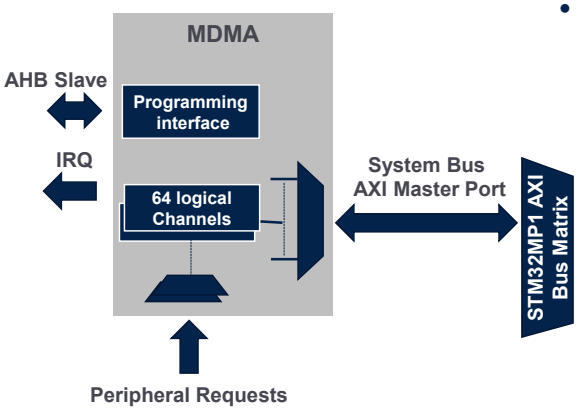
Welcome to the presentation of the STM32MP1 master direct memory access controller (MDMA). It covers the main features of this module, which is widely used to handle data transfers.

Overview



- STM32MP1 MDMA features
  - 64-bit AXI master bus
  - Flexible and independent configuration
  - 2 physical channels (1 x 128-byte FIFO for Read and 1 x 128-byte FIFO for Write) are shared by up to 64 logical channels.
  - Hardware and software transfer trigger
  - Hardware and software priority management
  - Configurable data transfer modes
    - Peripheral-to-Memory, Memory-to-Peripheral, and Memory-to-Memory modes

Application benefits

- MDMA support for block transfer and linked list
- Offload CPU from data transfer management
- Simple integration

life.augmented

2

The master direct memory access (MDMA) is optimized for data transfers between memories since it supports linked list transfers which allow a chained list of transfers to be performed without the need for CPU intervention. This keeps the CPU resources free for other operations.

The MDMA controller provides a master AXI interface for main memory and peripheral registers access (system access port) and a slave AHB interface for configuration.

Key features

- Each MDMA channel can perform:
  - **Single buffer transfer:** one buffer is transferred (up to 128 bytes). At the end of the transfer, the DMA channel is disabled and, if enabled, an interrupt is generated.
  - **Single block transfer:** one block is transferred (up to 64 Kbytes). At the end of the block, the DMA channel is disabled and, if enabled, an interrupt is generated.
  - **Repeated block transfer:** a number of blocks (up to 4096 blocks) are transferred before disabling the channel.
  - **Linked list transfer:** when the transfer of the current data is completed, a new block control structure is loaded from memory and a new block transfer is started.
- Individual channel flexibility:
  - Independent incrementing, decrementing or non-incrementing for source and destination
  - Independent transfer size, and increment size for source and destination
  - Endianness exchange: Byte, Half-word and Word granularity

life.augmented

3

Each of the DMA controller channels provides a unidirectional transfer link between a source and a destination.
Each channel can perform:
Single buffer transfer: one buffer is transferred (up to 128 bytes). At the end of the buffer, the DMA channel is disabled and, if enabled, an End of Channel Transfer interrupt is generated.
Single block transfer: one block is transferred.
Repeated block transfer: a number of blocks is transferred.
Linked list transfer: when the transfer of the current data block (or last block in a repeat) is completed, a new block control structure is loaded from memory and a new block transfer is started.

The MDMA also features incrementing, decrementing or non-incrementing (fixed) addressing for source and destination. The size and address increment for both source

and destination can be independently selected.

MDMA use case

- MDMA is useful to collect data from all memories in the system and make them available to the CPU.
- DMA linked lists are used to perform a set of DMA transfers without the need for CPU intervention:
  - Can be used to prepare data for other DMAs and then set the DMA configuration to start transfers.
  - It is used to support scatter/gathering. This means that the source and destination areas do not need to occupy contiguous areas in memory. The source and destination data areas are defined by a series of linked list descriptors that control the transfer of data blocks.

life.augmented

MDMA is useful to collect data from all memories in the system, especially from memories on Cortex-M4-side, and make them available to main CPU.
DMA linked lists are used to perform a set of DMA transfers without the need for CPU intervention.
Linked lists can also be used to load configuration data into DMA1 or DMA2 registers and then start them.
It is used to support scatter/gathering. This means that the source and destination areas do not need to occupy contiguous areas in memory. The source and destination data areas are defined by a series of linked list descriptors which control the transfer of data blocks.

Transfer types

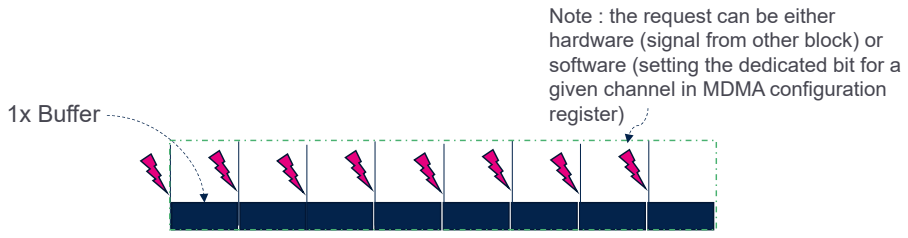
- MDMA supports single or incremental burst transfers
- Software-configurable burst sizes up to 128 bytes
- Maximum burst data size is 128 bytes. For larger data sizes, the burst length is limited by the 128-level FIFO that is used to store temporary the data to be transferred
  - e.g. 16x64-bit or 32x32-bit

life.augmented

The MDMA supports incremental burst transfers. The size of the burst is software-configurable, up to 128 bytes. For larger data sizes, the burst length is limited, so as to respect the maximum data burst size of 128 bytes (for example 16x64-bit or 32x32-bit).

MDMA channel triggering mode

- The size of the data array to be transferred for a single request is one of the following:
  1. Buffer transfer size
  - TRGM = ’00’



One buffer is transferred for each request.
If enabled, interrupt can be generated after every buffer transfer

life.augmented

The size is selected using the TRGM[1:0] (Trigger Mode) selection field. The size of the data array to be transferred for a single request will be the buffer transfer size when the TRGM (Trigger mode) = ‘00’.

MDMA channel triggering mode

- The size of the data array to be transferred for a single request is one of the following:
  1. The buffer transfer size
  - TRGM = ’00’
  2. The block size
  - TRGM = ’01’

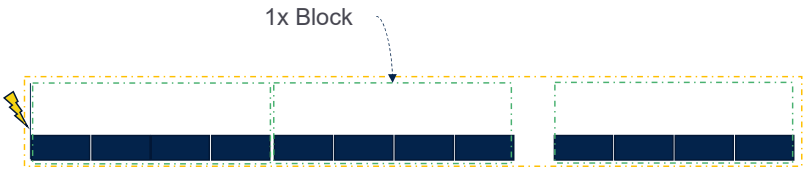
One block is transferred for each request
If enabled, interrupt can be generated after every buffer or every block transfer

life.augmented

The size of the data array to be transferred for a single request will be the block size when the TRGM (Trigger Mode) =’01’.

MDMA channel triggering mode

- The size of the data array to be transferred for a single request is one of the following:
  1. The buffer transfer size
  - TRGM = ’00’
  2. The block size
  - TRGM = ’01’
  3. Repeated block
  - TRGM = ’10’



Several blocks are transferred for each request
If enabled, interrupt can be generated after every buffer or every block transfer or only after last block transfer

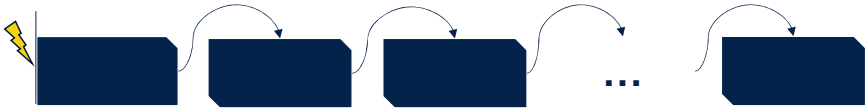
life.augmented

8

The size of the data array to be transferred for a single request will be a repeated block when TRGM = ’10’.

MDMA channel triggering mode

- The size of the data array to be transferred for a single request is one of the following:
  1. The buffer transfer size
  - TRGM = ’00’
  2. The block size
  - TRGM = ’01’
  3. Repeated block
  - TRGM = ’10’
  4. Complete channel data
  - TRGM = ‘11’



A request starts the transfer until the linked list pointer for the channel is null

life.augmented

The size of the data array to be transferred for a single request will be a complete channel data (until the linked list pointer for the channel is null) when TRGM = ’11’.

Request Arbitration (1/2)

- Each channel has programmable SW priority (4 priority levels)
  - If two channels have same SW priority, the lower channel number has priority
- A buffer transfer is the minimum data size to be transferred by the MDMA without performing a new arbitration between MDMA channel requests
- In case of a block transfer, after transferring an individual buffer, the MDMA enters in a new arbitration phase between new external requests and internally memorized ones.
  - If no other channel request with a higher priority is active, a new buffer transfer is started for the same channel
  - Highest level MDMA requests are never blocked for a duration longer than a buffer transfer period, as the channel arbitration is performed after each buffer transfer

life.augmented

10

Each channel has programmable priority. If two channels have the same programmable priority, the lower channel number has higher priority.
An arbiter manages the MDMA channel requests based on their priority. When MDMA is idle and after the end of each buffer transfer, all MDMA requests (hardware or software) are checked for all enabled channels.
A buffer transfer is the minimum data size to be transferred by the MDMA without starting a new arbitration between MDMA channel requests.
In case of a block transfer, after transferring an individual buffer, the MDMA enters a new arbitration phase between new external requests and internally memorized ones.
If no other channel request with a higher priority is active, a new buffer transfer is started for the same channel.
Highest level MDMA requests are never blocked for a duration longer than a buffer transfer period as the channel arbitration is performed after each buffer transfer.

Request Arbitration (2/2)

- Two data array size parameters have an impact on MDMA and application responsiveness
  1. Buffer transfer size: data transfer lengths which are uninterruptible at the MDMA level from other channels' requests.
  2. AXI Burst size: it defines the maximum data transfer length which may be uninterruptible at the bus arbitration level
     - It is the length of the data transfer which may be transferred in burst mode.
     - It may block other masters from gaining access to the bus
- It is important to make the correct tradeoff regarding burst and buffer transfer size, considering the "real-time" requirements for other MDMA channels and masters.

life.augmented

11

Two data array size parameters have an impact on MDMA and application responsiveness:

1. Buffer transfer size: data transfer lengths which are uninterruptible at the MDMA level from other channels' requests.
2. AXI Burst size: it defines the maximum data transfer length which may be uninterruptible at the bus arbitration level.
   - It is the length of the data transfer that may be transferred in burst mode.
   - It may block other masters from gaining access to the bus.

It is important to make the correct tradeoff regarding burst and buffer transfer size, considering the "real-time" requirements for other MDMA channels and masters.

MDMA buffer transfer

- A buffer transfer is the minimum amount of data which is transferred on an MDMA request event, for a given channel.
  - Note : the total amount of data to be transferred on the current channel, following a MDMA request, is determined by the triggering mode (buffer transfer, block transfer, repeated block transfer, all data transfer).
- The number of data items to be transferred, their width (8-bit, 16-bit, 32-bit or 64-bit) and the length of the burst used for data transfers, are software programmable.
- After servicing a request event, the request is acknowledged by MDMA writing the mask data value to the address given in the mask address.
  - This is a “SW” acknowledge, used to clear the request event source flag: the mask address can be e.g. the address of a register in DMA1 or DMA2.

life.augmented

A buffer transfer is the minimum logical amount of data (up to 128 bytes) which is transferred on an MDMA request event, on one channel.
An MDMA buffer transfer consists of a sequence of a given number of AXI data transfers (done as single or burst data transfers). The number of data items to be transferred and their width (8-bit, 16-bit, 32-bit or 64-bit) are software-programmable. The length of the burst used for data transfers is also programmable, independently.
After an event requiring a data array to be transferred, a request signal is sent to the MDMA controller. The MDMA controller serves the request according to the channel priorities. The request is acknowledged by writing the mask data value to the address given in the mask address, when these registers are set.

MDMA Block transfer (1/2)

- A block is a contiguous array of data, up to 64 Kbytes, which is transferred by successive buffer transfers.
- Each block of data is defined by a “start address” and “block length”.
- Depending on the triggering model configured, a whole block transfer can be triggered by one peripheral/DMA request.
  - A new buffer transfer is started for the same channel if no other channel request with a higher priority is active

life.augmented

A block is a “contiguous” array of data, up to 64 Kbytes, which is transferred by successive buffer transfers.
Each block of data is defined by the start address and the block length.

MDMA Block transfer (2/2)

- When a block transfer is completed, three actions may be executed:
  - If the block is part of a repeated block transfer: the block length is reloaded and a new block start address is computed based on the information in the Block Repeat address Update register.
  - If it is the last block which needs to be transferred for the current MDMA channel : the channel is disabled and no further MDMA requests is accepted for this channel.
  - If it is not the last block in a linked-list transfer: the next block information is loaded from the memory.

life.augmented

When a block transfer is completed, one of the following three actions can be executed:

- If the block is part of a repeated block transfer: the block length is reloaded and a new block start address is computed (based on the information in the CxBRUR register)
- If it is a single block or the last block in a repeated block transfer: the next block information is loaded from the memory (using the linked list address information, from the MDMA_CxLAR)
- If it is the last block that needs to be transferred for the current MDMA channel (MDMA_CxLAR = 0): the channel is disabled and no further MDMA requests will be accepted for this channel.

MDMA Block repeat mode

- The block repeat mode allows to repeat a block transfer, with different “start addresses” for source and destination.
- When the repeat block mode is active (repeat counter is not equal to ‘0’), at the end of the current block transfer, the block parameters is updated:
  - Reload block number of data bytes to transfer (BNDT) value
  - Update block Source/Destination address values according to block repeat Source/ Destination address update mode (BRSUM/BRDUM) configuration
  - Decrement by 1 the repeat counter
- When the repeat block counter reaches 0, the last block is treated as a single block transfer.

life.augmented

The block repeat mode allows repetition of a block transfer, with different start addresses for source and destination.
When the repeat block mode is active (repeat counter is not equal to ‘0’), at the end of the current block transfer, the block parameters are updated (the BNDT value reloaded and SAR/DAR values updated according to BRSUM/BRDUM configuration), and the repeat counter is decremented by 1.
When the repeat block counter reaches 0, the last block is treated as a single block transfer.

MDMA Linked list mode

- The Linked list mode allows to load a new MDMA configuration (CxTCR, CxBNDTR, CxSAR, CxDAR, CxBRUR, CxLAR, CxTBR, CxMAR and CxMDR registers) from the address given in the Channel Link Address register (CLAR)
- Following this operation, the channel is ready to accept new requests, as defined in the block/repeated block modes above, or continue the transfer if the triggering model is set to Complete channel data (TRGM = ’11’),
- The trigger source may be automatically changed, when loading the trigger and bus selection register (TBR) value.
- If triggering mode is set to TRGM = ’11’, the triggering mode and SW request selected must not be changed.

life.augmented

16

The Linked list mode allows loading of a new MDMA configuration from the address given in the CxLAR register. This address must address a memory mapped on the AXI system bus.
Following this operation, the channel is ready to accept new requests, as defined in the block/repeated block modes above, or to continue the transfer if TRGM[1:0] equals ‘11’.
The trigger source can be automatically changed when loading the CxTBR value.
The TRGM and SWRM values must not be changed when TRGM[1:0] equals ‘11’.

MDMA Linked list mode

- The channel configuration (channel link address LAR) must be in the AXI address space.
- LAR value must be aligned on a Double Word address boundary, i.e. LAR[2:0] = 0x0
- Note : beware that the addresses below are not all contiguous.

Register (32-bit word)	Offset from Link Address register	Description
CTCR	0x00	Transfer Configuration register
CBNDTR	0x04	Block number of data register
CSAR	0x08	Source address register
CDAR	0x0C	Destination address register
CBRUR	0x10	Block Repeat address Update register
CLAR	0x14	Link Address register: Next descriptor
CTBR	0x18	Trigger and Bus selection Register
CMAR	0x20	Mask address register
CMDR	0x24	Mask Data register

life.augmented

17

The channel configuration (channel link address LAR) must be in the AXI address space.
The LAR value must be aligned on a Double Word address boundary, i.e. LAR[2:0] = 0x0.

MDMA request mapping

MDMA request	Request Source	description
mdma_str0 – 7	dma1_tcf(0 – 7)	DMA1 stream 0-7 transfer complete flag
mdma_str8 – 15	dma2_tcf(0 – 7)	DMA2 stream 0-7 transfer complete flag
mdma_str16 – 23	dma3_tcf(0 – 7)	DMA3 stream 0-7 transfer complete flag
mdma_str24	FMC	NAND data transfer (Tx or Rx) channel
mdma_str25		NAND ECC/BCH Error channel
mdma_str26	QUADSPI	QUADSPI FIFO threshold
mdma_str27		QUADSPI transfer complete
mdma_str28	CRYP	4-word request from input
mdma_str29		4-word request from output
mdma_str30	HASH	16-word request
mdma_str31	SAES	Input data write request
mdma_str32		Output data read request

life.augmented

18

This table describes the MDMA requests and their mapping to devices. For example, an MDMA channel can be triggered by the end of transfer of the DMA1 stream. In response to this trigger, the MDMA can:

- perform data transfer from SRAM1, 2 or 3 to SYSRAM or DDR
- or reprogram the DMA1 zero flow for a new transfer

To enable the MDMA to efficiently offload the CPU, MDMA channels can be triggered by devices interrupt to automate data exchanges and processing. Examples of peripheral triggers are described in this table.

Interrupts

- Interrupt events for each channel

Interrupt event	Description
CTCIF	MDMA channel transfer complete. It is also set as a result of writing channel enable bit to 0.
BTIF	MDMA block transfer complete
BTRIF	MDMA block repeat transfer complete
TCIF	MDMA buffer transfer complete. It may be used as a debug feature (without interrupt), indicating that (at least) a MDMA buffer transfer had been generated since the last flag reset
TEIF	MDMA transfer error

life.augmented

For each MDMA channel, an interrupt can be produced on the following events:

- Channel transfer completed
- Block transfer completed
- Block transfer repeat completed
- Buffer transfer completed
- Transfer error

MDMA in low-power modes

Mode	Description
Run	Active.
Stop/LP-stop/LPLV-stop	Frozen. MDMA registers content is retained.
Standby	Powered-down. MDMA must be reinitialized after exiting Standby mode.
VBAT	Powered-down. MDMA must be reinitialized after exiting VBAT mode.

life.augmented

The MDMA is active in Run mode. In the various Stop modes, the MDMA is stopped and the contents of the MDMA registers are retained. The MDMA is powered down in Standby and VBAT modes, and the MDMA registers must be reinitialized after exiting these modes.

Thank you

life.augmented