
第3章 データ メモリ

ハイライト

本章では次のトピックについて説明します。

3.1	はじめに	8-2
3.2	データ メモリ構成	8-2
3.3	データ 配置	8-5
3.4	ソフトウェア スタック	8-6
3.5	プログラム メモリ空間とデータ メモリ空間のインターフェース	8-6
3.6	関連するアプリケーション ノート	8-7
3.7	改版履歴	8-8

3.1 はじめに

PIC24F マイクロ コントローラはハーバードアーキテクチャデバイスなので、プログラムメモリ空間とデータメモリ空間のバスが分離されています。さらに、PIC24F アーキテクチャにより、コード実行中にデータ空間からプログラムメモリをアクセス可能になります。

3.2 データメモリ構成

3.2.1 データアドレス空間

PIC24F コアは線形領域としてアドレスできる 16 ビット幅のデータメモリ空間を持っています。データ空間は読み出しと書き込み用の 2 つのアドレス生成ユニット (AGU) を使用してアクセスします。データ空間メモリマップを図 3-1 に示します。

データメモリ空間用の有効アドレスはすべて 16 ビット幅で、データ空間をバイトで指定できます。これにより、データ空間のアドレス範囲は 64 K バイトか 32 K ワードとなります。データメモリ空間の下位 32 K バイト (つまり、 $EA<15> = 0$ の場合) は、実装されたメモリアドレスとして使用され、上位半分 ($EA<15> = 1$) は、プログラム空間可視化 (PSV) 用として確保されています。PSV の詳細は、4.4 項「データ空間からのプログラム空間可視化」を参照して下さい。

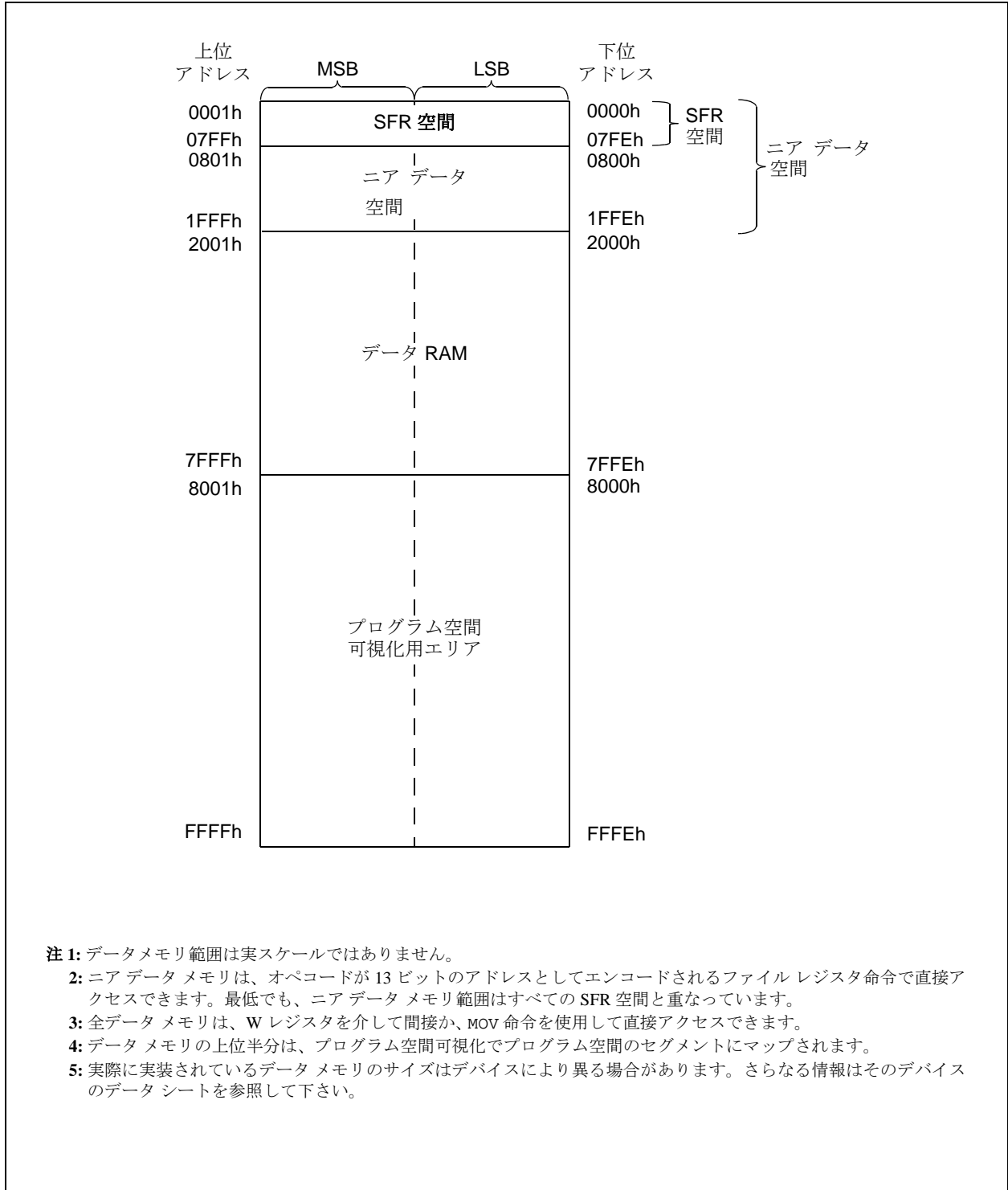
注 1: 特定のデバイスの実装データメモリについては、各デバイスのデータシートを参照して下さい。

2: もし EA がデバイスに物理的に実装されているデータメモリ範囲外の場所を指している場合は、すべてゼロのワードかバイトが返されます。

3.2.2 データ空間の幅

データメモリ空間は 16 ビット幅ブロックでバイトでアドレスできるように構成されています。データメモリとレジスタにはデータが 16 ビットワードで配置されますが、全空間の有効アドレス (EA) はバイトに分けて扱えます。各ワードの下位バイトは偶数アドレスで、上位バイトは奇数アドレスです。

図 3-1: PIC24F デバイスのデータ空間メモリ マップ



- 注 1:** データメモリ範囲は実スケールではありません。
- 2:** ニア データ メモリは、オペコードが 13 ビットのアドレスとしてエンコードされるファイル レジスタ命令で直接アクセスできます。最低でも、ニア データ メモリ範囲はすべての SFR 空間と重なっています。
- 3:** 全データ メモリは、W レジスタを介して間接か、MOV 命令を使用して直接アクセスできます。
- 4:** データ メモリの上位半分は、プログラム空間可視化でプログラム空間のセグメントにマップされます。
- 5:** 実際に実装されているデータ メモリのサイズはデバイスにより異なる場合があります。さらなる情報はそのデバイスのデータ シートを参照して下さい。

3.2.3 ニア データ メモリ

0x0000 から 0x1FFF の間の 8 K バイトのアドレス空間は、ニア データ メモリとして参照されます。ニア データ メモリは、すべてのファイル レジスタ命令内の 13 ビット絶対アドレス フィールドを介して直接アドレスできます。

さらに、ニア データ メモリは、すべての間接アドレッシング モードでアドレスでき、任意の 16 ビットの作業レジスタにより指定できます。0x1FFF を超えるデータ メモリ範囲は、間接アドレッシング モードのみでアドレス指定できます。

ニア データ範囲に含まれるメモリ範囲は、それぞれの PIC24F ファミリ デバイスに実装されたデータ メモリの量に依存します。最低でも、ニア データ範囲は、すべての SFR を含みます。詳しくは図 3-1 を参照してください。

3.2.4 SFR 空間

0000h から 07FFh のニア データ空間の最初の 2 K バイトは特殊機能レジスタ (SFR) として第一に確保されています。これらは、PIC24F コアと周辺モジュールで使用され、デバイスの動作を制御します。

SFRは制御するモジュールに分散されていて、モジュールごとにグループ化されています。SFR 空間の多くが未使用アドレスで、「0」として読み出されます。実際に実装されている SFR 空間の図を表 3-1 に示します。各実装範囲は 32 バイトの範囲で、少なくとも 1 アドレスが SFR として実装されています。

表 3-1: SFR データ空間に実装されている範囲⁽¹⁾

SFR 空間アドレス								
	xx00	xx20	xx40	xx60	xx80	xxA0	xxC0	xxE0
000h	コア			ICN	割り込み			—
100h	タイマ		キャプチャ	—	コンペア	—	—	—
200h	I ² C™	UART	SPI		—	—	I/O	
300h	A/D		—	—	—	—	—	—
400h	—	—	—	—	—	—	—	—
500h	—	—	—	—	—	—	—	—
600h	PMP	RTC/Comp	CRC	—	—	—	I/O	
700h	—	—	システム	NVM/PMD	—	—	—	—

凡例：— = このブロックには SFR は実装されていない。

注 1: 実際のレジスタ実装についてはそのデバイスのデータ シートを参照して下さい。

3.3 データ配置

PIC デバイスとの互換性を維持するためと、データ空間メモリの使用効率を良くするため、PIC24F 命令はワードとバイトアクセスの両方をサポートしています。バイトアクセス可能としていますが、有効アドレス (EA) の計算は内部的にすべてワード配置されたメモリに合わせて行われています。

16 ビットのデータアドレスの最下位ビットはワード操作の場合は無視されます。ワードデータは、リトルエンディアン形式で配置されます。つまり下位バイト (LSB) は偶数アドレス (LSb=0) に配置され、上位バイト (MSB) は奇数アドレス (LSb=1) に配置されます。

バイト操作では、データアドレスの最下位ビットは、アクセスするバイトを選択するために使われます。図 3-2 にワードとバイト操作時のデータ配置を示します。

図 3-2: データ配置

	15	MSB	8	7	LSB	0	
0001	バイト 1		バイト 0				0000
0003	バイト 3		バイト 2				0002
0005	バイト 5		バイト 4				0004
	ワード 0						0006
	ワード 1						0008
	Long ワード <15:0>						000A
	Long ワード <31:16>						000C

バイトデータの読み出しは、まずそのバイトを含むワード全体が読み出され、EA の LSb でどちらのバイトかが選択されます。選択されたバイトは、データバスの LSB 側に配置されます。つまり、データメモリとレジスタは同一のアドレス (ワード) を共用する 2 つのバイト幅の並列データとして扱われますが、書き込み信号は別々となっています。バイトデータの書き込みの場合は、バイトアドレスの一致する側だけのアレイまたはレジスタに書き込まれます。

有効アドレス (EA) のすべての計算は、バイトアクセスかワードアクセスかにより自動的に調整されます。例えば、後置インクリメントのアドレスポインタのアドレスは、ワード操作の場合は 2 つつインクリメントされます。同様に、後置インクリメントのアドレスポインタのアドレスは、バイト操作の場合は 1 つつインクリメントされます。

ワードアクセスはすべて偶数アドレスで行う必要があります。配置を誤ったデータフェッチはサポートされていないので、ワードとバイトを混在させる場合や、8 ビットの MCU コードを移植する場合には注意して下さい。誤配置の読み書きを実行すると、アドレスエラートラップが発生します。読み出し時にはエラーが発生しても命令は実行されますが、書き込み時に発生すると、命令は実行されますが書き込みは行われません。どちらの場合にもその後トラップが実行され、システムやユーザーがアドレスフォルトの前にマシン状態を確認することができます。割り込みに関する詳細は、第 8 章「割り込み」を参照して下さい。

W レジスタへのバイトロードはすべて下位バイト側に行われ、上位バイトは変更されません。

符号拡張命令 (SE) は、符号つき 8 ビットデータを符号つき 16 ビット値に変換します。逆に、符号なし 8 ビットデータの場合は、適切なアドレスにゼロ拡張命令 (ZE) を実行し、W レジスタの上位バイト (MSB) をゼロにします。

大部分の命令がワードとバイトの両方の操作ができますが、ワード操作しかできない命令もあることに注意して下さい。

3.4 ソフトウェア スタック

スタック動作のため、PIC24F デバイスのデータ メモリの一部がスタックとして使われません。ソフトウェア スタックの詳細については、**2.3 項 「ソフトウェア スタックポインタ」**を参照して下さい。

3.5 プログラム メモリ空間とデータ メモリ空間のインターフェース

PIC24F アーキテクチャでは、24 ビット幅のプログラム空間と、16 ビット幅のデータ空間を持っています。さらにこのアーキテクチャは、改良ハーバード構成となっていますので、データをプログラム空間に置くこともできます。このデータをうまく使用するには、両方の空間の配置情報にしたがってアクセスする必要があります。

通常の実行を除き、PIC24F アーキテクチャは動作中のプログラム空間のアクセスを可能とするために 2 つの方法を提供しています。

- プログラム空間の任意の位置のバイトまたはワードの個別アクセスをするためにテーブル命令を使用する
- プログラム空間の一部をデータ空間に再マップする (プログラム空間可視化)

テーブル命令では、アプリケーションでプログラム メモリの狭い範囲を読み書きできます。この方法は、頻繁に更新が必要なデータ テーブルのアクセスに理想的です。また、この方法ではプログラム ワードのすべてのバイトにアクセスが可能です。再マッピングする方法により、アプリケーションがスタティック データの大きなテーブルを参照するのに理想的な読み出し専用の大きなデータ ブロックをアクセスすることができます。さらにこの方法では、プログラム メモリの下位ワードにのみアクセスすることが可能です。

プログラム メモリとデータ メモリのインターフェースに関する詳細については、**4.3 項 「プログラム メモリからのデータ アクセス」**を参照して下さい。

3.6 関連するアプリケーションノート

この項では、マニュアルのこの章に関連するアプリケーションノートをリストアップします。これらのアプリケーションノートは、特に PIC24F デバイス ファミリー用に書かれているわけではありませんが、その概念は適切であり、変更あるいは制限事項も考慮に入れて使用可能です。現状、データメモリに関連するアプリケーションノートは次の通りです。

タイトル	アプリケーションノート#
現在関連するアプリケーションノートはありません。	

注: PIC24F ファミリー デバイスに関するその他のアプリケーションノートやコード例についてはマイクロチップ ウェブ サイト (www.microchip.com) をご覧下さい。

3.7 改版履歴

リビジョン A (2007 年 1 月)

本文書の初版リリース。