ID_AA64ISAR0_EL1, AArch64 Instruction Set Attribute Register 0

The ID_AA64ISAR0_EL1 characteristics are:

Purpose

Provides information about the instructions implemented in AArch64 state.

For general information about the interpretation of the ID registers, see 'Principles of the ID scheme for fields in ID registers'.

Configuration

There are no configuration notes.

Attributes

ID_AA64ISAR0_EL1 is a 64-bit register.

Field descriptions

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
RNDR				TLB				TS				FHM				DP				SM4				SM3				SHA3			
RDM				TME				Atomic				CRC32				SHA2				SHA1				AES				RES0			
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

RNDR, bits [63:60]

Indicates support for Random Number instructions in AArch64 state.

When FEAT_RNG_TRAP is implemented, the value returned by a direct read of ID_AA64ISAR0_EL1.RNDR is further controlled by the value of SCR_EL3.TRNDR.

Defined values are:

RNDR	Meaning
0b0000	No Random Number instructions are implemented.
0b0001	RNDR and RNDRRS registers are implemented.

All other values are reserved.

FEAT_RNG implements the functionality identified by the value `0b0001`.

From Armv8.5, the permitted values are `0b0000` and `0b0001`.

TLB, bits [59:56]

Indicates support for Outer Shareable and TLB range maintenance instructions. Defined values are:

TLB	Meaning
`0b0000`	Outer Shareable and TLB range maintenance instructions are not implemented.
`0b0001`	Outer Shareable TLB maintenance instructions are implemented.
`0b0010`	Outer Shareable and TLB range maintenance instructions are implemented.

All other values are reserved.

FEAT_TLBIOS implements the functionality identified by the values `0b0001` and `0b0010`.

FEAT_TLBIRANGE implements the functionality identified by the value `0b0010`.

From Armv8.4, the only permitted value is `0b0010`.

TS, bits [55:52]

Indicates support for flag manipulation instructions. Defined values are:

TS	Meaning
`0b0000`	No flag manipulation instructions are implemented.
`0b0001`	CFINV, RMIF, SETF16, and SETF8 instructions are implemented.
`0b0010`	CFINV, RMIF, SETF16, SETF8, AXFLAG, and XAFLAG instructions are implemented.

All other values are reserved.

FEAT_FlagM implements the functionality identified by the value `0b0001`.

FEAT_FlagM2 implements the functionality identified by the value `0b0010`.

In Armv8.2, the permitted values are `0b0000` and `0b0001`.

In Armv8.4, the only permitted value is `0b0001`.

From Armv8.5, the only permitted value is `0b0010`.

FHM, bits [51:48]

Indicates support for FMLAL and FMLSL instructions. Defined values are:

FHM	Meaning
`0b0000`	FMLAL and FMLSL instructions are not implemented.
`0b0001`	FMLAL and FMLSL instructions are implemented.

All other values are reserved.

FEAT_FHM implements the functionality identified by the value `0b0001`.

From Armv8.2, the permitted values are `0b0000` and `0b0001`.

DP, bits [47:44]

Indicates support for Dot Product instructions in AArch64 state. Defined values are:

DP	Meaning
`0b0000`	No Dot Product instructions implemented.
`0b0001`	UDOT and SDOT instructions implemented.

All other values are reserved.

FEAT_DotProd implements the functionality identified by the value `0b0001`.

From Armv8.2, the permitted values are `0b0000` and `0b0001`.

SM4, bits [43:40]

Indicates support for SM4 instructions in AArch64 state. Defined values are:

SM4	Meaning

0b0000	No SM4 instructions implemented.
0b0001	SM4E and SM4EKEY instructions implemented.

All other values are reserved.

If FEAT_SM4 is not implemented, the value 0b0001 is reserved.

From Armv8.2, the permitted values are 0b0000 and 0b0001.

This field must have the same value as ID_AA64ISAR0_EL1.SM3.

SM3, bits [39:36]

Indicates support for SM3 instructions in AArch64 state. Defined values are:

SM3	Meaning
0b0000	No SM3 instructions implemented.
0b0001	SM3SS1, SM3TT1A, SM3TT1B, SM3TT2A, SM3TT2B, SM3PARTW1, and SM3PARTW2 instructions implemented.

All other values are reserved.

If FEAT_SM3 is not implemented, the value 0b0001 is reserved.

FEAT_SM3 implements the functionality identified by the value 0b0001.

From Armv8.2, the permitted values are 0b0000 and 0b0001.

This field must have the same value as ID_AA64ISAR0_EL1.SM4.

SHA3, bits [35:32]

Indicates support for SHA3 instructions in AArch64 state. Defined values are:

SHA3	Meaning
0b0000	No SHA3 instructions implemented.
0b0001	EOR3, RAX1, XAR, and BCAX instructions implemented.

All other values are reserved.

If FEAT_SHA3 is not implemented, the value 0b0001 is reserved.

FEAT_SHA3 implements the functionality identified by the value 0b0001.

From Armv8.2, the permitted values are 0b0000 and 0b0001.

If the value of ID_AA64ISAR0_EL1.SHA1 is 0b0000, this field must have the value 0b0000.

If the value of this field is 0b0001, ID_AA64ISAR0_EL1.SHA2 must have the value 0b0010.

RDM, bits [31:28]

Indicates support for SQRDMLAH and SQRDMLSH instructions in AArch64 state. Defined values are:

RDM	Meaning
0b0000	No RDMA instructions implemented.
0b0001	SQRDMLAH and SQRDMLSH instructions implemented.

All other values are reserved.

FEAT_RDM implements the functionality identified by the value 0b0001.

From Armv8.1, the only permitted value is 0b0001.

TME, bits [27:24]

Indicates support for TME instructions. Defined values are:

TME	Meaning
0b0000	TME instructions are not implemented.
0b0001	TCANCEL, TCOMMIT, TSTART, and TTEST instructions are implemented.

All other values are reserved.

Accessing this field has the following behavior:

- Access is **RAZ/WI** if all of the following are true:
  - PSTATE.EL IN {EL2, EL1}
  - SCR_EL3.TME == 0
- Access is **RAZ/WI** if all of the following are true:
  - PSTATE.EL == EL1
  - EL2Enabled()
  - HCR_EL2.TME == 0

- Otherwise, access to this field is **RO**.

Atomic, bits [23:20]

Indicates support for Atomic instructions in AArch64 state. Defined values are:

Atomic	Meaning	Applies when
0b0000	No Atomic instructions implemented.	
0b0010	LDADD, LDCLR, LDEOR, LDSET, LDSMAX, LDSMIN, LDUMAX, LDUMIN, CAS, CASP, and SWP instructions implemented.	
0b0011	As for 0b0010, plus 128-bit instructions LDCLRP, LDSETP and SWPP.	When FEAT_LSE128 is implemented

All other values are reserved.

FEAT_LSE implements the functionality identified by the value 0b0010.

FEAT_LSE128 implements the functionality identified by the value 0b0011.

From Armv8.1, the value 0b0000 is not permitted.

CRC32, bits [19:16]

Indicates support for CRC32 instructions in AArch64 state. Defined values are:

CRC32	Meaning
0b0000	CRC32 instructions are not implemented.

0b0001	CRC32B, CRC32H, CRC32W, CRC32X, CRC32CB, CRC32CH, CRC32CW, and CRC32CX instructions are implemented.

All other values are reserved.

FEAT_CRC32 implements the functionality identified by the value 0b0001.

In Armv8.0, the permitted values are 0b0000 and 0b0001.

From Armv8.1, the only permitted value is 0b0001.

SHA2, bits [15:12]

Indicates support for SHA2 instructions in AArch64 state. Defined values are:

SHA2	Meaning
0b0000	No SHA2 instructions implemented.
0b0001	Implements instructions: SHA256H, SHA256H2, SHA256SU0, and SHA256SU1.
0b0010	Implements instructions: • SHA256H, SHA256H2, SHA256SU0, and SHA256SU1. • SHA512H, SHA512H2, SHA512SU0, and SHA512SU1.

All other values are reserved.

FEAT_SHA256 implements the functionality identified by the value 0b0001.

FEAT_SHA512 implements the functionality identified by the value 0b0010.

In Armv8, the permitted values are 0b0000 and 0b0001.

From Armv8.2, the permitted values are 0b0000, 0b0001, and 0b0010.

If the value of ID_AA64ISAR0_EL1.SHA1 is 0b0000, this field must have the value 0b0000.

If the value of this field is `0b0010`, ID_AA64ISAR0_EL1.SHA3 must have the value `0b0001`.

SHA1, bits [11:8]

Indicates support for SHA1 instructions in AArch64 state. Defined values are:

SHA1	Meaning
`0b0000`	No SHA1 instructions implemented.
`0b0001`	SHA1C, SHA1P, SHA1M, SHA1H, SHA1SU0, and SHA1SU1 instructions implemented.

All other values are reserved.

FEAT_SHA1 implements the functionality identified by the value `0b0001`.

From Armv8, the permitted values are `0b0000` and `0b0001`.

If the value of ID_AA64ISAR0_EL1.SHA2 is `0b0000`, this field must have the value `0b0000`.

AES, bits [7:4]

Indicates support for AES instructions in AArch64 state. Defined values are:

AES	Meaning
`0b0000`	No AES instructions implemented.
`0b0001`	AESE, AESD, AESMC, and AESIMC instructions implemented.
`0b0010`	As for `0b0001`, plus PMULL and PMULL2 instructions operating on 64-bit source elements.

FEAT_AES implements the functionality identified by the value `0b0001`.

FEAT_PMULL implements the functionality identified by the value `0b0010`.

All other values are reserved.

From Armv8, the permitted values are `0b0000` and `0b0010`.

Bits [3:0]

Reserved, res0.

Accessing ID_AA64ISAR0_EL1

Accesses to this register use the following encodings in the System register encoding space:

MRS <Xt>, ID_AA64ISAR0_EL1

op0	op1	CRn	CRm	op2
0b11	0b000	0b0000	0b0110	0b000

```
if PSTATE.EL == EL0 then
    if IsFeatureImplemented(FEAT_IDST) then
        if EL2Enabled() && HCR_EL2.TGE == '1' then
            AArch64.SystemAccessTrap(EL2, 0x18);
        else
            AArch64.SystemAccessTrap(EL1, 0x18);
    else
        UNDEFINED;
elsif PSTATE.EL == EL1 then
    if EL2Enabled() && HCR_EL2.TID3 == '1' then
        AArch64.SystemAccessTrap(EL2, 0x18);
    else
        X[t, 64] = ID_AA64ISAR0_EL1;
elsif PSTATE.EL == EL2 then
    X[t, 64] = ID_AA64ISAR0_EL1;
elsif PSTATE.EL == EL3 then
    X[t, 64] = ID_AA64ISAR0_EL1;
```


28/03/2023 16:01; 72747e43966d6b97dcbd230a1b3f0421d1ea3d94