MPAMF_MBWUMON_IDR, MPAM Features Memory Bandwidth Usage Monitoring ID register

The MPAMF_MBWUMON_IDR characteristics are:

Purpose

Indicates the number of memory bandwidth usage monitor instances implemented. This register also indicates several properties of MBWU monitoring, including whether the implementation supports capture, scaling, or long counters.

MPAMF_MBWUMON_IDR_s indicates the number of Secure memory bandwidth usage monitor instances. MPAMF_MBWUMON_IDR_ns indicates the number of Non-secure memory bandwidth usage monitor instances. MPAMF_MBWUMON_IDR_rt indicates the number of Root memory bandwidth usage monitor instances. MPAMF_MBWUMON_IDR_rl indicates the number of Realm memory bandwidth usage monitor instances.

If MPAMF_IDR.HAS_RIS is 1, fields that mention RIS must reflect the properties of the resource instance currently selected by MPAMCFG_PART_SEL.RIS. Fields that do not mention RIS are constant across all resource instances.

Configuration

This register is present only when FEAT_MPAM is implemented, MPAMF_IDR.HAS_MSMON == 1 and MPAMF_MSMON_IDR.MSMON_MBWU == 1. Otherwise, direct accesses to MPAMF_MBWUMON_IDR are res0.

The power and reset domain of each MSC component is specific to that component.

Attributes

MPAMF_MBWUMON_IDR is a 32-bit register.

Field descriptions

31	30	29	28	27	26	25	2
HAS_CAPTURE	HAS_LONG	LWD	HAS_RWBW	HAS_OFLOW_LNKG	HAS_OFSR	HAS_CEVNT_OFLW	HAS_OFL

HAS_CAPTURE, bit [31]

The implementation supports copying an MSMON_MBWU to the corresponding MSMON_MBWU_CAPTURE on a capture event.

HAS_CAPTURE	Meaning
0b0	MSMON_MBWU_CAPTURE is not implemented and there is no support for capture events in the MBWU monitor.
0b1	The MSMON_MBWU_CAPTURE register is implemented and the MBWU monitor supports the capture event behavior.

If RIS is implemented, this field indicates that MBWU monitor capture is implemented for the resource instance selected by MPAMCFG_PART_SEL.RIS.

If MPAMF_MBWUMON_IDR.HAS_LONG is 1, this also indicates that MSMON_MBWU_L_CAPTURE is implemented.

HAS_LONG, bit [30]
When FEAT_MPAMv0p1 is implemented or FEAT_MPAMv1p1 is implemented:

Indicates whether MSMON_MBWU_L is implemented.

If HAS_CAPTURE is 1, indicates whether MSMON_MBWU_L_CAPTURE is implemented.

HAS_LONG	Meaning
0b0	Does not implement MSMON_MBWU_L or MSMON_MBWU_L_CAPTURE.
0b1	Implements MSMON_MBWU_L. If HAS_CAPTURE == 1, MSMON_MBWU_L_CAPTURE is also implemented.

If RIS is implemented, this field indicates that the long MBWU monitor is implemented for the resource instance selected by MPAMCFG_PART_SEL.RIS.

If MPAMF_MBWUMON_IDR.HAS_CAPTURE is 1, this also indicates that MSMON_MBWU_L_CAPTURE is implemented.

Otherwise:

Reserved, res0.

LWD, bit [29]
When FEAT_MPAMv0p1 is implemented or FEAT_MPAMv1p1 is implemented:

Long register VALUE width.

If MPAMF_MBWUMON_IDR.HAS_LONG is 0, MPAMF_MBWUMON_IDR.LWD must also be 0.

LWD	Meaning
0b0	If MPAMF_MBWUMON_IDR.HAS_LONG is 1, MSMON_MBWU_L has 44-bit VALUE field in bits [43:0]. Bits [62:44] are res0. If HAS_LONG is 1 and MPAMF_MBWUMON_IDR.HAS_CAPTURE is 1, MSMON_MBWU_L_CAPTURE also has 44-bit VALUE field in bits [43:0].
0b1	MSMON_MBWU_L has 63-bit VALUE field in bits [62:0]. If MPAMF_MBWUMON_IDR.HAS_CAPTURE == 1, MSMON_MBWU_L_CAPTURE also has 63-bit VALUE field in bits [62:0].

If RIS is implemented, this field indicates the length of the MSMON_MBWU_L.VALUE field implemented for the resource instance selected by MPAMCFG_PART_SEL.RIS.

Otherwise:

Reserved, res0.

HAS_RWBW, bit [28]
When FEAT_MPAMv0p1 is implemented or FEAT_MPAMv1p1 is implemented:

Read/write bandwidth selection is implemented in MSMON_CFG_MBWU_FLT.

HAS_RWBW	Meaning
0b0	Read/write bandwidth selection is not implemented.
0b1	Read/write bandwidth selection is implemented.

If RIS is implemented, this field indicates whether read/write bandwidth collection selection is available in MSMON_CFG_MBWU_FLT for resource instance selected by MPAMCFG_PART_SEL.RIS.

Otherwise:

Reserved, res0.

HAS_OFLOW_LNKG, bit [27]

When FEAT_MPAMv0p1 is implemented or FEAT_MPAMv1p1 is implemented:

Supports MSMON_CFG_MBWU_CTL.OFLOW_LNKG field to control how overflow on an instance affects other monitor instances in this MSC.

HAS_OFLOW_LNKG	Meaning
0b0	Does not support MBWU overflow linkage.
0b1	Supports MBWU overflow linkage and the MSMON_CFG_MBWU_CTL.OFLOW_LNKG field.

If RIS is implemented, this field indicates that MSMON_CFG_MBWU_CTL.OFLOW_LNKG is implemented for the resource instance selected by MPAMCFG_PART_SEL.RIS.

Otherwise:

Reserved, res0.

HAS_OFSR, bit [26]

When FEAT_MPAMv0p1 is implemented or FEAT_MPAMv1p1 is implemented:

The MBWU monitor overflow status bitmap register, MSMON_MBWU_OFSR, is implemented.

HAS_OFSR	Meaning
0b0	MSMON_MBWU_OFSR register is not implemented.
0b1	MSMON_MBWU_OFSR register is implemented.

If RIS is implemented, this field indicates that MBWU monitor overflow status bitmap register is implemented for the resource instance selected by MPAMCFG_PART_SEL.RIS.

Otherwise:

Reserved, res0.

HAS_CEVNT_OFLW, bit [25]

Supports MSMON_CFG_MBWU_CTL.CEVNT_OFLW field which can enable the MBWU monitor instance to perform overflow behaviors on a capture event.

HAS_CEVNT_OFLW	Meaning
0b0	Does not support MSMON_CFG_MBWU_CTL.CEVNT_OFLW.
0b1	Supports MSMON_CFG_MBWU_CTL.CEVNT_OFLW.

If RIS is implemented, this field indicates that MSMON_CFG_MBWU_CTL.CEVNT_OFLW is implemented for the resource instance selected by MPAMCFG_PART_SEL.RIS.

HAS_OFLOW_CAPT, bit [24]

When FEAT_MPAMv0p1 is implemented or FEAT_MPAMv1p1 is implemented:

Supports MSMON_CFG_MBWU_CTL.OFLOW_CAPT field which can enable the MBWU monitor instance to capture the monitor on an overflow.

HAS_OFLOW_CAPT	Meaning
0b0	Does not support MSMON_CFG_MBWU_CTL.OFLOW_CAPT.
0b1	Supports MSMON_CFG_MBWU_CTL.OFLOW_CAPT.

If RIS is implemented, this field indicates that MSMON_CFG_MBWU_CTL.OFLOW_CAPT is implemented for the resource instance selected by MPAMCFG_PART_SEL.RIS.

Otherwise:

Reserved, res0.

Bits [23:21]

Reserved, res0.

SCALE, bits [20:16]

Scaling of MSMON_MBWU.VALUE in bits. If scaling is enabled by MSMON_CFG_MBWU_CTL.SCLEN, the byte count in the VALUE field has been shifted by SCALE bits to the right.

SCALE	Meaning
0b00000	Scaling is not implemented.

`0bxxxxx`	Other values are right shift count when scaling is enabled.

If RIS is implemented, this field indicates the scale value for MSMON_MBWU.VALUE field for the resource instance selected by MPAMCFG_PART_SEL.RIS.

NUM_MON, bits [15:0]

The number of memory bandwidth usage monitor instances implemented. The largest monitor instance selector, MSMON_CFG_MON_SEL.MON_SEL, is NUM_MON minus 1.

If RIS is implemented, this field indicates the number of MBWU monitor instances for MSMON_MBWU.VALUE field for the resource instance selected by MPAMCFG_PART_SEL.RIS.

Accessing MPAMF_MBWUMON_IDR

This register is within the MPAM feature page memory frames. In a system that supports Secure, Non-secure, Root, and Realm memory maps, there must be MPAM feature pages in all four address maps.

MPAMF_MBWUMON_IDR is read-only.

MPAMF_MBWUMON_IDR must be readable from the Non-secure, Secure, Root, and Realm MPAM feature pages.

MPAMF_MBWUMON_IDR is permitted to have the same contents when read from the Secure, Non-secure, Root, and Realm MPAM feature pages unless the register contents are different for the different versions:

- MPAMF_MBWUMON_IDR_s is permitted to have either the same or different contents to MPAMF_MBWUMON_IDR_ns, MPAMF_MBWUMON_IDR_rt, or MPAMF_MBWUMON_IDR_rl.
- MPAMF_MBWUMON_IDR_ns is permitted to have either the same or different contents to MPAMF_MBWUMON_IDR_rt or MPAMF_MBWUMON_IDR_rl.
- MPAMF_MBWUMON_IDR_rt is permitted to have either the same or different contents to MPAMF_MBWUMON_IDR_rl.

There must be separate registers in the Secure (MPAMF_MBWUMON_IDR_s), Non-secure (MPAMF_MBWUMON_IDR_ns), Root (MPAMF_MBWUMON_IDR_rt), and Realm (MPAMF_MBWUMON_IDR_rl) MPAM feature pages.

When MPAMF_IDR.HAS_RIS is 1, MPAMF_MBWUMON_IDR shows the configuration of memory bandwidth monitoring for the bandwidth resource instance selected by MPAMCFG_PART_SEL.RIS. Fields that mention RIS in their field descriptions have values that track the implemented properties of the resource instance. Fields that do not mention RIS are constant across all resource instances.

Access to MPAMF_MBWUMON_IDR is not affected by MSMON_CFG_MON_SEL.RIS.

MPAMF_MBWUMON_IDR can be accessed through the memory-mapped interfaces:

Component	Frame	Offset	Instance
MPAM	MPAMF_BASE_s	`0x0090`	MPAMF_MBWUMON_IDR_s

Accesses on this interface are **RO**.

Component	Frame	Offset	Instance
MPAM	MPAMF_BASE_ns	`0x0090`	MPAMF_MBWUMON_IDR_ns

Accesses on this interface are **RO**.

Component	Frame	Offset	Instance
MPAM	MPAMF_BASE_rt	`0x0090`	MPAMF_MBWUMON_IDR_rt

When FEAT_RME is implemented, accesses on this interface are **RO**.

Component	Frame	Offset	Instance
MPAM	MPAMF_BASE_rl	`0x0090`	MPAMF_MBWUMON_IDR_rl

When FEAT_RME is implemented, accesses on this interface are **RO**.

AArch32 Registers AArch64 Registers AArch32 Instructions AArch64 Instructions Index by Encoding External Registers

28/03/2023 16:02; 72747e43966d6b97dcbd230a1b3f0421d1ea3d94