MPAMF_IDR, MPAM Features Identification Register

The MPAMF_IDR characteristics are:

Purpose

Indicates which memory partitioning and monitoring features are present on this MSC.

MPAMF_IDR_s indicates the MPAM features accessed from the Secure MPAM feature page. MPAMF_IDR_ns indicates the MPAM features accessed from the Non-secure MPAM feature page. MPAMF_IDR_rt indicates the MPAM features accessed from the Root MPAM feature page. MPAMF_IDR_rl indicates the MPAM features accessed from the Realm MPAM feature page.

When MPAMF_IDR.HAS_RIS is 1, some fields in this register give information for the resource instance selected by MPAMCFG_PART_SEL.RIS. The description of every field that is affected by MPAMCFG_PART_SEL.RIS has that information within the field description.

Configuration

This register is present only when FEAT_MPAM is implemented. Otherwise, direct accesses to MPAMF_IDR are res0.

MPAMF_IDR is 64-bit register when MPAM v0.1 or v1.1 is implemented.

Otherwise, MPAMF_IDR is a 32-bit register.

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

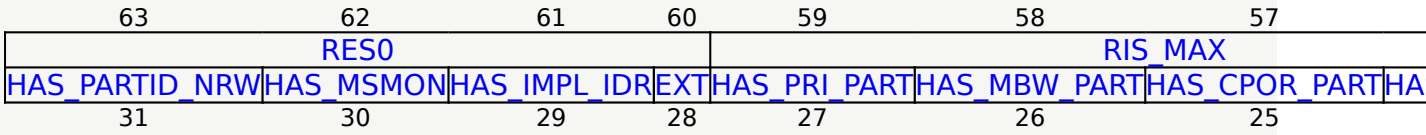
Attributes

MPAMF_IDR is a:

- 64-bit register when FEAT_MPAMv0p1 is implemented or FEAT_MPAMv1p1 is implemented
- 32-bit register otherwise

Field descriptions

When FEAT_MPAMv0p1 is implemented or FEAT_MPAMv1p1 is implemented:



Bits [63:60]

Reserved, res0.

RIS_MAX, bits [59:56]
When MPAMF_IDR.EXT == 1 and MPAMF_IDR.HAS_RIS == 1:

Maximum RIS value supported in MPAMCFG_PART_SEL. Must be `0b0000` if MPAMF_IDR.HAS_RIS == 0.

Otherwise:

Reserved, res0.

Bits [55:44]

Reserved, res0.

HAS_NFU, bit [43]
When FEAT_MPAMv1p1 is implemented or FEAT_MPAMv0p1 is implemented:

Has No Future Use field in MPAMCFG_DIS. Indicates that MPAMCFG_DIS.NFU is implemented.

HAS_NFU	Meaning
`0b0`	MPAMCFG_DIS.NFU is not implemented. A PARTID disabled through access to MPAMCFG_DIS must preserve the control settings of the disabled PARTID.
`0b1`	Implements MPAMCFG_DIS.NFU. A PARTID disabled with NFU as 1 may have its control settings forgotten.

If MPAMF_IDR.HAS_ENDIS is `0b0`, this field must also be `0b0`.

This field must be the same in each instance of this register and for any value in MPAMCFG_PART_SEL.RIS.

Otherwise:

Reserved, res0.

HAS_ENDIS, bit [42]
When FEAT_MPAMv1p1 is implemented or FEAT_MPAMv0p1 is implemented:

Has PARTID enable and disable. Indicates that this MSC supports PARTID disable and enable via MPAMCFG_DIS, MPAMCFG_EN and MPAMCFG_EN_FLAGS registers.

HAS_ENDIS	Meaning
0b0	Does not support PARTID enable and disable functionality, and MPAMCFG_EN, MPAMCFG_DIS and MPAMCFG_EN_FLAGS registers are not implemented.
0b1	Supports PARTID enable and disable through the MPAMCFG_EN, MPAMCFG_DIS and MPAMCFG_EN_FLAGS registers.

All three registers must be implemented when this field is 1, MPAMCFG_EN, MPAMCFG_DIS, and MPAMCFG_EN_FLAGS.

This field must be the same in each instance of this register and for any value in MPAMCFG_PART_SEL.RIS.

Otherwise:

Reserved, res0.

SP4, bit [41]
When FEAT_RME is implemented:

Indicates whether this MSC supports 4 PARTID spaces.

SP4	Meaning
0b0	This MSC supports two PARTID spaces.
0b1	This MSC supports four PARTID spaces.

This field must read the same in each instance of this register and for any value in MPAMCFG_PART_SEL.RIS.

Otherwise:

Reserved, res0.

HAS_ERR_MSI, bit [40]
When MPAMF_IDR.EXT == 1:

Has support for MSI writes to signal MPAM error interrupts. These registers are implemented: MPAMF_ERR_MSI_ADDR_L, MPAMF_ERR_MSI_ADDR_H, MPAMF_ERR_MSI_ATTR, MPAMF_ERR_MSI_DATA, and MPAMF_ERR_MSI_MPAM.

HAS_ERR_MSI	Meaning
0b0	MPAMF_ERR_MSI_ADDR_L, MPAMF_ERR_MSI_ADDR_H, MPAMF_ERR_MSI_DATA, and MPAMF_ERR_MSI_MPAM
0b1	MPAMF_ERR_MSI_ADDR_L, MPAMF_ERR_MSI_ADDR_H, MPAMF_ERR_MSI_DATA, and MPAMF_ERR_MSI_MPAM writes to signal error interrupts.

If MPAMF_IDR.HAS_ESR is 0, this bit must also be 0.

Otherwise:

Reserved, res0.

HAS_ESR, bit [39]
When MPAMF_IDR.EXT == 1:

MPAMF_ESR is implemented.

HAS_ESR	Meaning
0b0	MPAMF_ESR, MPAMF_ECR, and MPAM error handling are not implemented.
0b1	MPAMF_ESR, MPAMF_ECR, and MPAM error handling are implemented.

If an MSC cannot encounter any of the error conditions listed in 'Errors in MSCs' in ArmÂ® Architecture Reference Manual Supplement, Memory System Resource Partitioning and Monitoring (MPAM), for Armv8-A (ARM DDI 0598), both the MPAMF_ESR and MPAMF_ECR must be RAZ/WI.

Otherwise:

Reserved, res0.

HAS_EXTD_ESR, bit [38]
When MPAMF_IDR.EXT == 1:

MPAMF_ESR is 64 bits.

HAS_EXTD_ESR	Meaning
0b0	MPAMF_ESR is 32 bits.
0b1	MPAMF_ESR is 64 bits.

When MPAMF_IDR.HAS_RIS and MPAMF_IDR.HAS_ESR, this field must be 1.

Otherwise:

Reserved, res0.

NO_IMPL_MSMON, bit [37]
When MPAMF_IDR.EXT == 1 and MPAMF_IDR.HAS_IMPL_IDR == 1:

MPAMF_IMPL_IDR defines no implementation defined resource monitors.

NO_IMPL_MSMON	Meaning
0b0	MPAMF_IMPL_IDR defines at least one implementation defined resource monitor.
0b1	MPAMF_IMPL_IDR does not define any implementation defined resource monitors.

If RIS is implemented, this field indicates the presence of implementation defined resource monitors described in MPAMF_IMPL_IDR for the selected resource instance.

Otherwise:

Reserved, res0.

NO_IMPL_PART, bit [36]
When MPAMF_IDR.EXT == 1 and MPAMF_IDR.HAS_IMPL_IDR == 1:

MPAMF_IMPL_IDR defines no implementation defined resource controls.

NO_IMPL_PART	Meaning
0b0	MPAMF_IMPL_IDR defines at least one implementation defined resource control.
0b1	MPAMF_IMPL_IDR does not define any implementation defined resource controls.

If RIS is implemented, this field indicates the presence of implementation defined resource controls described in MPAMF_IMPL_IDR for the selected resource instance.

Otherwise:

Reserved, res0.

Bits [35:33]

Reserved, res0.

HAS_RIS, bit [32]
When MPAMF_IDR.EXT == 1:

Has resource instance selector. Indicates that MPAMCFG_PART_SEL contains the RIS field that selects a resource instance to control.

HAS_RIS	Meaning
0b0	MPAMCFG_PART_SEL does not implement the MPAMCFG_PART_SEL.RIS field or multiple resource instance support.
0b1	MPAMCFG_PART_SEL implements the MPAMCFG_PART_SEL.RIS field and MPAM resource instance numbers up to and including MPAMF_IDR.RIS_MAX.

Otherwise:

Reserved, res0.

HAS_PARTID_NRW, bit [31]

Has PARTID narrowing.

HAS_PARTID_NRW	Meaning
0b0	Does not have MPAMF_PARTID_NRW_IDR, MPAMCFG_INTPARTID, or intPARTID mapping supp
0b1	Supports the MPAMF_PARTID_NRW_IDR, MPAMCFG_INTPARTID registers.

HAS_MSMON, bit [30]

Has resource Monitors. Indicates whether this MSC has MPAM resource monitors.

HAS_MSMON	Meaning
0b0	Does not support MPAM resource monitoring by groups or MPAMF_MSMON_IDR.
0b1	Supports resource monitoring by matching a combination of PARTID and PMG. See MPAMF_MSMON_IDR.

HAS_IMPL_IDR, bit [29]

Has MPAMF_IMPL_IDR. Indicates whether this MSC has the implementation specific MPAM features register, MPAMF_IMPL_IDR.

HAS_IMPL_IDR	Meaning
0b0	Does not have MPAMF_IMPL_IDR.
0b1	Has MPAMF_IMPL_IDR.

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

Extended MPAMF_IDR.

EXT	Meaning

0b0	MPAMF_IDR has no defined bits in [63:32]. The register is effectively 32 bits.
0b1	MPAMF_IDR has bits defined in [63:32]. The register is 64-bits.

Otherwise:

Reserved, res0.

HAS_PRI_PART, bit [27]

Has Priority Partitioning. Indicates that MPAM priority partitioning is implemented and MPAMF_PRI_IDR exists.

HAS_PRI_PART	Meaning
0b0	Does not support priority partitioning or have MPAMF_PRI_IDR.
0b1	Has priority partitioning and MPAMF_PRI_IDR.

If RIS is implemented, this field indicates the presence of priority partitioning resource controls as described in MPAMF_PRI_IDR for the selected resource instance.

HAS_MBW_PART, bit [26]

Has Memory Bandwidth Partitioning. Indicates whether this MSC implements MPAM memory bandwidth partitioning and MPAMF_MBW_IDR.

HAS_MBW_PART	Meaning
0b0	Does not support memory bandwidth partitioning or have MPAMF_MBW_IDR register.
0b1	Has MPAMF_MBW_IDR register.

If RIS is implemented, this field indicates the presence of memory bandwidth partitioning resource controls as described in MPAMF_MBW_IDR for the selected resource instance.

HAS_CPOR_PART, bit [25]

Has Cache Portion Partitioning. Indicates whether this MSC implements MPAM cache portion partitioning and MPAMF_CPOR_IDR.

HAS_CPOR_PART	Meaning
0b0	Does not support cache portion partitioning or have MPAMF_CPOR_IDR or MPAMCFG_CPBM<n> registers.
0b1	Has MPAMF_CPOR_IDR and MPAMCFG_CPBM<n> registers.

If RIS is implemented, this field indicates the presence of cache portion partitioning resource controls as described in MPAMF_CPOR_IDR for the selected resource instance.

HAS_CCAP_PART, bit [24]

Has Cache Capacity Partitioning. Indicates whether this MSC implements MPAM cache capacity partitioning and the MPAMF_CCAP_IDR and MPAMCFG_CMAX registers.

HAS_CCAP_PART	Meaning
0b0	Does not support cache capacity partitioning or have MPAMF_CCAP_IDR and MPAMCFG_CMAX registers.
0b1	Has MPAMF_CCAP_IDR and MPAMCFG_CMAX registers.

If RIS is implemented, this field indicates the presence of cache capacity partitioning resource controls as described in MPAMF_CPOR_IDR for the selected resource instance.

PMG_MAX, bits [23:16]

Maximum supported value of PMG.

The value of this field is permitted to vary between the instances of MPAMF_IDR, each reporting the maximum supported PMG value in the PARTID space associated with that instance.

In MPAMF_IDR_s, this field is permitted to report the maximum PMG value for the Non-secure PARTID space or for the Secure PARTID space. The maximum PMG value for the Secure PARTID space can be read from MPAMF_SIDR.PMG_MAX.

PARTID_MAX, bits [15:0]

Maximum supported value of PARTID.

The value of this field is permitted to vary between the instances of MPAMF_IDR, each reporting the maximum supported PARTID value in the PARTID space associated with that instance.

In MPAMF_IDR_s, this field is permitted to report the maximum PARTID value for the Non-secure PARTID space or for the Secure PARTID space. The maximum PARTID value for the Secure PARTID space can be read from MPAMF_SIDR.PARTID_MAX.

Otherwise:

31	30	29	28	27	26	25	
HAS_PARTID_NRW	HAS_MSMON	HAS_IMPL_IDR	EXT	HAS_PRI_PART	HAS_MBW_PART	HAS_CPOR_PART	HA

HAS_PARTID_NRW, bit [31]

Has PARTID Narrowing.

HAS_PARTID_NRW	Meaning
0b0	Does not have MPAMF_PARTID_NRW_IDR, MPAMCFG_INTPARTID, or intPARTID mapping supp
0b1	Supports the MPAMF_PARTID_NRW_IDR, MPAMCFG_INTPARTID registers.

HAS_MSMON, bit [30]

Has resource Monitors. Indicates whether this MSC has MPAM resource monitors.

HAS_MSMON	Meaning
0b0	Does not support MPAM resource monitoring by groups or MPAMF_MSMON_IDR.
0b1	Supports resource monitoring by matching a combination of PARTID and PMG. See MPAMF_MSMON_IDR.

HAS_IMPL_IDR, bit [29]

Has MPAMF_IMPL_IDR. Indicates whether this MSC has the implementation specific MPAM features register, MPAMF_IMPL_IDR.

HAS_IMPL_IDR	Meaning
0b0	Does not have MPAMF_IMPL_IDR.
0b1	Has MPAMF_IMPL_IDR.

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

Extended MPAMF_IDR.

EXT	Meaning
0b0	MPAMF_IDR has no defined bits in [63:32]. The register is effectively 32 bits.
0b1	MPAMF_IDR has bits defined in [63:32]. The register is 64-bits.

Otherwise:

Reserved, res0.

HAS_PRI_PART, bit [27]

Has Priority Partitioning. Indicates whether this MSC implements MPAM priority partitioning and MPAMF_PRI_IDR.

HAS_PRI_PART	Meaning
0b0	Does not support priority partitioning or have MPAMF_PRI_IDR.
0b1	Has MPAMF_PRI_IDR.

HAS_MBW_PART, bit [26]

Has Memory Bandwidth Partitioning. Indicates whether this MSC implements MPAM memory bandwidth partitioning and MPAMF_MBW_IDR.

HAS_MBW_PART	Meaning

0b0	Does not support memory bandwidth partitioning or have MPAMF_MBW_IDR register.
0b1	Has MPAMF_MBW_IDR register.

HAS_CPOR_PART, bit [25]

Has Cache Portion Partitioning. Indicates whether this MSC implements MPAM cache portion partitioning and MPAMF_CPOR_IDR.

HAS_CPOR_PART	Meaning
0b0	Does not support cache portion partitioning or have MPAMF_CPOR_IDR or MPAMCFG_CPBM<n> registers.
0b1	Has MPAMF_CPOR_IDR and MPAMCFG_CPBM<n> registers.

HAS_CCAP_PART, bit [24]

Has Cache Capacity Partitioning. Indicates whether this MSC implements MPAM cache capacity partitioning and the MPAMF_CCAP_IDR and MPAMCFG_CMAX registers.

HAS_CCAP_PART	Meaning
0b0	Does not support cache capacity partitioning or have MPAMF_CCAP_IDR and MPAMCFG_CMAX registers.
0b1	Has MPAMF_CCAP_IDR and MPAMCFG_CMAX registers.

PMG_MAX, bits [23:16]

Maximum supported value of PMG.

The value of this field is permitted to vary between the instances of MPAMF_IDR, each reporting the maximum supported PMG value in the PARTID space associated with that instance.

In MPAMF_IDR_s this field is permitted to report the maximum PMG value for the Non-secure PARTID space or for the Secure PARTID space. The maximum PMG value for the Secure PARTID space can be read from MPAMF_SIDR.PMG_MAX.

PARTID_MAX, bits [15:0]

Maximum supported value of PARTID.

The value of this field is permitted to vary between the instances of MPAMF_IDR, each reporting the maximum supported PARTID value in the PARTID space associated with that instance.

In MPAMF_IDR_s this field is permitted to report the maximum PARTID value for the Non-secure PARTID space or for the Secure PARTID space. The maximum PARTID value for the Secure PARTID space can be read from MPAMF_SIDR.PARTID_MAX.

Accessing MPAMF_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_IDR is read-only.

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

MPAMF_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_IDR_s is permitted to have either the same or different contents to MPAMF_IDR_ns, MPAMF_IDR_rt, or MPAMF_IDR_rl.
- MPAMF_IDR_ns is permitted to have either the same or different contents to MPAMF_IDR_rt or MPAMF_IDR_rl.
- MPAMF_IDR_rt is permitted to have either the same or different contents to MPAMF_IDR_rl.

There must be separate registers in the Secure (MPAMF_IDR_s), Non-secure (MPAMF_IDR_ns), Root (MPAMF_IDR_rt), and Realm (MPAMF_IDR_rl) MPAM feature pages.

When MPAMF_IDR.HAS_RIS is 1, MPAMF_IDR shows the configuration of MSC MPAM 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.

MPAMF_IDR can be accessed through the memory-mapped interfaces:

Component	Frame	Offset	Instance
MPAM	MPAMF_BASE_s	0x0000	MPAMF_IDR_s

Accesses on this interface are **RO**.

Component	Frame	Offset	Instance
MPAM	MPAMF_BASE_ns	0x0000	MPAMF_IDR_ns

Accesses on this interface are **RO**.

Component	Frame	Offset	Instance
MPAM	MPAMF_BASE_rt	0x0000	MPAMF_IDR_rt

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

Component	Frame	Offset	Instance
MPAM	MPAMF_BASE_rl	0x0000	MPAMF_IDR_rl

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

28/03/2023 16:02; 72747e43966d6b97dcbd230a1b3f0421d1ea3d94