EDSCR, External Debug Status and Control Register

The EDSCR characteristics are:

Purpose

Main control register for the debug implementation.

Configuration

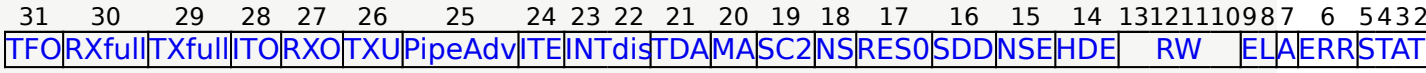
External register EDSCR bits [30:29] are architecturally mapped to AArch64 System register MDCCSR_EL0[30:29].

EDSCR is in the Core power domain.

Attributes

EDSCR is a 32-bit register.

Field descriptions



TFO, bit [31]
When FEAT_TRF is implemented:

Trace Filter Override. Overrides the Trace Filter controls allowing the external debugger to trace any visible Exception level.

TFO	Meaning
0b0	Trace Filter controls are not affected.
0b1	Trace Filter controls in TRFCR_EL1 and TRFCR_EL2 are ignored. Trace Filter controls TRFCR and HTRFCR are ignored.

When OSLSR_EL1.OSLK is 1, this bit can be indirectly read and written through the following System registers:

- MDSCR_EL1.
- DBGDSCRext.

This bit is ignored by the PE when any of the following is true:

- ExternalSecureNoninvasiveDebugEnabled() is FALSE and the Effective value of MDCR_EL3.STE is 1.
- FEAT_RME is implemented, ExternalRealmNoninvasiveDebugEnabled() is FALSE, and the Effective value of MDCR_EL3.RLTE is 1.

The reset behavior of this field is:

- On a Cold reset, this field resets to `0`.

Otherwise:

Reserved, res0.

RXfull, bit [30]

DTRRX full.

When OSLSR_EL1.OSLK is 1, this bit can be indirectly read and written through the following System registers:

- MDSCR_EL1.
- DBGDSCRext.

The reset behavior of this field is:

- On a Cold reset, this field resets to `0`.

Access to this field is **RO**.

TXfull, bit [29]

DTRTX full.

When OSLSR_EL1.OSLK is 1, this bit can be indirectly read and written through the following System registers:

- MDSCR_EL1.
- DBGDSCRext.

The reset behavior of this field is:

- On a Cold reset, this field resets to `0`.

Access to this field is **RO**.

ITO, bit [28]

ITR overrun. Set to 0 on entry to Debug state.

Accessing this field has the following behavior:

- When the PE is in Non-debug state, access to this field is **UNKNOWN/WI**.
- Otherwise, access to this field is **RO**.

RXO, bit [27]

DTRRX overrun.

When OSLSR_EL1.OSLK is 1, this bit can be indirectly read and written through the following System registers:

- MDSCR_EL1.
- DBGDSCRext.

The reset behavior of this field is:

- On a Cold reset, this field resets to `0`.

Access to this field is **RO**.

TXU, bit [26]

DTRTX underrun.

When OSLSR_EL1.OSLK is 1, this bit can be indirectly read and written through the following System registers:

- MDSCR_EL1.
- DBGDSCRext.

The reset behavior of this field is:

- On a Cold reset, this field resets to `0`.

Access to this field is **RO**.

PipeAdv, bit [25]

Pipeline Advance. Indicates that software execution is progressing.

PipeAdv	Meaning
`0b0`	No progress has been made by the PE since the last time this field was cleared to zero by writing 1 to EDRCR.CSPA.
`0b1`	Progress has been made by the PE since the last time this field was cleared to zero by writing 1 to EDRCR.CSPA.

The architecture does not define precisely when this field is set to 1. It requires only that this happen periodically in Non-debug state to indicate that software execution is progressing. For example, a PE might set this field to 1 each time the PE retires one or more instructions, or at periodic intervals during the progression of an instruction.

When FEAT_MOPS is implemented, `CPY`, `CPYF`, `SET`, and `SETG` are examples of instructions that periodically make forward progress.

The reset behavior of this field is:

- On a Warm reset, this field resets to an architecturally unknown value.

Access to this field is **RO**.

ITE, bit [24]

ITR empty.

Accessing this field has the following behavior:

- When the PE is in Non-debug state, access to this field is **UNKNOWN/WI**.
- Otherwise, access to this field is **RO**.

INTdis, bits [23:22]
When FEAT_RME is implemented:

Interrupt disable. Disables taking interrupts in Non-debug state.

INTdis	Meaning
`0b00`	This bit has no effect on the masking of interrupts.
`0b01`	If ExternalInvasiveDebugEnabled() is TRUE, then all interrupts taken to Non-secure state are masked. If ExternalSecureInvasiveDebugEnabled() is TRUE, then all interrupts taken to Secure state are masked. If ExternalRootInvasiveDebugEnabled() is TRUE, then all interrupts taken to Root state are masked. If ExternalRealmInvasiveDebugEnabled() is TRUE, then all interrupts taken to Realm state are masked.

Note

All interrupts includes virtual and SError interrupts.

When OSLSR_EL1.OSLK is 1, this field can be indirectly read and written through the following System registers:

- MDSCR_EL1.
- DBGDSCRext.

The Effective value of this field is `0b00` when ExternalInvasiveDebugEnabled() is FALSE.

When FEAT_RME is implemented, bit[23] of this register is res0.

The reset behavior of this field is:

- On a Cold reset, this field resets to `0`.

When FEAT_Debugv8p4 is implemented:

Interrupt disable. Disables taking interrupts in Non-debug state.

INTdis	Meaning
`0b00`	Masking of interrupts is controlled by PSTATE and interrupt routing controls.
`0b01`	If ExternalInvasiveDebugEnabled() is TRUE, then all interrupts taken to Non-secure state are masked. If ExternalSecureInvasiveDebugEnabled() is TRUE, then all interrupts taken to Secure state are masked.

Note

All interrupts includes virtual and SError interrupts.

When OSLSR_EL1.OSLK is 1, this field can be indirectly read and written through the following System registers:

- MDSCR_EL1.
- DBGDSCRext.

The Effective value of this field is `0b00` when ExternalInvasiveDebugEnabled() is FALSE.

When FEAT_Debugv8p4 is implemented, bit[23] of this register is res0.

The reset behavior of this field is:

- On a Cold reset, this field resets to `0`.

Otherwise:

Interrupt disable. Disables taking interrupts in Non-debug state.

INTdis	Meaning
`0b00`	Masking of interrupts is controlled by PSTATE and interrupt routing controls.
`0b01`	If ExternalInvasiveDebugEnabled() is TRUE, then all interrupts taken to Non-secure EL1 are masked.
`0b10`	If ExternalInvasiveDebugEnabled() is TRUE, then all interrupts taken to Non-secure state are masked. If ExternalSecureInvasiveDebugEnabled() is TRUE, then all interrupts taken to Secure EL1 are masked.
`0b11`	If ExternalInvasiveDebugEnabled() is TRUE, then all interrupts taken to Non-secure state are masked. If ExternalSecureInvasiveDebugEnabled() is TRUE, then all interrupts taken to Secure state are masked.

Note

All interrupts includes virtual and SError interrupts.

When OSLSR_EL1.OSLK is 1, this field can be indirectly read and written through the following System registers:

- MDSCR_EL1.
- DBGDSCRext.

The Effective value of this field is `0b00` when ExternalInvasiveDebugEnabled() is FALSE.

Support for the values `0b01` and `0b10` is implementation defined. If these values are not supported, they are reserved. If programmed with a reserved value, the PE behaves as if INTdis has been programmed with a defined value, other than for a direct read of EDSCR, and the value returned by a read of EDSCR.INTdis is unknown.

The reset behavior of this field is:

- On a Cold reset, this field resets to `0`.

TDA, bit [21]

Traps accesses to the following debug System registers:

- AArch64: DBGBCR<n>_EL1, DBGBVR<n>_EL1, DBGWCR<n>_EL1, DBGWVR<n>_EL1.
- AArch32: DBGBCR<n>, DBGBVR<n>, DBGBXVR<n>, DBGWCR<n>, DBGWVR<n>.

TDA	Meaning
`0b0`	Accesses to debug System registers do not generate a Software Access Debug event.
`0b1`	Accesses to debug System registers generate a Software Access Debug event, if OSLSR_EL1.OSLK is 0 and if halting is allowed.

When OSLSR_EL1.OSLK is 1, this bit can be indirectly read and written through the following System registers:

- MDSCR_EL1.
- DBGDSCRext.

The reset behavior of this field is:

- On a Cold reset, this field resets to `0`.

MA, bit [20]

Memory access mode. Controls the use of memory-access mode for accessing ITR and the DCC. This bit is ignored if in Non-debug state and set to zero on entry to Debug state.

Possible values of this field are:

MA	Meaning
`0b0`	Normal access mode.
`0b1`	Memory access mode.

The reset behavior of this field is:

- On a Cold reset, this field resets to `0`.

SC2, bit [19]
When FEAT_PCSRv8 is implemented, (FEAT_VHE is implemented or FEAT_Debugv8p2 is implemented) and FEAT_PCSRv8p2 is not implemented:

Sample CONTEXTIDR_EL2. Controls whether the PC Sample-based Profiling Extension samples CONTEXTIDR_EL2 or VTTBR_EL2.VMID.

SC2	Meaning
0b0	Sample VTTBR_EL2.VMID.
0b1	Sample CONTEXTIDR_EL2.

When OSLSR_EL1.OSLK is 1, this bit can be indirectly read and written through the following System registers:

- MDSCR_EL1.
- DBGDSCRext.

The reset behavior of this field is:

- On a Cold reset, this field resets to 0.

Otherwise:

Reserved, res0.

NS, bit [18]
When FEAT_RME is implemented:

Non-secure status. Together with the NSE field, gives the current Security state:

NSE	NS	Meaning
0b0	0b0	When Secure state is implemented, Secure. Otherwise reserved.
0b0	0b1	Non-secure.
0b1	0b0	Root.
0b1	0b1	Realm.

Accessing this field has the following behavior:

- When the PE is in Non-debug state, access to this field is **UNKNOWN/WI**.
- Otherwise, access to this field is **RO**.

Otherwise:

Non-secure status. In Debug state, gives the current Security state:

NS	Meaning
0b0	Secure state.
0b1	Non-secure state.

Accessing this field has the following behavior:

- When the PE is in Non-debug state, access to this field is **UNKNOWN/WI**.
- Otherwise, access to this field is **RO**.

Bit [17]

Reserved, res0.

SDD, bit [16]
When FEAT_RME is implemented:

EL3 debug disabled.

On entry to Debug state:

- If entering from EL3, SDD is set to 0.
- Otherwise, SDD is set to the inverse of `ExternalRootInvasiveDebugEnabled`().

In Debug state, the value of SDD does not change, even if `ExternalRootInvasiveDebugEnabled`() changes.

In Non-debug state, SDD returns the inverse of `ExternalRootInvasiveDebugEnabled`().

Access to this field is **RO**.

Otherwise:

Secure debug disabled.

On entry to Debug state:

- If entering in Secure state, SDD is set to 0.
- If entering in Non-secure state, SDD is set to the inverse of `ExternalSecureInvasiveDebugEnabled`().

In Debug state, the value of the SDD bit does not change, even if `ExternalSecureInvasiveDebugEnabled`() changes.

In Non-debug state:

- SDD returns the inverse of `ExternalSecureInvasiveDebugEnabled`(). If the authentication signals that control `ExternalSecureInvasiveDebugEnabled`()

change, a context synchronization event is required to guarantee their effect.
- This bit is unaffected by the Security state of the PE.

If EL3 is not implemented and the implementation is Non-secure, this bit is res1.

Access to this field is **RO**.

NSE, bit [15]
When FEAT_RME is implemented:

Together with the NS field, this field gives the current Security state.

For a description of the values derived by evaluating NS and NSE together, see EDSCR.NS.

In Non-debug state, this bit is unknown.

Access to this field is **RO**.

Otherwise:

Reserved, res0.

HDE, bit [14]

Halting debug enable.

HDE	Meaning
`0b0`	Halting disabled for Breakpoint, Watchpoint and Halt Instruction debug events.
`0b1`	Halting enabled for Breakpoint, Watchpoint and Halt Instruction debug events.

When OSLSR_EL1.OSLK is 1, this bit can be indirectly read and written through the following System registers:

- MDSCR_EL1.
- DBGDSCRext.

The reset behavior of this field is:

- On a Cold reset, this field resets to `0`.

RW, bits [13:10]

Exception level Execution state status. In Debug state, each bit gives the current Execution state of each Exception level.

RW	Meaning	Applies when
0b1111	Any of the following: • The PE is in Non-debug state. • The PE is at EL0 using AArch64. • The PE is not at EL0, and EL1, EL2, and EL3 are using AArch64.	
0b1110	The PE is in Debug state at EL0. EL0 is using AArch32. EL1, EL2, and EL3 are using AArch64.	When AArch32 is supported
0b110x	The PE is in Debug state. EL0 and EL1 are using AArch32. EL2 is enabled in the current Security state and is using AArch64. If implemented, EL3 is using AArch64.	When AArch32 is supported and EL2 is implemented

0b10xx	The PE is in Debug state. EL0 and EL1 are using AArch32. EL2 is not implemented, disabled in the current Security state, or using AArch32. EL3 is using AArch64.	When AArch32 is supported and EL3 is implemented
0b0xxx	The PE is in Debug state. All Exception levels are using AArch32.	When AArch32 is supported

Accessing this field has the following behavior:

- When the PE is in Non-debug state, access to this field is **RAO/WI**.
- Otherwise, access to this field is **RO**.

EL, bits [9:8]

Exception level. In Debug state, gives the current Exception level of the PE.

Accessing this field has the following behavior:

- When the PE is in Non-debug state, access to this field is **RAZ/WI**.
- Otherwise, access to this field is **RO**.

A, bit [7]

SError interrupt pending. In Debug state, indicates whether an SError interrupt is pending:

- If HCR_EL2.{AMO, TGE} = {1, 0}, EL2 is enabled in the current Security state, and the PE is executing at EL0 or EL1, a virtual SError interrupt.
- Otherwise, a physical SError interrupt.

A	Meaning
0b0	No SError interrupt pending.
0b1	SError interrupt pending.

A debugger can read EDSCR to check whether an SError interrupt is pending without having to execute further instructions. A pending SError might indicate data from target memory is corrupted.

Accessing this field has the following behavior:

- When the PE is in Non-debug state, access to this field is **UNKNOWN/WI**.
- Otherwise, access to this field is **RO**.

ERR, bit [6]

Cumulative error flag. This bit is set to 1 following exceptions in Debug state and on any signaled overrun or underrun on the DTR or EDITR.

When OSLSR_EL1.OSLK is 1, this bit can be indirectly read and written through the following System registers:

- MDSCR_EL1.
- DBGDSCRext.

The reset behavior of this field is:

- On a Cold reset, this field resets to `0`.

Access to this field is **RO**.

STATUS, bits [5:0]

Debug status flags.

STATUS	Meaning
`0b000001`	PE is restarting, exiting Debug state.
`0b000010`	PE is in Non-debug state.
`0b000111`	Breakpoint.
`0b010011`	External debug request.
`0b011011`	Halting step, normal.
`0b011111`	Halting step, exclusive.
`0b100011`	OS Unlock Catch.
`0b100111`	Reset Catch.
`0b101011`	Watchpoint.
`0b101111`	HLT instruction.
`0b110011`	Software access to debug register.
`0b110111`	Exception Catch.
`0b111011`	Halting step, no syndrome.

All other values of STATUS are reserved.

Access to this field is **RO**.

Accessing EDSCR

EDSCR can be accessed through the external debug interface:

Component	Offset	Instance
Debug	0x088	EDSCR

This interface is accessible as follows:

- When IsCorePowered(), !DoubleLockStatus(), !OSLockStatus() and SoftwareLockStatus(), accesses to this register are **RO**.
- When IsCorePowered(), !DoubleLockStatus(), !OSLockStatus() and ! SoftwareLockStatus(), accesses to this register are **RW**.
- Otherwise, accesses to this register generate an error response.

28/03/2023 16:02; 72747e43966d6b97dcbd230a1b3f0421d1ea3d94