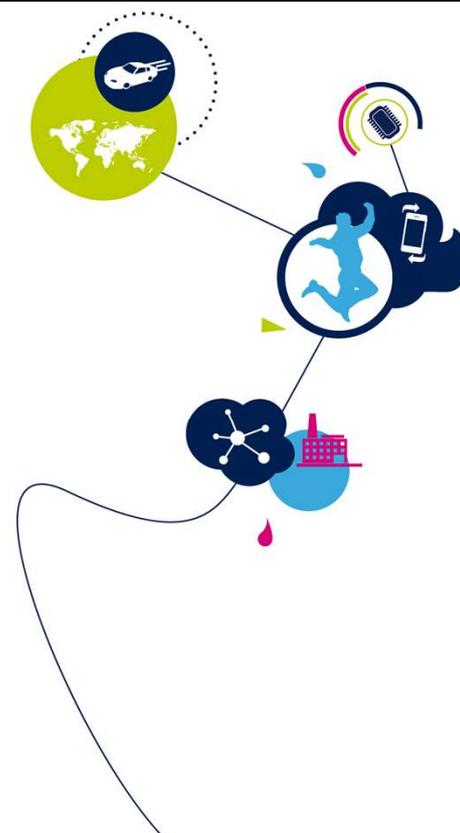
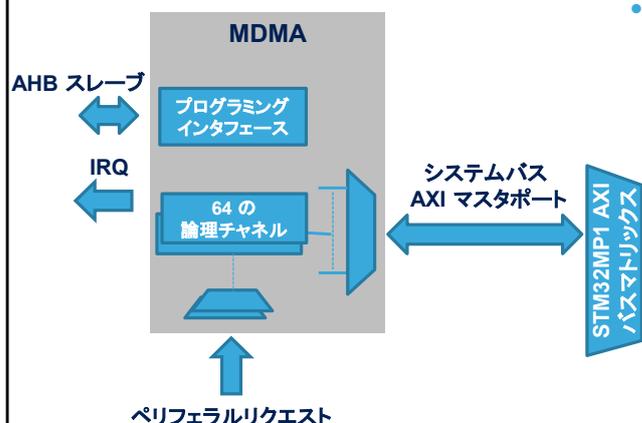


# STM32MP1 - MDMA

マスタダイレクトメモリアクセスコントローラ (MDMA)  
1.0 版



STM32MP1 マスタダイレクトメモリアクセスコントローラ (MDMA) のプレゼンテーションへようこそ。ここでは、データ転送の処理に広く使われている本モジュールの主な特徴を説明します。



### • STM32MP1 MDMA の特徴

- 64bit AXI マスタバス
- 柔軟性の高い個別の設定
- 2つの物理チャンネル(読出し用の 1 x 128バイト FIFO と書き込み用の 1 x 128バイト FIFO)を最大 64 の論理チャンネルで共有
- ハードウェアおよびソフトウェアの転送トリガ
- ハードウェアおよびソフトウェアの優先順位管理
- 設定可能なデータ転送モード
  - ペリフェラルからメモリ、メモリからペリフェラル、メモリからメモリのモード

### アプリケーション側の利点

- ブロック転送とリンクされたリストの MDMA のサポート
- CPU からデータ転送管理負荷を軽減
- 簡単な組込み



マスタダイレクトメモリアクセス(MDMA)は、CPU の介入なしに転送のチェーンリストを実行できるリンクされたリスト転送をサポートしているため、メモリ間のデータ転送用に最適化されています。これにより、CPU リソースを他の操作のためにあけておくことができます。

MDMA コントローラは、メインメモリとペリフェラルレジスタのアクセス(システムアクセスポート)用のマスタ AXI インタフェースと、設定用のスレーブ AHB インタフェースを備えています。

- 各 MDMA チャンネルでは、以下のことが可能
  - **単一バッファの転送:** 1 つのバッファが転送される(最大 128バイト)。転送の最後で、DMA チャンネルは無効化され、割込み(有効な場合)が生成
  - **単一のブロック転送:** 1 つのブロックが転送される(最大 64KB)。ブロックの最後で、DMA チャンネルは無効化され、割込み(有効な場合)が生成される
  - **繰り返しブロック転送:** チャンネルを無効化する前に、いくつかのブロック(最大 4096ブロック)が転送される
  - **リンクされたリストの転送:** 現在のデータブロックの転送が完了すると、メモリから新しいブロック制御構造がロードされ、新しいブロック転送が開始される
- チャンネル個別の柔軟性:
  - 転送元および転送先に対する個別のインクリメント、デクリメント、またはノンインクリメントアドレッシング
  - 転送元および転送先に対する個別の転送サイズおよびインクリメントサイズ
  - エンディアン形式の入れ替え: バイト、ハーフワード、ワードの粒度



各 DMA コントローラチャンネルは、転送元と転送先の間には単方向転送リンクを備えています。

各チャンネルでは、以下を実行できます。

**単一バッファの転送:** 1 つのバッファが転送されます(最大 128バイト)。バッファの最後で、DMA チャンネルは無効化され、チャンネル転送終了割込み(有効な場合)が生成されます。

**単一のブロック転送:** 1 つのブロックが転送されます。

**繰り返しブロック転送:** いくつかのブロックが転送されます。

**リンクされたリストの転送:** 現在のデータブロック(または繰り返しの最後のブロック)の転送が完了すると、メモリから新しいブロック制御構造がロードされ、新しいブロック転送が開始されます。

MDMA は、送信元および転送先に対するアドレス指定をインクリメント、デクリメント、またはノンインクリメント(固定)にする機能も備えています。サイズとアドレスインクリメントは、転送元と転送先の両方について、個別に選択できます。

- MDMA は、システム内のすべてのメモリからデータを収集し、CPU で利用できるようにするのに役立つ
- DMA のリンクされたリストは、CPU の介入を必要とせずの一連の DMA 転送を実行するために使用される
  - 他の DMA 用のデータを準備し、その DMA 設定をセットして転送を開始するために使用可能
  - 分散／集合をサポートするために使用される。  
これは、転送元領域と転送先領域がメモリ内の連続した領域を占有する必要がないことを意味する。  
転送元および転送先データ領域は、データブロックの転送を制御する一連のリンクされたリストのディスクリプタによって定義される



MDMA は、システム内のすべてのメモリ、特に Cortex®-M4 側のメモリからデータを収集し、メイン CPU で利用できるようにするのに役立ちます。

DMA のリンクされたリストは、CPU の介入を必要とせずの一連の DMA 転送を実行するために使用されます。

設定データを DMA1 または DMA2 レジスタにロードしてから、それらの転送を開始するために、リンクされたリストを使用することもできます。

分散／集合をサポートするために使用されます。これは、転送元領域と転送先領域がメモリ内の連続した領域を占有する必要がないことを意味します。転送元および転送先データ領域は、データブロックの転送を制御する一連のリンクされたリストのディスクリプタによって定義されます。

- MDMA は、シングルまたはインクリメンタルバースト転送をサポート
- 最大 128バイトのソフトウェア設定可能なバーストサイズ
- 最大バーストデータサイズは 128バイトです。データサイズがこれより大きい場合、バースト長は、転送データを一時的に格納するために使用される 128レベルの FIFO によって制限される
  - たとえば、16x64bit または 32x32bit



MDMA では、インクリメンタルバースト転送をサポートしています。各バーストのサイズはソフトウェアで設定可能で、最大値は 128 バイトです。これよりも大きなデータサイズの場合、最大 128 バイトのデータバーストサイズを考慮して、バースト長は制限されません(16x64bit または 32x32bit など)。

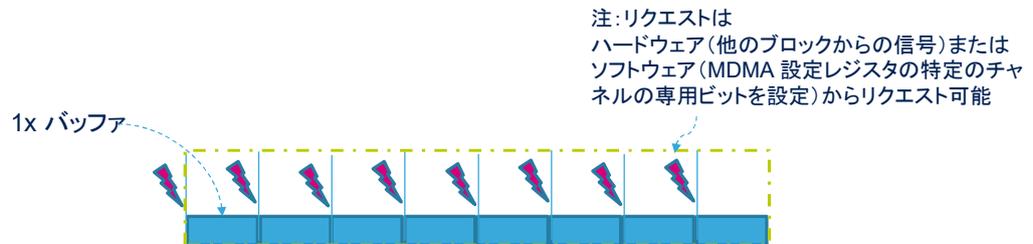
# MDMA チャンネルトリガモード

6

- シングルリクエストで転送されるデータ配列のサイズは、以下のいずれか

1. バッファ転送サイズ

- TRGM = "00"



リクエストごとに 1 バッファが転送  
割込みを有効にしていれば、バッファ転送のたびに割込みを生成



サイズは、TRGM[1:0] (トリガモード) 選択フィールドを使用して選択されます。シングルリクエストで転送されるデータ配列のサイズは、TRGM (トリガモード) = "00" の場合のバッファ転送サイズになります。

# MDMA チャネルトリガモード

7

- シングルリクエストで転送されるデータ配列のサイズは、以下のいずれか
  1. バッファ転送サイズ
    - TRGM = "00"
  2. ブロックサイズ
    - TRGM = "01"



リクエストごとに 1ブロックが転送

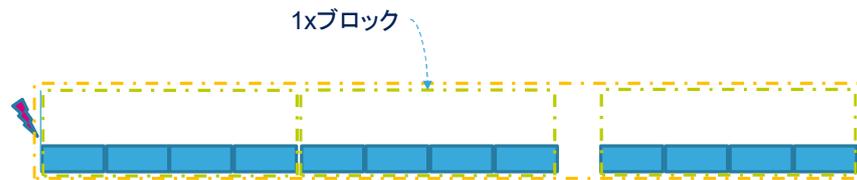
割込みを有効にしていれば、バッファ転送またはブロック転送のたびに割込みを生成



life.augmented

TRGM(トリガモード)="01"の場合、シングルリクエストで転送されるデータ配列のサイズはブロックサイズとなります。

- シングルリクエストで転送されるデータ配列のサイズは、以下のいずれか
  1. バッファ転送サイズ
    - TRGM = "00"
  2. ブロックサイズ
    - TRGM = "01"
  3. 繰り返しブロック
    - TRGM = "10"



リクエストごとに複数ブロックが転送

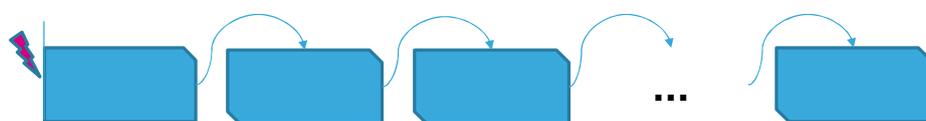
割込みを有効にしていれば、バッファ転送またはブロック転送のたびに、または最後のブロック転送後のみに割込みを生成

TRGM = "10" の場合、シングルリクエストで転送されるデータ配列のサイズは繰り返されるブロックとなります。

# MDMA チャンネルトリガモード

9

- シングルリクエストで転送されるデータ配列のサイズは、以下のいずれか
  1. バッファ転送サイズ
    - TRGM = "00"
  2. ブロックサイズ
    - TRGM = "01"
  3. 繰り返しブロック
    - TRGM = "10"
  4. チャンネルデータ全体
    - TRGM = "11"



1 回のリクエストで転送が開始され、チャンネルのリンクされたリストポインタが null になるまで継続



TRGM = "11" の場合、シングルリクエストで転送されるデータ配列のサイズは、チャンネルデータ全体 (チャンネルのリンクリストポインタが null になるまで) になります。

## リクエストアービトレーション(1/2)

10

- 各チャンネルには、プログラム可能な SW 優先順位(4 つの優先順位レベル)がある
  - 2 つのチャンネルの SW 優先順位が同じ場合、小さいチャンネル番号が優先
- MDMA チャンネルリクエスト間の新たなアービトレーションを実行せずに MDMA によって転送される最小データサイズは 1 回のバッファ転送
- ブロック転送の場合、個々のバッファの転送後、MDMA は、新しい外部リクエストと内部的に記憶されたリクエスト間での新たなアービトレーションのフェーズに移行
  - 他に優先順位の高いチャンネルリクエストがアクティブでない場合は、新しいバッファ転送が同じチャンネルで開始される
  - 各バッファ転送後にチャンネルアービトレーションが実行されるため、最高レベルの MDMA リクエストがバッファ転送期間よりも長い期間ブロックされることはない



各チャンネルには、プログラム可能な優先順位があります。2 つのチャンネルのプログラム可能優先順位が同じ場合、小さいチャンネル番号が優先されます。

アービタは、優先順位に基づいて MDMA チャンネルリクエストを管理します。MDMA がアイドル状態の場合、各バッファの転送終了後に、すべての MDMA リクエスト(ハードウェアまたはソフトウェア)がすべての有効チャンネルで確認されます。

MDMA チャンネルリクエスト間の新たなアービトレーションを開始せずに MDMA によって転送される最小データサイズは 1 回のバッファ転送です。

ブロック転送の場合、個々のバッファの転送後、MDMA は、新しい外部リクエストと内部的に記憶されたリクエスト間での新たなアービトレーションのフェーズに移行します。

他に優先順位の高いチャンネルリクエストがアクティブでない場合は、新しいバッファ転送が同じチャンネルで開始されます。

各バッファ転送後にチャンネルアービトレーションが実行されるため、最高レベルの MDMA リクエストがバッファ転送期間よりも長い期間ブロックされることはありません。

## リクエストアービトレーション(2/2)

11

- 2つのデータ配列サイズパラメータが MDMA とアプリケーションの応答性に影響がある
  - 1. バッファ転送サイズ:他のチャンネルのリクエストによって MDMA レベルでは中断できないデータ転送長
  - 2. AXI バーストサイズ:バスアービトレーションレベルで中断できない最大データ転送長を定義
    - バーストモードで転送できるデータ転送長
    - 他のマスタがバスへアクセスすることを阻止することがある
- 他の MDMA チャンネルおよびマスタの「リアルタイム」要件を考慮して、バーストおよびバッファ転送サイズに関して適正なトレードオフを行うことが重要



以下の 2 つのデータ配列サイズパラメータが MDMA とアプリケーションの応答性に影響を与えます。

1. バッファ転送サイズ:他のチャンネルのリクエストによって MDMA レベルでは中断できないデータ転送長
2. AXI バーストサイズ:これはバスアービトレーションレベルで中断できない最大データ転送長を定義します。
  - バーストモードで転送できるデータ転送長です。
  - 他のマスタがバスへアクセスすることを阻止することがあります。

他の MDMA チャンネルおよびマスタの「リアルタイム」要件を考慮して、バーストおよびバッファ転送サイズに関して適正なトレードオフを行うことが重要です。

- バッファ転送は、指定されたチャンネルに対して MDMA リクエストイベントで転送される最小データ量
  - 注: MDMA リクエストに続いて現在のチャンネルで転送されるデータの総量は、トリガモード(バッファ転送、ブロック転送、繰り返しブロック転送、全データ転送)によって決定
- 転送されるデータ項目の数、その幅(8bit、16bit、32bit、または 64bit)、およびデータ転送に使われるバースト長はソフトウェアでプログラム可能
- リクエストイベントを処理した後、マスクデータ値をマスクアドレスで指定されたアドレスに書き込むことによって、リクエストが MDMA によって確認応答される
  - これは「SW」確認応答であり、リクエストイベントソースフラグをクリアするために使用されるマスクアドレスは、たとえば、DMA1 または DMA2 内のレジスタのアドレスになる可能性がある



バッファ転送は、1 つのチャンネルの MDMA リクエストイベントで転送される、論理的に最小のデータ量です(最大 128バイト)。MDMA バッファ転送は、一定数の AXI データ転送(シングルまたはバーストデータ転送)のシーケンスで構成されています。転送されるデータ項目の数とその幅(8bit、16bit、32bit、または 64bit)は、ソフトウェアでプログラム可能です。データ転送に使用するバースト長も個別にプログラム可能です。データ配列の転送を必要とするイベントの後、リクエスト信号が MDMA コントローラに送信されます。MDMA コントローラは、チャンネルの優先順位に応じて、リクエストを処理します。これらのレジスタが設定されている場合に、マスクアドレスに指定されたアドレスにマスクデータ値を書き込むことによって、リクエストが確認応答されます。

## MDMA ブロック転送 (1/2)

13

- ブロックは、最大 64KB の連続したデータの配列で、連続したバッファ転送によって転送
- データの各ブロックは、「開始アドレス」と「ブロック長」で定義される
- 設定されたトリガモデルに応じて、1 つのペリフェラル/DMA リクエストによってブロック全体の転送をトリガ
  - 優先順位の高い他のチャンネルリクエストがアクティブでない場合は、新しいバッファ転送が同じチャンネルで開始される



ブロックは、最大 64KB の「連続した」データの配列で、連続したバッファ転送によって転送されます。  
データの各ブロックは、開始アドレスとブロック長で定義されます。

- ブロック転送が完了すると、以下の 3 つのアクションが実行される可能性がある
  - ブロックが、繰り返しブロック転送の一部である場合: ブロック長を再ロードし、ブロック繰り返しアドレス更新レジスタの情報に基づいて新しいブロック開始アドレスを計算
  - 現在の MDMA チャンネルに転送する必要がある最後のブロックである場合: チャンネルを無効にし、このチャンネルに対するその後の MDMA リクエストは受け付けない
  - リンクされたリスト転送の最後のブロックではない場合: 次のブロックの情報がメモリからロードされる



ブロック転送が完了すると、以下の 3 つのアクションのいずれかが実行されます。

- ブロックが、繰り返しブロック転送の一部である場合: ブロック長を再ロードし、新しいブロック開始アドレスを計算します (CxBRUR レジスタの情報に基づく)。
- 単一のブロックである場合、または繰り返しブロック転送の最後のブロックである場合: メモリから次のブロック情報をロードします (MDMA\_CxLAR のリンクされたリストアドレス情報を使用)。
- 現在の MDMA チャンネルに転送する必要がある最後のブロックである場合 (MDMA\_CxLAR = 0): チャンネルを無効にします。このチャンネルに対するその後の MDMA リクエストは受け付けません。

## MDMA ブロック繰り返しモード

15

- ブロック繰り返しモードにより、転送先と転送元の「開始アドレス」が異なる場合でもブロック転送を繰り返すことが可能
- 繰り返しブロックモードがアクティブな場合（繰り返しカウンタが“0”に等しくない場合）、現在のブロック転送の終了時に、ブロックパラメータが以下のように更新
  - 転送するデータバイトのブロック数(BNDT)の値を再ロード
  - ブロック繰り返しの転送元／転送先アドレス更新モード(BRSUM/BRDUM)の設定に従ってブロックの転送元／転送先アドレス値を更新
  - 繰り返しカウンタを 1 だけデクリメント
- 繰り返しブロックカウンタが 0 に達すると、最後のブロックは、単一のブロック転送として処理される



ブロック繰り返しモードにより、転送先と転送元の開始アドレスが異なる場合でもブロック転送を繰り返すことができます。

繰り返しブロックモードがアクティブな場合（繰り返しカウンタが“0”でない場合）、現在のブロック転送の終了時にブロックパラメータが更新され（BRSUM/BRDUM の設定に従って BNDT 値が再ロードされ、SAR/DAR 値が更新されます）、繰り返しカウンタが 1 ずつデクリメントされます。

繰り返しブロックカウンタが 0 に達すると、この最後のブロックは、単一のブロック転送として処理されます。

## MDMA のリンクされたリストモード

16

- リンクされたリストモードでは、チャンネルリンクアドレスレジスタ (CLAR) に指定されたアドレスから、新しい MDMA の設定 (CxTCR、CxBNDTR、CxSAR、CxLAR、CxBRUR、CxLAR、CxTBR、CxMAR、および CxMDR レジスタ) をロード可能
- この操作後、前述のブロック／繰り返しブロックモードで定義されたように、チャンネルは新しいリクエストを受け付ける準備ができるか、またはトリガモードがチャンネルデータ全体 (TRGM = “11”) に設定されている場合は転送を継続
- トリガソースは、トリガおよびバス選択レジスタ (TBR) の値をロードする際に自動的に変更することが可能
- トリガモードが TRGM = “11” に設定されている場合、トリガモードと選択した SW リクエストは変更してはいけない



life.augmented

リンクされたリストモードでは、CxLAR レジスタに指定されたアドレスから、新しい MDMA の設定をロードできます。このアドレスでは、AXI システムバス上に配置されたメモリを処理する必要があります。

この操作後、前述のブロック／繰り返しブロックモードで定義されたように、チャンネルは新しいリクエストを受け付ける準備ができるか、TRGM[1:0] が “11” の場合は転送を続けます。

トリガソースは CxTBR 値をロードする際に自動的に変更することができます。

TRGM および SWRM の値は、TRGM[1:0] が “11” の場合は変更しないでください。

# MDMA のリンクされたリストモード

- チャンネル設定(チャンネルリンクアドレス LAR)は AXI アドレス空間になければいけない
- LAR 値は、LAR[2:0] = 0x0 などのダブルワードアドレス境界に合わせる必要あり
- 注: 以下のアドレスがすべて連続しているわけではないことに注意

レジスタ(32bit ワード)	リンクアドレスレジスタからのオフセット	説明
CTCR	0x00	転送設定レジスタ
CBNDTR	0x04	データのブロック数レジスタ
CSAR	0x08	転送元アドレスレジスタ
CDAR	0x0C	転送先アドレスレジスタ
CBRUR	0x10	ブロック繰り返しアドレス更新レジスタ
CLAR	0x14	リンクアドレスレジスタ: 次のディスクリプタ
CTBR	0x18	トリガおよびバス選択レジスタ
CMAR	0x20	マスクアドレスレジスタ
CMDR	0x24	マスクデータレジスタ



チャンネル設定(チャンネルリンクアドレス LAR)は AXI アドレス空間になければなりません。  
LAR 値は、LAR[2:0] = 0x0 などのダブルワードアドレス境界に合わせる必要があります。

# MDMA リクエストマッピング

MDMA リクエスト	リクエストソース	説明
mdma_str0 – 7	dma1_tcf(0 – 7)	DMA1 ストリーム 0-7 転送完了フラグ
mdma_str8 – 15	dma2_tcf(0 – 7)	DMA2 ストリーム 0-7 転送完了フラグ
mdma_str17	SDMMC1	SDMMC1 転送完了
mdma_str18	SDMMC2	SDMMC2 転送完了
mdma_str22	QUADSPI	QUADSPI FIFO 閾値
mdma_str23		QUADSPI 転送完了
mdma_str32	UART1 RX	UART 受信転送完了
mdma_str33	UART1 TX	UART 送信転送完了
mdma_str34	SPI6 RX	SPI6 受信転送完了
mdma_str35	SPI6 TX	SPI6 送信転送完了
mdma_str36	I2C4 RX	I2C4 受信転送完了
mdma_str37	I2C4 TX	I2C4 送信転送完了
mdma_str38	I2C6 RX	I2C6 受信転送完了
mdma_str39	I2C6 TX	I2C6 送信転送完了



この表は、MDMA リクエストと、デバイスへのそれらのマッピングを示しています。たとえば、MDMA チャンネルは、DMA1 ストリームの転送の終了によってトリガできます。このトリガに反応して、MDMA は次のことができます。

- MCU SRAM1、2、3、4 から SYSRAM または DDR へのデータ転送を実行する。
- または、新しい転送のために DMA1 ゼロフローを再プログラムする。

MDMA によって CPU 負荷を効率的に軽減できるようにするには、MDMA チャンネルをデバイスの割込みによってトリガして、データ交換と処理を自動化することができます。ペリフェラルのトリガの例をこの表に示します。

• 各チャンネルの割込みイベント

割込みイベント	説明
CTCIF	MDMA チャンネル転送完了。これはチャンネルイネーブルビットに 0 を書き込むことによって、セットされる
BTIF	MDMA ブロック転送完了
BRTIF	MDMA ブロック繰り返し転送完了
TCIF	MDMA バッファ転送完了。これはデバッグ機能として使用できません(割込みなし)。最後のフラグリセット以降に(少なくとも) MDMA バッファ転送が生成されたことを示している
TEIF	MDMA 転送エラー



各 MDMA チャンネルに対し、次のイベントにて割込みを生成することができます。

- ・ チャンネル転送完了
- ・ ブロック転送完了
- ・ ブロック転送繰り返し完了
- ・ バッファ転送完了
- ・ 転送エラー

# 低電力モードでの MDMA

モード	説明
RUN	アクティブ
Stop/LP-stop/LPLV-stop	停止。MDMA レジスタの内容は保持される
STANDBY	パワーダウン状態。Standby モード終了後に MDMA を再初期化する必要あり
VBAT	パワーダウン状態。VBAT モード終了後に MDMA を再初期化する必要あり

MDMA は Run モードでアクティブです。各種 Stop モードでは、MDMA は停止し、MDMA のレジスタの内容は保持されます。Standby モードおよび VBAT モードでは、MDMA はパワーダウンされ、これらのモード終了後に MDMA レジスタを再初期化する必要があります。