MPAMF_MSMON_IDR, MPAM Resource Monitoring Identification Register

The MPAMF_MSMON_IDR characteristics are:

Purpose

Indicates which MPAM monitoring features are present on this MSC.

MPAMF_MSMON_IDR_s indicates Secure monitoring features.
MPAMF_MSMON_IDR_ns indicates Non-secure monitoring features.
MPAMF_MSMON_IDR_rt indicates Root monitoring features.
MPAMF_MSMON_IDR_rl indicates Realm monitoring features.

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 and MPAMF_IDR.HAS_MSMON == 1. Otherwise, direct accesses to MPAMF_MSMON_IDR are res0.

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

Attributes

MPAMF_MSMON_IDR is a 32-bit register.

Field descriptions

31	30	29	28	27262524232221201918	
HAS_LOCAL_CAPT_EVNT	NO_HW_OFLW_INTR	HAS_OFLW_MSI	HAS_OFLOW_SR	RES0	MS

HAS_LOCAL_CAPT_EVNT, bit [31]

Has local capture event generator. Indicates whether this MSC has the MPAM local capture event generator and the MSMON_CAPT_EVNT register.

HAS_LOCAL_CAPT_EVNT	Meaning

0b0	Does not support MPAM local capture event generator or MSMON_CAPT_EVNT.
0b1	Supports the MPAM local capture event generator and the MSMON_CAPT_EVNT register.

NO_HW_OFLW_INTR, bit [30]

When FEAT_MPAMv1p1 is implemented:

Does not have hardwired MPAM monitor overflow interrupt.

NO_HW_OFLW_INTR	Meaning
0b0	Supports generating a hardwired interrupt to signal MPAM monitor overflow.
0b1	No support for a hardwired interrupt to signal MPAM monitor overflow.

If this field is 0, the MSC supports generating a hardwired interrupt for monitor overflow events.

If this field is 0 and the HAS_OFLW_MSI field in this register is 1, the MSC supports generating both hardwired interrupts and MSI writes to signal interrupts.

Otherwise:

Reserved, res0.

HAS_OFLW_MSI, bit [29]

When FEAT_MPAMv1p1 is implemented:

Has support for MSI writes to signal MPAM monitor overflow interrupts. These registers are implemented: MSMON_OFLOW_MSI_ADDR_L, MSMON_OFLOW_MSI_ADDR_H, MSMON_OFLOW_MSI_ATTR, MSMON_OFLOW_MSI_DATA and MSMON_OFLOW_MSI_MPAM.

HAS_OFLW_MSI	Meaning
0b0	MSMON_OFLOW_MSI_ADDR_L, MSMON_OFLOW_MSI MSMON_OFLOW_MSI_DATA and MSMON_OFLOW_MS
0b1	MSMON_OFLOW_MSI_ADDR_L, MSMON_OFLOW_MSI MSMON_OFLOW_MSI_DATA and MSMON_OFLOW_MS signal MPAM monitor overflow interrupts.

If MPAMF_MSMON_IDR.NO_HW_OFLW_INTR is 1 and this bit is 0, this MSC does not support monitor overflow interrupts.

Otherwise:

Reserved, res0.

HAS_OFLOW_SR, bit [28]
When FEAT_MPAMv1p1 is implemented:

Has MPAM monitor overflow status register MSMON_OFLOW_SR.

HAS_OFLOW_SR	Meaning
0b0	Does not have MSMON_OFLOW_SR.
0b1	Supports MSMON_OFLOW_SR.

Otherwise:

Reserved, res0.

Bits [27:18]

Reserved, res0.

MSMON_MBWU, bit [17]

Memory bandwidth usage monitoring. Indicates whether MPAM monitoring for Memory Bandwidth Usage by PARTID and PMG is implemented and whether the following bandwidth usage registers are accessible:

- MPAMF_MBWUMON_IDR, MSMON_CFG_MBWU_CTL, MSMON_CFG_MBWU_FLT, MSMON_MBWU.
- The optional MSMON_MBWU_CAPTURE.
- If MPAM v0.1 or MPAM v1.1 is implemented, the optional MSMON_MBWU_L and the optional MSMON_MBWU_L_CAPTURE.

MSMON_MBWU	Meaning

0b0	Does not have monitoring for memory bandwidth usage and does not use the bandwidth usage registers.
0b1	Has monitoring of memory bandwidth usage and uses the bandwidth usage registers.

If RIS is implemented, this field indicates that memory bandwidth usage monitoring is implemented for the resource instance selected by MPAMCFG_PART_SEL.RIS as described in MPAMF_MBWUMON_IDR.

MSMON_CSU, bit [16]

Cache storage usage monitoring. Indicates whether MPAM monitoring of cache storage usage by PARTID and PMG is implemented and the following registers are accessible:

- MPAMF_CSUMON_IDR, MSMON_CFG_CSU_CTL, MSMON_CFG_CSU_FLT, MSMON_CSU.
- The optional MSMON_CSU_CAPTURE.

MSMON_CSU	Meaning
0b0	Does not have monitoring for cache storage usage or the M MSMON_CFG_CSU_CTL, MSMON_CFG_CSU_FLT, MSMO MSMON_CSU_CAPTURE registers.
0b1	Has monitoring of cache storage usage and the MPAMF_C MSMON_CFG_CSU_CTL, MSMON_CFG_CSU_FLT, MSMO MSMON_CSU_CAPTURE registers.

If RIS is implemented, this field indicates that cache storage usage monitoring is implemented for the resource instance selected by MPAMCFG_PART_SEL.RIS as described in MPAMF_CSUMON_IDR.

Bits [15:0]

Reserved, res0.

Accessing MPAMF_MSMON_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_MSMON_IDR is read-only.

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

MPAMF_MSMON_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_MSMON_IDR_s is permitted to have either the same or different contents to MPAMF_MSMON_IDR_ns, MPAMF_MSMON_IDR_rt, or MPAMF_MSMON_IDR_rl.
- MPAMF_MSMON_IDR_ns is permitted to have either the same or different contents to MPAMF_MSMON_IDR_rt or MPAMF_MSMON_IDR_rl.
- MPAMF_MSMON_IDR_rt is permitted to have either the same or different contents to MPAMF_MSMON_IDR_rl.

There must be separate registers in the Secure (MPAMF_MSMON_IDR_s), Non-secure (MPAMF_MSMON_IDR_ns), Root (MPAMF_MSMON_IDR_rt), and Realm (MPAMF_MSMON_IDR_rl) MPAM feature pages.

When MPAMF_IDR.HAS_RIS is 1, MPAMF_MSMON_IDR shows the configuration of memory system monitoring for the 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_MSMON_IDR is not affected by MSMON_CFG_MON_SEL.RIS.

MPAMF_MSMON_IDR can be accessed through the memory-mapped interfaces:

Component	Frame	Offset	Instance
MPAM	MPAMF_BASE_s	`0x0080`	MPAMF_MSMON_IDR_s

Accesses on this interface are **RO**.

Component	Frame	Offset	Instance
MPAM	MPAMF_BASE_ns	`0x0080`	MPAMF_MSMON_IDR_ns

Accesses on this interface are **RO**.

Component	Frame	Offset	Instance
MPAM	MPAMF_BASE_rt	`0x0080`	MPAMF_MSMON_IDR_rt

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

Component	Frame	Offset	Instance

MPAM	MPAMF_BASE_rl	`0x0080`	MPAMF_MSMON_IDR_rl

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