (SIMSAM = 0) を使用するという点で異なります。両例では、交互入力選択を使用し、S&H に差動入力指定します。

図 16-35 と表 16-23 に、同時サンプリングの例を示します。同時サンプリングで複数チャンネルを変換する場合、ADC モジュールは全てのチャンネルをサンプリングしてから変換シーケンスを実行します。この例では ASAM ビットをセットするため、変換完了後自動的にサンプリングが開始されます。

図 16-35: 同時サンプリングを使用して 8 入力をサンプリングする



dsPIC33F/PIC24H ファミリ リファレンス マニュアル

表 16-23: 同時サンプリングを使用して 8 入力をサンプリングする
制御ビット

シーケンス選択	
SMPI<3:0> = 0001	サンプリング / 変換動作 2 回ごとに DMA アドレスをインクリメントする
CHPS<1:0> = 1X	CH0/CH1/CH2/CH3 をサンプリングする
SIMSAM = 1	全てのチャンネルを同時にサンプリング
ADDMABM = 0	変換順に DMA バッファに書き込む
ALTS = 1	MUXA/MUXB 入力の交互選択

MUXA 入力選択	
CH0SA<3:0> = 1101	CH0+ 入力に AN13 を選択する
CH0NA = 1	CH0- 入力に AN1 を選択する
CSCNA = 0	入カスキャンしない
CSSL<15:0> = n/a	スキャン入力選択を使用しない
CH123SA = 0	CH1+ = AN0、CH2+ = AN1、CH3+ = AN2
CH123NA<1:0> = 0X	CH1-、CH2-、CH3- = VREF-

MUXB 入力選択	
CH0SB<3:0> = 1110	CH0+ 入力に AN14 を選択する
CH0NB = 0	CH0- 入力に VREF- を選択する
CH123SB = 1	CH1+ = AN3、CH2+ = AN4、CH3+ = AN5
CH123NB<1:0> = 10	CH1- = AN6、CH2- = AN7、CH3- = AN8

初回 DMA 割り込み時の
DMA バッファの状態

(AN13-AN1) サンプル 1
AN0 サンプル 1
AN1 サンプル 1
AN2 サンプル 1
AN14 サンプル 1
(AN3-AN6) サンプル 1
(AN4-AN7) サンプル 1
(AN5-AN8) サンプル 1
(AN13-AN1) サンプル 2
AN0 サンプル 2
AN1 サンプル 2
AN2 サンプル 2
AN14 サンプル 2
(AN3-AN6) サンプル 2
(AN4-AN7) サンプル 2
(AN5-AN8) サンプル 2

動作シーケンス

MUXA 入力をサンプリングする: (AN13-AN1)→CH0, AN0→CH1, AN1→CH2, AN2→CH3	
CH0 を変換する	
CH1 を変換する	
CH2 を変換する	
CH3 を変換する	
MUXB 入力をサンプリングする: AN14 → CH0、 (AN3-AN6)→CH1, (AN4-AN7)→CH2, (AN5-AN8)→CH3	
CH0 を変換する	
CH1 を変換する	
CH2 を変換する	
CH3 を変換する	
MUXA 入力をサンプリングする: (AN13-AN1)→CH0, AN0→CH1, AN1→CH2, AN2→CH3	
CH0 を変換する	
CH1 を変換する	
CH2 を変換する	
CH3 を変換する	
MUXB 入力をサンプリングする: AN14 @ CH0、 (AN3-AN6)→CH1, (AN4-AN7)→CH2, (AN5-AN8)→CH3	
CH0 を変換する	
CH1 を変換する	
CH2 を変換する	
CH3 を変換する	
DMA 割り込み	
以上を繰り返す	

2 回目 DMA 割り込み時の
DMA バッファの状態

(AN13-AN1) サンプル 3
AN0 サンプル 3
AN1 サンプル 3
AN2 サンプル 3
AN14 サンプル 3
(AN3-AN6) サンプル 3
(AN4-AN7) サンプル 3
(AN5-AN8) サンプル 3
(AN13-AN1) サンプル 4
AN0 サンプル 4
AN1 サンプル 4
AN2 サンプル 4
AN14 サンプル 4
(AN3-AN6) サンプル 4
(AN4-AN7) サンプル 4
(AN5-AN8) サンプル 4

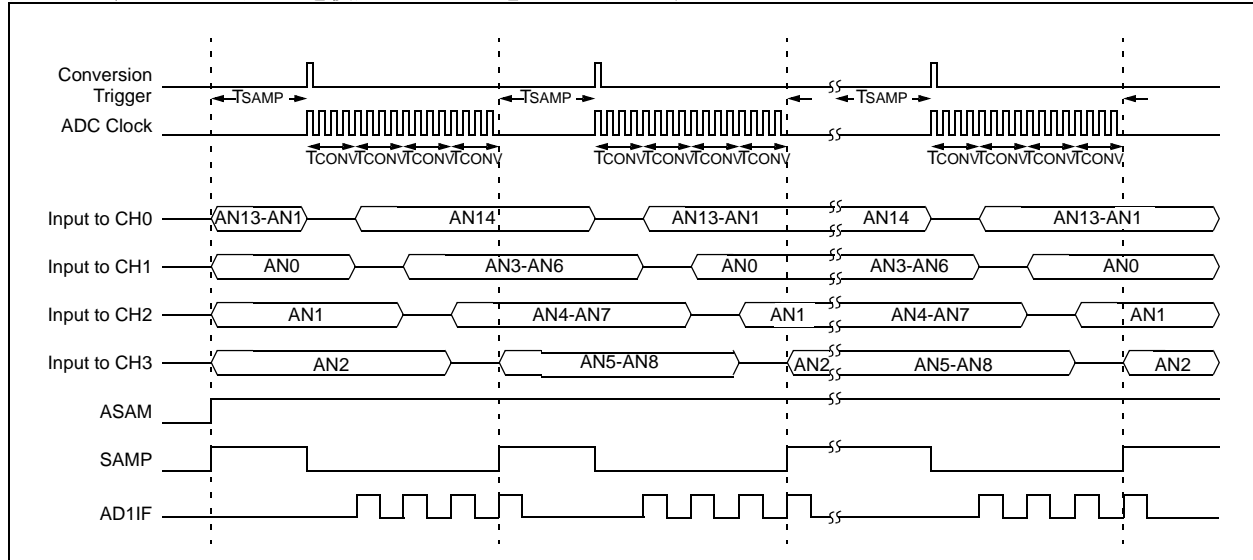
16.11.5 逐次サンプリングを使用して 8 入力をサンプリングする

図 16-36 と表 16-24 に、逐次サンプリングの例を示します。逐次サンプリングで複数チャンネルを変換する場合、ADC モジュールは可能な限り早期に 1 チャンネルのサンプリングを開始し、続いてそのチャンネルの変換を実行します。この例では ASAM ビットをセットするため、各チャンネルのサンプリングは、そのチャンネルの変換が完了した後に自動的に開始されます。

ASAM ビットをクリアした場合、変換が完了しても SAMP ビットがセットされるまでサンプリングは開始されません。

複数チャンネルを使用する場合、逐次サンプリングでは他チャンネルで変換実行中に 1 チャンネルをサンプリングできるため、同時サンプリングよりも長いサンプリング時間が得られます。

図 16-36: 逐次サンプリングを使用して 8 入力をサンプリングする



dsPIC33F/PIC24H ファミリ リファレンス マニュアル

表 16-24: 逐次サンプリングを使用して 8 入力をサンプリングする
制御ビット

シーケンス選択	
SMPI<3:0> = 0001	サンプリング / 変換動作 2 回ごとに DMA アドレスをインクリメントする
CHPS<1:0> = 1X	CH0/CH1/CH2/CH3 をサンプリングする
SIMSAM = 0	全てのチャンネルを逐次サンプリングする
ADDMABM = 1	変換順に DMA バッファに書き込む
ALTS = 1	MUXA/MUXB 入力の交互選択

MUXA 入力選択	
CH0SA<4:0> = 01101	CH0+ 入力に AN13 を選択する
CH0NA = 1	CH0- 入力に AN1 を選択する
CSCNA = 0	入カスキャンしない
CSSL<15:0> = n/a	スキャン入力選択を使用しない
CH123SA = 0	CH1+ = AN0、CH2+ = AN1、CH3+ = AN2
CH123NA<1:0> = 0X	CH1-、CH2-、CH3- = VREF-

MUXB 入力選択	
CH0SB<4:0> = 01110	CH0+ 入力に AN14 を選択する
CH0NB = 0	CH0- 入力に VREF- を選択する
CH123SB = 1	CH1+ = AN3、CH2+ = AN4、CH3+ = AN5
CH123NB<1:0> = 10	CH1- = AN6、CH2- = AN7、CH3- = AN8

初回の DMA 割り込み時の
DMA バッファの状態

(AN13-AN1) サンプル 1
AN0 サンプル 1
AN1 サンプル 1
AN2 サンプル 1
AN14 サンプル 1
(AN3-AN6) サンプル 1
(AN4-AN7) サンプル 1
(AN5-AN8) サンプル 1
(AN13-AN1) サンプル 2
AN0 サンプル 2
AN1 サンプル 2
AN2 サンプル 2
AN14 サンプル 2
(AN3-AN6) サンプル 2
(AN4-AN7) サンプル 2
(AN5-AN8) サンプル 2

動作シーケンス

サンプリング : (AN13-AN1) @ CH0
CH0 を変換する
サンプリング : AN0 → CH1
CH1 を変換する
サンプリング : AN1 → CH2
CH2 を変換する
サンプリング : AN2 → CH3
CH3 を変換する
サンプリング : AN14 → CH0
CH0 を変換する
サンプリング : (AN3-AN6) → CH1
CH1 を変換する
サンプリング : (AN4-AN7) → CH2
CH2 を変換する
サンプリング : (AN5-AN8) → CH3
CH3 を変換する
サンプリング : (AN13-AN1) → CH0
CH0 を変換する
サンプリング : AN0 → CH1
CH1 を変換する
サンプリング : AN1 → CH2
CH2 を変換する
サンプリング : AN2 → CH3
CH3 を変換する
サンプリング : AN14 → CH0
CH0 を変換する
サンプリング : (AN3-AN6) → CH1
CH1 を変換する
サンプリング : (AN4-AN7) → CH2
CH2 を変換する
サンプリング : (AN5-AN8) → CH3
CH3 を変換する
DMA 割り込み
以上を繰り返す

2 回目の DMA 割り込み時の
DMA バッファの状態

(AN13-AN1) サンプル 3
AN0 サンプル 3
AN1 サンプル 3
AN2 サンプル 3
AN14 サンプル 3
(AN3-AN6) サンプル 3
(AN4-AN7) サンプル 3
(AN5-AN8) サンプル 3
(AN13-AN1) サンプル 4
AN0 サンプル 4
AN1 サンプル 4
AN2 サンプル 4
AN14 サンプル 4
(AN3-AN6) サンプル 4
(AN4-AN7) サンプル 4
(AN5-AN8) サンプル 4

16.13 ADC 結果バッファの読み出し

RAM は 10 ビットまたは 12 ビット幅ですが、バッファ読み出し時にデータは選択可能な 4 種類のフォーマットのいずれかに自動的にフォーマットされます。このフォーマットの選択には FORM<1:0> ビット (ADCON1<9:8>) を使用します。どのデータ フォーマットを選択しても、フォーマット ハードウェアはデータバスに 16 ビット結果を供給します。図 16-39 と図 16-40 に、FORM<1:0> 制御ビットで選択可能なデータ出力フォーマットを示します。

図 16-39: A/D 出力データフォーマット (10 ビットモード)

RAM の内容：	<table><tr><td>d09</td><td>d08</td><td>d07</td><td>d06</td><td>d05</td><td>d04</td><td>d03</td><td>d02</td><td>d01</td><td>d00</td></tr></table>										d09	d08	d07	d06	d05	d04	d03	d02	d01	d00												
d09	d08	d07	d06	d05	d04	d03	d02	d01	d00																							
バスへの読み出し：																																
符号なし整数	<table><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>d09</td><td>d08</td><td>d07</td><td>d06</td><td>d05</td><td>d04</td><td>d03</td><td>d02</td><td>d01</td><td>d00</td></tr></table>																0	0	0	0	0	0	d09	d08	d07	d06	d05	d04	d03	d02	d01	d00
0	0	0	0	0	0	d09	d08	d07	d06	d05	d04	d03	d02	d01	d00																	
符号付き整数	<table><tr><td>$\overline{d09}$</td><td>$\overline{d09}$</td><td>$\overline{d09}$</td><td>$\overline{d09}$</td><td>$\overline{d09}$</td><td>$\overline{d09}$</td><td>$\overline{d09}$</td><td>d08</td><td>d07</td><td>d06</td><td>d05</td><td>d04</td><td>d03</td><td>d02</td><td>d01</td><td>d00</td></tr></table>																$\overline{d09}$	$\overline{d09}$	$\overline{d09}$	$\overline{d09}$	$\overline{d09}$	$\overline{d09}$	$\overline{d09}$	d08	d07	d06	d05	d04	d03	d02	d01	d00
$\overline{d09}$	$\overline{d09}$	$\overline{d09}$	$\overline{d09}$	$\overline{d09}$	$\overline{d09}$	$\overline{d09}$	d08	d07	d06	d05	d04	d03	d02	d01	d00																	
符号なし小数	<table><tr><td>d09</td><td>d08</td><td>d07</td><td>d06</td><td>d05</td><td>d04</td><td>d03</td><td>d02</td><td>d01</td><td>d00</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>																d09	d08	d07	d06	d05	d04	d03	d02	d01	d00	0	0	0	0	0	0
d09	d08	d07	d06	d05	d04	d03	d02	d01	d00	0	0	0	0	0	0																	
符号付き小数 (1, 15)	<table><tr><td>$\overline{d09}$</td><td>d08</td><td>d07</td><td>d06</td><td>d05</td><td>d04</td><td>d03</td><td>d02</td><td>d01</td><td>d00</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>																$\overline{d09}$	d08	d07	d06	d05	d04	d03	d02	d01	d00	0	0	0	0	0	0
$\overline{d09}$	d08	d07	d06	d05	d04	d03	d02	d01	d00	0	0	0	0	0	0																	

図 16-40: A/D 出力データフォーマット (12 ビットモード)

RAM の内容：	<table><tr><td>d11</td><td>d10</td><td>d09</td><td>d08</td><td>d07</td><td>d06</td><td>d05</td><td>d04</td><td>d03</td><td>d02</td><td>d01</td><td>d00</td></tr></table>															d11	d10	d09	d08	d07	d06	d05	d04	d03	d02	d01	d00							
d11	d10	d09	d08	d07	d06	d05	d04	d03	d02	d01	d00																							
バスへの読み出し：																																		
符号なし整数	<table><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>d11</td><td>d10</td><td>d09</td><td>d08</td><td>d07</td><td>d06</td><td>d05</td><td>d04</td><td>d03</td><td>d02</td><td>d01</td><td>d00</td></tr></table>																0	0	0	0	d11	d10	d09	d08	d07	d06	d05	d04	d03	d02	d01	d00		
0	0	0	0	d11	d10	d09	d08	d07	d06	d05	d04	d03	d02	d01	d00																			
符号付き整数	<table><tr><td>$\overline{d11}$</td><td>$\overline{d11}$</td><td>$\overline{d11}$</td><td>$\overline{d11}$</td><td>$\overline{d11}$</td><td>d10</td><td>d09</td><td>d08</td><td>d07</td><td>d06</td><td>d05</td><td>d04</td><td>d03</td><td>d02</td><td>d01</td><td>d00</td></tr></table>																$\overline{d11}$	$\overline{d11}$	$\overline{d11}$	$\overline{d11}$	$\overline{d11}$	d10	d09	d08	d07	d06	d05	d04	d03	d02	d01	d00		
$\overline{d11}$	$\overline{d11}$	$\overline{d11}$	$\overline{d11}$	$\overline{d11}$	d10	d09	d08	d07	d06	d05	d04	d03	d02	d01	d00																			
符号なし小数	<table><tr><td>d11</td><td>d10</td><td>d09</td><td>d08</td><td>d07</td><td>d06</td><td>d05</td><td>d04</td><td>d03</td><td>d02</td><td>d01</td><td>d00</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>																d11	d10	d09	d08	d07	d06	d05	d04	d03	d02	d01	d00	0	0	0	0		
d11	d10	d09	d08	d07	d06	d05	d04	d03	d02	d01	d00	0	0	0	0																			
符号付き小数 (1, 15)	<table><tr><td>$\overline{d11}$</td><td>d10</td><td>d09</td><td>d08</td><td>d07</td><td>d06</td><td>d05</td><td>d04</td><td>d03</td><td>d02</td><td>d01</td><td>d00</td><td>d01</td><td>d00</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>																$\overline{d11}$	d10	d09	d08	d07	d06	d05	d04	d03	d02	d01	d00	d01	d00	0	0	0	0
$\overline{d11}$	d10	d09	d08	d07	d06	d05	d04	d03	d02	d01	d00	d01	d00	0	0	0	0																	

表 16-25 と表 16-26 に、それぞれ 10 ビットモードと 12 ビットモードでの各種結果コードの等価値を示します。

表 16-25: 各種結果コードの等価値 (10 ビットモード)

VIN/VREF	10 ビット 出力コード	16 ビット符号なし 整数フォーマット	16 ビット符号付き 整数フォーマット	16 ビット符号なし 小数フォーマット	16 ビット符号付き 小数フォーマット
1023/1024	11 1111 1111	0000 0011 1111 1111 = 1023	0000 0001 1111 1111 = 511	1111 1111 1100 0000 = 0.999	0111 1111 1100 0000 = 0.99804
1022/1024	11 1111 1110	0000 0011 1111 1110 = 1022	0000 0001 1111 1110 = 510	1111 1111 1000 0000 = 0.998	0111 1111 1000 0000 = 0.499609
⋮					
513/1024	10 0000 0001	0000 0010 0000 0001 = 513	0000 0000 0000 0001 = 1	1000 0000 0100 0000 = 0.501	0000 0000 0100 0000 = 0.00195
512/1024	10 0000 0000	0000 0010 0000 0000 = 512	0000 0000 0000 0000 = 0	1000 0000 0000 0000 = 0.500	0000 0000 0000 0000 = 0
511/1024	01 1111 1111	0000 0001 1111 1111 = 511	1111 1111 1111 1111 = -1	0111 1111 1100 0000 = .499	1111 1111 1100 0000 = -0.00195
⋮					
1/1024	00 0000 0001	0000 0000 0000 0001 = 1	1111 1110 0000 0001 = -511	0000 0000 0100 0000 = 0.001	1000 0000 0100 0000 = -0.99804
0/1024	00 0000 0000	0000 0000 0000 0000 = 0	1111 1110 0000 0000 = -512	0000 0000 0000 0000 = 0	1000 0000 0000 0000 = -1

表 16-26: 各種結果コードの等価値 (12 ビットモード)

VIN/VREF	12 ビット 出力コード	16 ビット符号なし 整数フォーマット	16 ビット符号付き 整数フォーマット	16 ビット符号なし 小数フォーマット	16 ビット符号付き 小数フォーマット
4095/4096	1111 1111 1111	0000 1111 1111 1111 = 4095	0000 0111 1111 1111 = 2047	1111 1111 1111 0000 = 0.9998	0111 1111 1111 0000 = 0.9995
4094/4096	1111 1111 1110	0000 1111 1111 1110 = 4094	0000 0111 1111 1110 = 2046	1111 1111 1110 0000 = 0.9995	0111 1111 1110 0000 = 0.9990
⋮					
2049/4096	1000 0000 0001	0000 1000 0000 0001 = 2049	0000 0000 0000 0001 = 1	1000 0000 0001 0000 = 0.5002	0000 0000 0001 0000 = 0.0005
2048/4096	1000 0000 0000	0000 1000 0000 0000 = 2048	0000 0000 0000 0000 = 0	1000 0000 0000 0000 = 0.500	0000 0000 0000 0000 = 0.000
2047/4096	0111 1111 1111	0000 0111 1111 1111 = 2047	1111 1111 1111 1111 = -1	0111 1111 1111 0000 = 0.4998	1111 1111 1111 0000 = -0.0005
⋮					
1/4096	0000 0000 0001	0000 0000 0000 0001 = 1	1111 1000 0000 0001 = -2047	0000 0000 0001 0000 = 0.0002	1000 0000 0001 0000 = -0.9995
0/4096	0000 0000 0000	0000 0000 0000 0000 = 0	1111 1000 0000 0000 = -2048	0000 0000 0000 0000 = 0	1000 0000 0000 0000 = -1.000

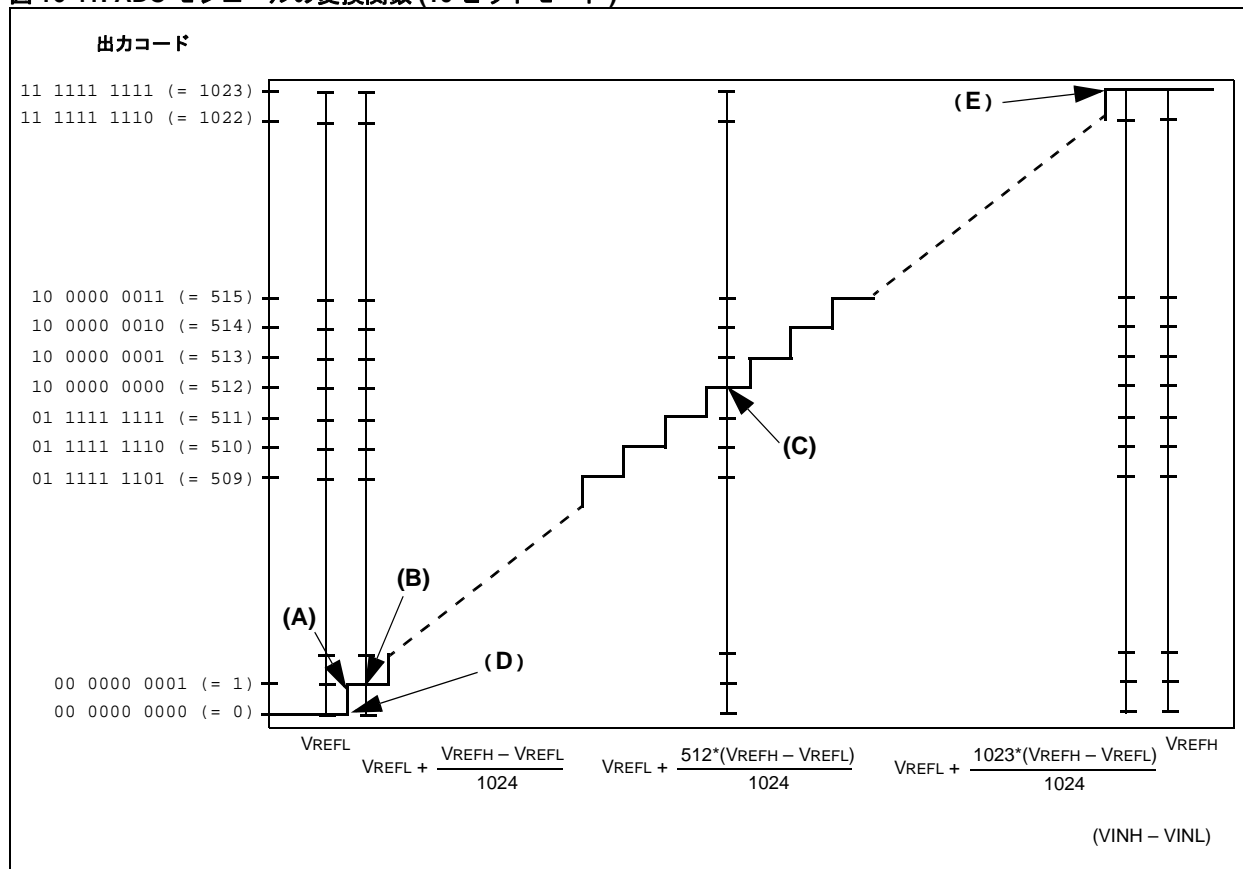
16.14 変換関数

16.14.1 10 ビットモード

ADC モジュールの理想的変換関数を図 16-41 に示します。入力差動電圧 ($V_{INH} - V_{INL}$) はリファレンス電圧 ($V_{REFH} - V_{REFL}$) と比較されます。

- 最初のコード変化 (A) は、入力電圧が $(V_{REFH} - V_{REFL}/2048)$ または 0.5 LSb の時に発生します。
- コード 00 0000 0001 は $(V_{REFH} - V_{REFL}/1024)$ または 1.0 LSb を中心値とします (B)。
- コード 10 0000 0000 は $(512 \cdot (V_{REFH} - V_{REFL})/1024)$ を中心値とします (C)。
- $(1 \cdot (V_{REFH} - V_{REFL})/2048)$ より低い入力電圧はコード 00 0000 0000 に変換されます (D)。
- $(2045 \cdot (V_{REFH} - V_{REFL})/2048)$ よりも高い入力電圧はコード 11 1111 1111 に変換されます (E)。

図 16-41: ADC モジュールの変換関数 (10 ビットモード)

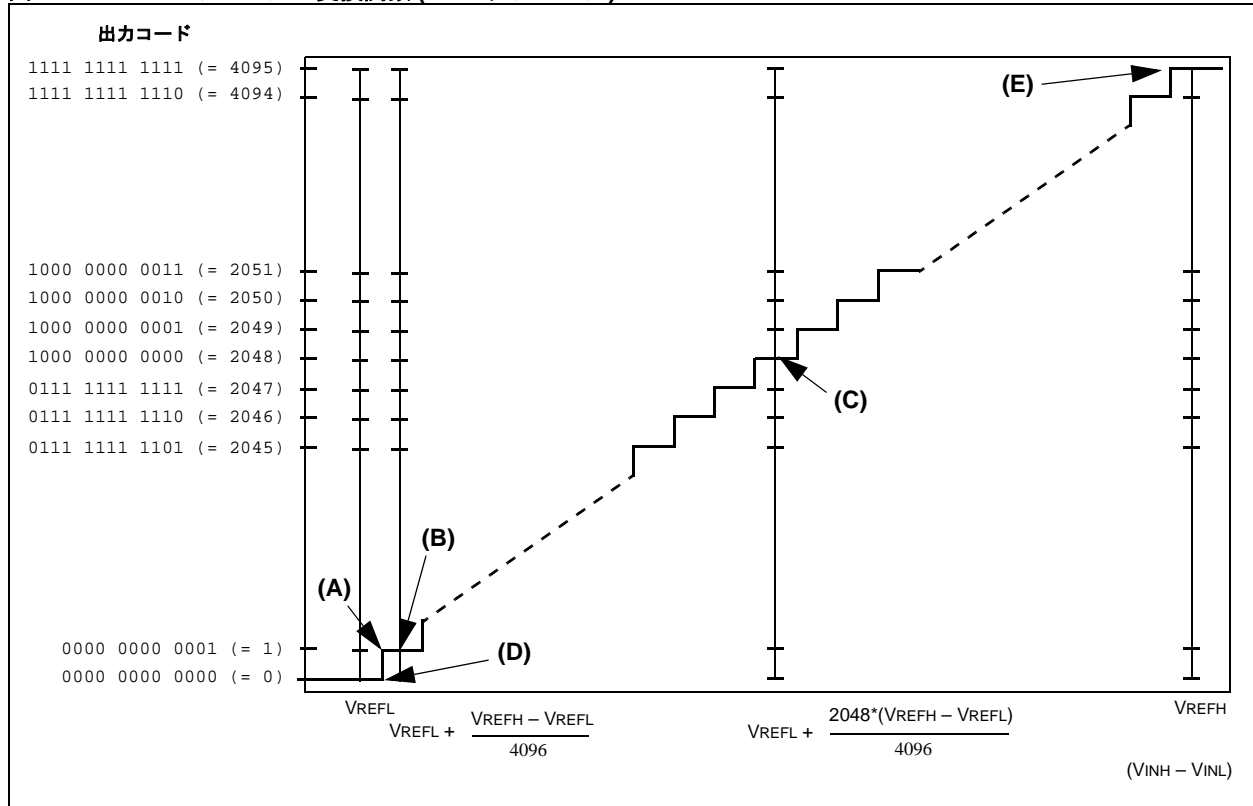


16.14.2 12 ビットモード

ADC モジュールの理想的変換関数を図 16-42 に示します。入力差動電圧 ($V_{INH} - V_{INL}$) はリファレンス電圧 ($V_{REFH} - V_{REFL}$) と比較されます。

- 最小のコード変換 (A) は、入力電圧が $(V_{REFH} - V_{REFL}/8192)$ または 0.5 LSb の時に発生します。
- コード 00 0000 0001 は $(V_{REFH} - V_{REFL}/4096)$ または 1.0 LSb を中心値とします (B)。
- コード 10 0000 0000 は $(2048 * (V_{REFH} - V_{REFL})/4096)$ を中心値とします (C)。
- $(1 * (V_{REFH} - V_{REFL})/8192)$ より低い入力電圧はコード 00 0000 0000 に変換されます (D)。
- $(8192 * (V_{REFH} - V_{REFL})/8192)$ よりも高い入力電圧はコード 11 1111 1111 に変換されます (E)。

図 16-42: ADC モジュールの変換関数 (12 ビットモード)



16.15 ADC の精度と誤差

INL、DNL、ゲインエラー、オフセットエラーについては各デバイス データシートの「電気的特性」を参照してください。ADC 精度に関連する文書の一覧 (16.21「関連アプリケーションノート」) もご覧ください。

16.16 接続に関する注意事項

アナログ入力は ESD 保護のために V_{DD} と V_{SS} に対するダイオードを備えます。このため、アナログ入力電圧は $V_{DD} \sim V_{SS}$ のレンジ内である必要があります。入力電圧がこのレンジをいずれかの方向に 0.3 V よりも大きく超えると一方のダイオードが順バイアスとなり、入力電流が仕様値を超えるとデバイスを損傷する可能性があります。

入力信号のアンチエイリアシング用に外付け RC フィルタを追加する場合、サンプリング時間要件を満たせるように抵抗部品を選択する必要があります。アナログ入力ピンに外付け部品 (コンデンサ、ツェナーダイオード等) を高インピーダンスで接続する場合、それらの部品によるピンリーク電流を極めて低く抑える必要があります。

16.17 スリープおよびアイドルモード時の動作

スリープおよびアイドルモードでは CPU、バス、その他の周辺モジュールのデジタル動作が最小限になるため、変換ノイズを最小限に抑える事ができます。

16.17.1 CPU スリープモード時の ADC 動作 (RC A/D クロックを使用しない場合)

デバイスがスリープモードに移行すると、ADC モジュールへの全てのクロック源は停止して論理状態「0」を維持します。

ADC が内部 RC クロック ジェネレータからクロック供給を受けていない場合、変換動作の途中でスリープモードへの移行が発生すると、その変換は中止されます。中止された変換は、スリープモード終了時に再開されません。

デバイスがスリープモードへ移行またはスリープモードから復帰しても、レジスタの内容は影響を受けません。

16.17.2 CPU スリープモード時の ADC 動作 (RC A/D クロックを使用する場合)

内部 A/D RC オシレータを A/D クロック源として設定した (ADRC = 1) 場合、ADC モジュールはスリープモード時にも動作可能です。これにより、変換時のデジタル スイッチング ノイズを除去できます。変換が完了すると DONE ビットがセットされ、結果が ADC 結果バッファ (ADCxBUF0) に書き込まれます。

ADC 割り込みを有効 (ADxIE = 1) にした場合、デバイスは ADC 割り込み発生時にスリープモードからウェイクアップします。ADC 割り込みの優先度が CPU 割り込み優先度よりも高い場合、プログラム実行は ADC 割り込みサービス ルーチン (ISR) で再開されます。これ以外の場合、プログラム実行はデバイスをスリープモードへ移行させた PWRSAV 命令の直後の命令から再開されます。

ADC 割り込みを無効にした場合、ADON ビットはセットされたままですが、ADC モジュールは動作を停止します。

ADC モジュールの動作に対するデジタルノイズの影響を最小限に抑えるには、スリープモード時に A/D 変換を実行できるように変換トリガ源を選択する必要があります。SSRC<2:0> を「111」に設定する事により、スリープモード時のサンプリング / 変換に自動変換トリガオプションを使用できます。自動変換オプションを使用するには、PWRSAV 命令より前の命令で ADON ビットをセットする必要があります。

Note: スリープモード時に ADC モジュール動作させるには、ADC クロック源を RC に設定する (ADRC = 1) 必要があります。

16.17.3 CPU アイドルモード時の ADC 動作

アイドルモード時に ADC モジュールの動作を継続するかどうかは、ADSIDL ビット (ADxCON1<13>) で選択します。ADSIDL = 0 の場合、デバイスがアイドルモードに移行しても ADC モジュールは通常動作を継続します。ADC 割り込みを有効 (ADxIE = 1) にした場合、デバイスは ADC 割り込み発生時にアイドルモードからウェイクアップします。ADC 割り込みの優先度が CPU 割り込み優先度よりも高い場合、プログラム実行は ADC 割り込みサービス ルーチンで再開されます。これ以外の場合、プログラム実行はデバイスをアイドルモードへ移行させた PWRSAV 命令の直後の命令から再開されます。

ADSIDL = 1 の場合、ADC モジュールはアイドルモード時に停止します。変換動作の途中でデバイスがアイドルモードに移行した場合、その変換は中止されます。中止された変換は、アイドルモード終了時に再開されません。

16.18 リセットの影響

デバイスリセットが発生すると、全てのレジスタはリセット状態に戻されます。これにより ADC モジュールは停止し、実行中の変換は全て中止されます。アナログ入力と多重化されている全てのピンは、アナログ入力として設定されます。対応する TRIS ビットはセットされます。

ADCxBUF0 レジスタの値は、パワーオン リセット (POR) 時に初期化されず、未知のデータを格納します。

16.19 特殊機能レジスタ

dsPIC33F/PIC24H のアナログ / デジタル コンバータ (ADC) モジュールに関連するレジスタの要約を表 16-27 に示します。

表 16-27: ADC 関連のレジスタマップ

レジスタ名	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	全リ セット	
ADC1BUF0	ADC1 データバッファ 0																uuuu	
ADC1BUF1	ADC1 データバッファ 1																uuuu	
ADC1BUF2	ADC1 データバッファ 2																uuuu	
ADC1BUF3	ADC1 データバッファ 3																uuuu	
ADC1BUF4	ADC1 データバッファ 4																uuuu	
ADC1BUF5	ADC1 データバッファ 5																uuuu	
ADC1BUF6	ADC1 データバッファ 6																uuuu	
ADC1BUF7	ADC1 データバッファ 7																uuuu	
ADC1BUF8	ADC1 データバッファ 8																uuuu	
ADC1BUF9	ADC1 データバッファ 9																uuuu	
ADC1BUFA	ADC1 データバッファ 10																uuuu	
ADC1BUFB	ADC1 データバッファ 11																uuuu	
ADC1BUFC	ADC1 データバッファ 12																uuuu	
ADC1BUFD	ADC1 データバッファ 13																uuuu	
ADC1BUFE	ADC1 データバッファ 14																uuuu	
ADC1BUFF	ADC1 データバッファ 15																uuuu	
ADxCON1	ADON	—	ADSIDL	ADDMABM ⁽¹⁾	—	AD12B ⁽⁴⁾	FORM<1:0>		SSRC<2:0>			—	SIMSAM	ASAM	SAMP	DONE ⁽²⁾	0000	
ADxCON2	VCFG<2:0>			—	—	CSCNA	CHPS<1:0>		BUFS	—	SMPI<3:0>				BUFM	ALTS	0000	
ADxCON3	ADRC	—	—	SAMC<4:0>				ADCS<7:0>										0000
ADxCHS123	—	—	—	—	—	CH123NB<1:0>		CH123SB	—	—	—	—	—	CH123NA<1:0>		CH123SA	0000	
ADxCHS0	CH0NB	—	—	CH0SB<4:0>					CH0NA	—	—	CH0SA<4:0>					0000	
ADxPCFGH	PCFG31	PCFG30	PCFG29	PCFG28	PCFG27	PCFG26	PCFG25	PCFG24	PCFG23	PCFG22	PCFG21	PCFG20	PCFG19	PCFG18	PCFG17	PCFG16	0000	
ADxPCFGL	PCFG15	PCFG14	PCFG13	PCFG12	PCFG11	PCFG10	PCFG9	PCFG8	PCFG7	PCFG6	PCFG5	PCFG4	PCFG3	PCFG2	PCFG1	PCFG0	0000	
ADxCSSH	CSS31	CSS30	CSS29	CSS28	CSS27	CSS26	CSS25	CSS24	CSS23	CSS22	CSS21	CSS20	CSS19	CSS18	CSS17	CSS16	0000	
ADxCSSL	CSS15	CSS14	CSS13	CSS12	CSS11	CSS10	CSS9	CSS8	CSS7	CSS6	CSS5	CSS4	CSS3	CSS2	CSS1	CSS0	0000	
ADxCON4 ⁽³⁾	—	—	—	—	—	—	—	—	—	—	—	—	—	DMABL<2:0>			0000	

凡例: u = 未実装、x = リセット時に未知の値、— = 未実装、「0」として読み出し、リセット値は 16 進数で表記

- Note**
- 1: DMA なしデバイスはこのビットを備えません。詳細は各デバイスのデータシートを参照してください。
 - 2: DMA 付きデバイスでは、毎回の変換後に割り込みが発生し、DONE ビットがセットされます (DONE ビットは割り込みフラグ (ADxIF) の状態を反映するため)。DMA なしデバイスでは、割り込みの生成は SMPI<3:0> ビット (ADxCON2<5:2>) および CHPS ビット (ADxCON2<9:8>) に基づくため、DONE ビットは毎回の変換後にセットされず、割り込みフラグ (ADxIF) がセットされた時にセットされます。
 - 3: DMA なしデバイスはこのレジスタを備えません。詳細は各デバイスのデータシートを参照してください。
 - 4: 一部のデバイスはこのビットを備えません。詳細は各デバイスのデータシートを参照してください。

16.20 設計のヒント

質問 1: ADC モジュールのシステム性能を最適化する方法を教えてください。

回答: 性能の最適化については、下記の 3 項目を提案します。

1. タイミング仕様を全て満たしている事を確認してください。停止した ADC モジュールを再起動する場合、サンプリングを開始する前に一定遅延時間が経過するまで待機する必要があります。入力チャンネルを変更する場合も、一定遅延時間が経過するまで待機する必要があります。最後に、各ビットの変換時間として選択する TAD が重要です。TAD の選択には ADxCON3 を使用します。この際、各デバイス データシートの「**電気的特性**」で指定されているレンジ内で TAD を選択する必要があります。TAD が短かすぎると、結果が完全に変換される前に変換プロセスが終了する可能性があります。TAD が長すぎる場合、変換中にサンプリング コンデンサの充電電圧が低下する可能性があります。タイミング要件の詳細は各デバイス データシートの「**電気的特性**」を参照してください。
2. しばしばアナログ信号源のインピーダンスが 10 kΩ を超える事があり、このような場合、信号源からサンプリング コンデンサに流れる充電電流が精度に影響する可能性があります。入力信号が高速に変化しない場合、0.1 mF のコンデンサをアナログ入力に付加してみてください。このコンデンサはサンプリング時にアナログ電圧まで充電されて、4.4 pF 内部サンプリング コンデンサの充電に必要な瞬時電流を供給します。
3. A/D 変換を開始する前にデバイスをスリープモードに移行します。スリープモード中に変換を実行するには、RC クロック源を選択する必要があります。この方法では、CPU とその他の周辺モジュールからのデジタルノイズが最小限になるため、A/D 変換の精度が向上します。

質問 2: ADC に関して適当な参考書はありますか。

回答: A/D 変換の参考書としては「*Analog-Digital Conversion Handbook*」、第三版、Prentice Hall 刊 (ISBN 0-13-03-2848-0) をお勧めします。

質問 3: チャンネル数/サンプルとサンプル数/割り込みの組み合わせがバッファのサイズを超えるとバッファはどうなりますか。

回答: そのようなコンフィグレーションはお勧めできません。未知の結果がバッファに格納されます。

16.21 関連アプリケーション ノート

本セクションに関連するアプリケーション ノートの一覧を下に記載します。一部のアプリケーション ノートは dsPIC33F/PIC24H 製品ファミリ向けではありません。ただし概念は共通しており、変更が必要であったり制限事項が存在するものの利用が可能です。アナログ / デジタル コンバータ (ADC) モジュールに関連する最新のアプリケーション ノートは以下の通りです。

タイトル	アプリケーション ノート番号
アナログ / デジタル コンバータ (ADC) の使用	AN546
ディスプレイとキーボードを備えた 4 チャンネルデジタル電圧計	AN557
A/D コンバータの性能仕様について	AN693
センサーレス BLDC コントロール用 dsPIC30F の使用	AN901
dsPIC30F による AC 誘導モータのベクトル制御	AN908
dsPIC30F2010 を使用するセンサ BLDC モータ制御	AN957
dsPIC30F MCU による AC 誘導モータ制御の紹介	AN984

Note: dsPIC33F/PIC24H ファミリ関連のアプリケーション ノートとサンプルコードはマイクロチップ社のウェブサイト (www.microchip.com) でご覧になれます。

16.22 改訂履歴

リビジョン A (2006 年 12 月)

本書の初版

リビジョン B (2010 年 1 月)

このリビジョンでの変更内容は下記の通りです。

Note: 下記の複数文書をまとめて本リビジョンを作成

- dsPIC33F ファミリ リファレンス マニュアル、セクション 16「アナログ / デジタル コンバータ (ADC)」
- dsPIC33F ファミリ リファレンス マニュアル、セクション 28「DMA を使用しないアナログ / デジタル コンバータ (ADC)」
- dsPIC24H ファミリ リファレンス マニュアル、セクション 16「アナログ / デジタル コンバータ (ADC)」
- dsPIC24H ファミリ リファレンス マニュアル、セクション 28「DMA を使用しないアナログ / デジタル コンバータ (ADC)」

本書全体を通して「DMA 付きデバイス」と「DMA なしデバイス」を区別して記載

- 補足文書に関する情報を記載した網掛け注釈ボックスを本セクションの冒頭に追加
- 下記項目を更新
 - 16.1「はじめに」
 - 16.2.1「ADC 結果バッファ」
 - 16.5「ADC 割り込みの生成」
 - 16.6「変換するアナログ入力の選択」
 - 16.7「DMA 付きデバイスにおける変換結果のバッファリングの指定」
 - 16.10「DMA なしデバイスのサンプリング / 変換シーケンス例」
 - 16.15「ADC の精度と誤差」
- SOC トリガ選択テーブル (表 16-2) を更新
- 例 16-1 の後に網掛け注釈ボックスを追加
- 図 16-2「DMA なしデバイスの ADC ブロック図」を追加
- 式 16-1、式 16-4、式 16-5、式 16-6、式 16-7、式 16-9 を追加
- 下記図を更新
 - 図 16-1、表題を「DMA 付きデバイスの ADC ブロック図」に変更
 - 図 16-6
 - 図 16-9
 - 図 16-10
 - 図 16-11
 - 図 16-27
 - 図 16-28
 - 図 16-29
 - 図 16-30
 - 図 16-31
 - 図 16-39
 - 図 16-40
- 下記例を更新
 - 例 16-1
 - 例 16-2
 - 例 16-3

リビジョン B (2010 年 1 月) (続き)

- 下記式を更新
 - 式 16-2
 - 式 16-3
- 下記表を更新
 - 表 16-14
 - 表 16-15
 - 表 16-16
 - 表 16-17
 - 表 16-18
 - 表 16-19
 - 表 16-25
 - 表 16-26
- 下記レジスタに関する注釈を更新
 - ADxCON1: ADCx 制御レジスタ 1 (レジスタ 16-1)
 - ADxCON3: ADCx 制御レジスタ 3 (レジスタ 16-3)
 - ADxCON4: ADCx 制御レジスタ 4 (レジスタ 16-4)
 - ADxCHS0: ADCx 入力チャンネル 0 選択レジスタ (レジスタ 16-5)
 - AD1CSSH: ADC1 入カスキャン選択レジスタ HIGH(レジスタ 16-6)
 - ADxCSSL: ADCx 入カスキャン選択レジスタ LOW (レジスタ 16-7)
 - AD1PCFGH: ADC1 ポート コンフィグレーション レジスタ HIGH (レジスタ 16-8)
 - ADxPCFGL: ADCx ポート コンフィグレーション レジスタ LOW (レジスタ 16-8)
- ADxCON2 レジスタの SMPI ビット値に関する説明を更新 : ADCx 制御レジスタ 2 (レジスタ 16-2)
- 下記項目を新たに追加
 - 16.3.4 「自動サンプリング / 手動変換シーケンス」
 - 16.4.10 「ADC モジュールの停止」
 - 16.4.7 「変換トリガ源」
 - 16.5 「ADC 割り込みの生成」
- 16.8 「サンプリング / 変換動作の制御」を削除
- 16.18 「サンプルコード」を削除
- レジスタマップ テーブル (表 16-27) 内のアドレス列を削除
- 文章および体裁の変更等、本書全体の細部を修正

リビジョン C (2011 年 6 月)

このリビジョンでの変更内容は下記の通りです。

- 16.1「はじめに」の第2段落で、ADC タイプ(10ビットと12ビット)による機能の違いを明記
- 図 16-1 と図 16-2 内のアナログ入力ピン名 (ANx) を更新
- ADCx 制御レジスタ 1 (レジスタ 16-1) で、SSRC<2:0> の 101 および 011 ビット値の定義と Note 2 を更新、Note 3 を AD12B ピンの説明に追加
- ADCx 制御レジスタ 2 (レジスタ 16-2) に VREF+ および VREF- ピンに関する Note 4 を追加
- 16.3.2「変換時間」内の第3段落の後に、12 ビットモードに関する網掛け注釈を追加
- 16.4.1「ADC 動作モードの選択」内の先頭段落の後の網掛け注釈に、12 ビットモードに関する Note 2 を追加
- 16.4.3「電圧リファレンスの選択」内の先頭段落の最後に、VREF+ および VREF- ピンに関する1文を追加
- 表 16-10 と表 16-11 で、アナログ入力 AN12 を AN31 に変更
- 表 16-11 で、アナログ入力「AN0 ~ AN12」を「VREF-、AN1」に変更
- 例 16-4、例 16-5、例 16-7 内の AD1CHS123bits.CH124NA を AD1CHS123bits.CH123NA に変更
- 例 16-6 のタイトルを更新
- ADC 変換クロック (式 16-7) の後に新たな1段落を追加し、同式のタイトルを更新
- 図 16-38 に 12 ビットモードに関する Note 2 を追加
- ADC レジスタマップ (表 16-27) 内の AD12B ビットに Note 4 を追加
- 文章および体裁の変更等、本書全体の細部を修正

NOTE:

マイクロチップ社製デバイスのコード保護機能に関して次の点にご注意ください。

- マイクロチップ社製品は、該当するマイクロチップ社データシートに記載の仕様を満たしています。
- マイクロチップ社では、通常の条件ならびに仕様に従って使用した場合、マイクロチップ社製品のセキュリティレベルは、現在市場に流通している同種製品の中でも最も高度であると考えています。
- しかし、コード保護機能を解除するための不正かつ違法な方法が存在する事もまた事実です。弊社の理解ではこうした手法は、マイクロチップ社データシートにある動作仕様書以外の方法でマイクロチップ社製品を使用する事になります。このような行為は知的所有権の侵害に該当する可能性が非常に高いと言えます。
- マイクロチップ社は、コードの保全性に懸念を抱くお客様と連携し、対応策に取り組んでいきます。
- マイクロチップ社を含む全ての半導体メーカーで、自社のコードのセキュリティを完全に保証できる企業はありません。コード保護機能とは、マイクロチップ社が製品を「解読不能」として保証するものではありません。

コード保護機能は常に進歩しています。マイクロチップ社では、常に製品のコード保護機能の改善に取り組んでいます。マイクロチップ社のコード保護機能の侵害は、デジタル ミレニアム著作権法に違反します。そのような行為によってソフトウェアまたはその他の著作物に不正なアクセスを受けた場合は、デジタル ミレニアム著作権法の定めるところにより損害賠償訴訟を起こす権利があります。

本書に記載されているデバイス アプリケーション等に関する情報は、ユーザの便宜のためにのみ提供されているものであり、更新によって無効とされる事があります。お客様のアプリケーションが仕様を満たす事を保証する責任は、お客様にあります。マイクロチップ社は、明示的、暗黙的、書面、口頭、法定のいずれであるかを問わず、本書に記載されている情報に関して、状態、品質、性能、品性、特定目的への適合性をはじめとする、いかなる類の表明も保証も行いません。マイクロチップ社は、本書の情報およびその使用に起因する一切の責任を否認します。マイクロチップ社の明示的な書面による承認なしに、生命維持装置あるいは生命安全用途にマイクロチップ社の製品を使用する事は全て購入者のリスクとし、また購入者はこれによって発生したあらゆる損害、クレーム、訴訟、費用に関して、マイクロチップ社は擁護され、免責され、損害受けない事に同意するものとします。暗黙的あるいは明示的を問わず、マイクロチップ社が知的財産権を保有しているライセンスは一切譲渡されません。

商標

マイクロチップ社の名称と Microchip ロゴ、dsPIC、KEELOQ、KEELOQ ロゴ、MPLAB、PIC、PICmicro、PICSTART、PIC³² ロゴ、rPIC、UNI/O は、米国およびその他の国におけるマイクロチップ・テクノロジー社の登録商標です。

FilterLab、Hampshire、HI-TECH C、Linear Active Thermistor、MXDEV、MXLAB、SEEVAL、Embedded Control Solutions Company は、米国におけるマイクロチップ・テクノロジー社の登録商標です。

Analog-for-the-Digital Age、Application Maestro、CodeGuard、dsPICDEM、dsPICDEM.net、dsPICworks、dsSPEAK、ECAN、ECONOMONITOR、FanSense、HI-TIDE、In-Circuit Serial Programming、ICSP、Mindi、MiWi、MPASM、MPLAB Certified ロゴ、MPLIB、MPLINK、mTouch、Omniscient Code Generation、PICC、PICC-18、PICDEM、PICDEM.net、PICkit、PICKtail、REAL ICE、rLAB、Select Mode、Total Endurance、TSHARC、UniWinDriver、WiperLock、ZENA は、米国およびその他の国におけるマイクロチップ・テクノロジー社の登録商標です。

SQTP は、米国におけるマイクロチップ・テクノロジー社のサービスマークです。

その他、本書に記載されている商標は各社に帰属します。

© 2011, Microchip Technology Incorporated, All Rights Reserved.

ISBN: 978-1-60932-865-8

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
== ISO/TS 16949:2002 ==

マイクロチップ社では、Chandler および Tempe (アリゾナ州)、Gresham (オレゴン州) の本部、設計部およびウェハー製造工場そしてカリフォルニア州とイダホのデザインセンターが ISO/TS-16949:2002 認証を取得しています。マイクロチップ社の品質システムプロセスおよび手順は、PIC@MCU および dsPIC@DSC、KEELOQ® コード ホッピング デバイス、シリアル EEPROM、マイクロベリフェラル、不揮発性メモリ、アナログ製品に採用されています。さらに、開発システムの設計と製造に関するマイクロチップ社の品質システムは ISO 9001:2000 認証を取得しています。

各国の営業所とサービス

北米

本社
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel:480-792-7200
Fax:480-792-7277
技術サポート :
<http://www.microchip.com/support>
URL:
www.microchip.com

アトランタ
Duluth, GA
Tel:678-957-9614
Fax:678-957-1455

ボストン
Westborough, MA
Tel:774-760-0087
Fax:774-760-0088

シカゴ
Itasca, IL
Tel:630-285-0071
Fax:630-285-0075

クリーブランド
Independence, OH
Tel:216-447-0464
Fax:216-447-0643

ダラス
Addison, TX
Tel:972-818-7423
Fax:972-818-2924

デトロイト
Farmington Hills, MI
Tel:248-538-2250
Fax:248-538-2260

インディアナポリス
Noblesville, IN
Tel:317-773-8323
Fax:317-773-5453

ロサンゼルス
Mission Viejo, CA
Tel:949-462-9523
Fax:949-462-9608

サンタクララ
Santa Clara, CA
Tel:408-961-6444
Fax:408-961-6445

トロント
Mississauga, Ontario,
Canada
Tel:905-673-0699
Fax:905-673-6509

アジア / 太平洋

アジア太平洋支社
Suites 3707-14, 37th Floor
Tower 6, The Gateway
Harbour City, Kowloon
Hong Kong
Tel:852-2401-1200
Fax:852-2401-3431

オーストラリア - シドニー
Tel:61-2-9868-6733
Fax:61-2-9868-6755

中国 - 北京
Tel:86-10-8569-7000
Fax:86-10-8528-2104

中国 - 成都
Tel:86-28-8665-5511
Fax:86-28-8665-7889

中国 - 重慶
Tel:86-23-8980-9588
Fax:86-23-8980-9500

中国 - 武漢
Tel:86-571-2819-3180
Fax:86-571-2819-3189

中国 - 香港 SAR
Tel:852-2401-1200
Fax:852-2401-3431

中国 - 南京
Tel:86-25-8473-2460
Fax:86-25-8473-2470

中国 - 青島
Tel:86-532-8502-7355
Fax:86-532-8502-7205

中国 - 上海
Tel:86-21-5407-5533
Fax:86-21-5407-5066

中国 - 瀋陽
Tel:86-24-2334-2829
Fax:86-24-2334-2393

中国 - 深圳
Tel:86-755-8203-2660
Fax:86-755-8203-1760

中国 - 武漢
Tel:86-27-5980-5300
Fax:86-27-5980-5118

中国 - 西安
Tel:86-29-8833-7252
Fax:86-29-8833-7256

中国 - 厦門
Tel:86-592-2388138
Fax:86-592-2388130

中国 - 珠海
Tel:86-756-3210040
Fax:86-756-3210049

アジア / 太平洋

インド - バンガロール
Tel:91-80-3090-4444
Fax:91-80-3090-4123

インド - ニューデリー
Tel:91-11-4160-8631
Fax:91-11-4160-8632

インド - プネ
Tel:91-20-2566-1512
Fax:91-20-2566-1513

日本 - 横浜
Tel:81-45-471- 6166
Fax:81-45-471-6122

韓国 - 大邱
Tel:82-53-744-4301
Fax:82-53-744-4302

韓国 - ソウル
Tel:82-2-554-7200
Fax:82-2-558-5932 または
82-2-558-5934

マレーシア - クアラルンプール
Tel:60-3-6201-9857
Fax:60-3-6201-9859

マレーシア - ペナン
Tel:60-4-227-8870
Fax:60-4-227-4068

フィリピン - マニラ
Tel:63-2-634-9065
Fax:63-2-634-9069

シンガポール
Tel:65-6334-8870
Fax:65-6334-8850

台湾 - 新竹
Tel:886-3-6578-300
Fax:886-3-6578-370

台湾 - 高雄
Tel:886-7-213-7830
Fax:886-7-330-9305

台湾 - 台北
Tel:886-2-2500-6610
Fax:886-2-2508-0102

タイ - バンコク
Tel:66-2-694-1351
Fax:66-2-694-1350

ヨーロッパ

オーストラリア - ヴェルス
Tel:43-7242-2244-39
Fax:43-7242-2244-393

デンマーク - コペンハーゲン
Tel:45-4450-2828
Fax:45-4485-2829

フランス - パリ
Tel:33-1-69-53-63-20
Fax:33-1-69-30-90-79

ドイツ - ミュンヘン
Tel:49-89-627-144-0
Fax:49-89-627-144-44

イタリア - ミラノ
Tel:39-0331-742611
Fax:39-0331-466781

オランダ - ドリユール
Tel:31-416-690399
Fax:31-416-690340

スペイン - マドリッド
Tel:34-91-708-08-90
Fax:34-91-708-08-91

イギリス - ウォーキンガム
Tel:44-118-921-5869
Fax:44-118-921-5820