S3_<op1>_<Cn>_<Cm>_<op2>, IMPLEMENTATION DEFINED registers

The S3_<op1>_<Cn>_<Cm>_<op2> characteristics are:

Purpose

This area of the instruction set space is reserved for implementation defined registers.

Configuration

Each register in this space is a 128-bit register that can also be accessed as a 64-bit value. If it is accessed as a 64-bit register, accesses read and write bits [63:0] and do not modify bits [127:64].

Attributes

S3_<op1>_<Cn>_<Cm>_<op2> is a:

- 128-bit register when FEAT_SYSREG128 is implemented
- 64-bit register otherwise

Field descriptions

When FEAT_SYSREG128 is implemented:

127	126	125	124	123	122	121	120	119	118	117	116	115	114	113	112	111	110	109	108	107	106	105	104	103	102	101	100	99	98	97	96
IMPLEMENTATION DEFINED																															

95	94	93	92	91	90	89	88	87	86	85	84	83	82	81	80	79	78	77	76	75	74	73	72	71	70	69	68	67	66	65	64
IMPLEMENTATION DEFINED																															

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
IMPLEMENTATION DEFINED																															

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
IMPLEMENTATION DEFINED																															

IMPLEMENTATION DEFINED, bits [127:0]

implementation defined.

Otherwise:

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
IMPLEMENTATION DEFINED																															
IMPLEMENTATION DEFINED																															
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

IMPLEMENTATION DEFINED, bits [63:0]

implementation defined.

Accessing S3_<op1>_<Cn>_<Cm>_<op2>

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

MRS <Xt>, S3_<op1>_C<Cn>_C<Cm>_<op2>

op0	op1	CRn	CRm	op2
0b11	op1[2:0]	0b1x11	Cm[3:0]	op2[2:0]

```
if PSTATE.EL == EL0 then
    if !(EL2Enabled() && HCR_EL2.<E2H,TGE> == '11')
&& SCTLR_EL1.TIDCP == '1' then
        if EL2Enabled() && HCR_EL2.TGE == '1' then
            AArch64.SystemAccessTrap(EL2, 0x18);
        else
            AArch64.SystemAccessTrap(EL1, 0x18);
    elsif EL2Enabled() && HCR_EL2.<E2H,TGE> == '11'
&& SCTLR_EL2.TIDCP == '1' then
        AArch64.SystemAccessTrap(EL2, 0x18);
    else
        AArch64.ImpDefSysRegRead(op0, op1, CRn, CRm,
op2, t);
elsif PSTATE.EL == EL1 then
    if EL2Enabled() && HCR_EL2.TIDCP == '1' then
        AArch64.SystemAccessTrap(EL2, 0x18);
    else
        AArch64.ImpDefSysRegRead(op0, op1, CRn, CRm,
op2, t);
elsif PSTATE.EL == EL2 then
    AArch64.ImpDefSysRegRead(op0, op1, CRn, CRm,
op2, t);
elsif PSTATE.EL == EL3 then
    AArch64.ImpDefSysRegRead(op0, op1, CRn, CRm,
op2, t);
```

MSR S3_<op1>_C<Cn>_C<Cm>_<op2>, <Xt>

op0	op1	CRn	CRm	op2
0b11	op1[2:0]	0b1x11	Cm[3:0]	op2[2:0]

```
if PSTATE.EL == EL0 then
```

```
    if !(EL2Enabled() && HCR_EL2.<E2H,TGE> == '11')
&& SCTLR_EL1.TIDCP == '1' then
        if EL2Enabled() && HCR_EL2.TGE == '1' then
            AArch64.SystemAccessTrap(EL2, 0x18);
        else
            AArch64.SystemAccessTrap(EL1, 0x18);
    elsif EL2Enabled() && HCR_EL2.<E2H,TGE> == '11'
&& SCTLR_EL2.TIDCP == '1' then
        AArch64.SystemAccessTrap(EL2, 0x18);
    else
        AArch64.ImpDefSysRegWrite(op0, op1, CRn,
CRm, op2, t);
elsif PSTATE.EL == EL1 then
    if EL2Enabled() && HCR_EL2.TIDCP == '1' then
        AArch64.SystemAccessTrap(EL2, 0x18);
    else
        AArch64.ImpDefSysRegWrite(op0, op1, CRn,
CRm, op2, t);
elsif PSTATE.EL == EL2 then
    AArch64.ImpDefSysRegWrite(op0, op1, CRn, CRm,
op2, t);
elsif PSTATE.EL == EL3 then
    AArch64.ImpDefSysRegWrite(op0, op1, CRn, CRm,
op2, t);
```

When FEAT_SYSREG128 is implemented

MRRS <Xt+1>, <Xt>, S3_<op1>_C<Cn>_C<Cm>_<op2>

op0	op1	CRn	CRm	op2
0b11	op1[2:0]	0b1x11	Cm[3:0]	op2[2:0]

```
if PSTATE.EL == EL0 then
    if Halted() && HaveEL(EL3) && EDSCR.SDD == '1'
&& boolean IMPLEMENTATION_DEFINED "EL3 trap priority
when SDD == '1'" && SCR_EL3.EnIDCP128 == '0' then
        UNDEFINED;
    elsif !(EL2Enabled() && HCR_EL2.<E2H,TGE> ==
'11') && SCTLR_EL1.TIDCP == '1' then
        if EL2Enabled() && HCR_EL2.TGE == '1' then
            AArch64.SystemAccessTrap(EL2, 0x14);
        else
            AArch64.SystemAccessTrap(EL1, 0x14);
    elsif EL2Enabled() && HCR_EL2.<E2H,TGE> == '11'
&& SCTLR_EL2.TIDCP == '1' then
        AArch64.SystemAccessTrap(EL2, 0x14);
    elsif !(EL2Enabled() && HCR_EL2.<E2H,TGE> ==
'11') && (!IsSCTLR2EL1Enabled() ||
SCTLR2_EL1.EnIDCP128 == '0') then
        if EL2Enabled() && HCR_EL2.TGE == '1' then
            AArch64.SystemAccessTrap(EL2, 0x14);
        else
```

```
            AArch64.SystemAccessTrap(EL1, 0x14);
    elsif EL2Enabled() && HCR_EL2.<E2H,TGE> == '11'
&& (!IsSCTLR2EL2Enabled() || SCTLR2_EL2.EnIDCP128 ==
'0') then
        AArch64.SystemAccessTrap(EL2, 0x14);
    elsif EL2Enabled() && HCR_EL2.<E2H,TGE> != '11'
&& (!IsHCRXEL2Enabled() || HCRX_EL2.EnIDCP128 ==
'0') then
        AArch64.SystemAccessTrap(EL2, 0x14);
    elsif HaveEL(EL3) && SCR_EL3.EnIDCP128 == '0'
then
        if Halted() && EDSCR.SDD == '1' then
            UNDEFINED;
        else
            AArch64.SystemAccessTrap(EL3, 0x14);
    else
        AArch64.ImpDefSysRegRead128(op0, op1, CRn,
CRm, op2, t, t + 1);
elsif PSTATE.EL == EL1 then
    if Halted() && HaveEL(EL3) && EDSCR.SDD == '1'
&& boolean IMPLEMENTATION_DEFINED "EL3 trap priority
when SDD == '1'" && SCR_EL3.EnIDCP128 == '0' then
        UNDEFINED;
    elsif EL2Enabled() && HCR_EL2.TIDCP == '1' then
        AArch64.SystemAccessTrap(EL2, 0x14);
    elsif EL2Enabled() && (!IsHCRXEL2Enabled() ||
HCRX_EL2.EnIDCP128 == '0') then
        AArch64.SystemAccessTrap(EL2, 0x14);
    elsif HaveEL(EL3) && SCR_EL3.EnIDCP128 == '0'
then
        if Halted() && EDSCR.SDD == '1' then
            UNDEFINED;
        else
            AArch64.SystemAccessTrap(EL3, 0x14);
    else
        AArch64.ImpDefSysRegRead128(op0, op1, CRn,
CRm, op2, t, t + 1);
elsif PSTATE.EL == EL2 then
    if Halted() && HaveEL(EL3) && EDSCR.SDD == '1'
&& boolean IMPLEMENTATION_DEFINED "EL3 trap priority
when SDD == '1'" && SCR_EL3.EnIDCP128 == '0' then
        UNDEFINED;
    elsif HaveEL(EL3) && SCR_EL3.EnIDCP128 == '0'
then
        if Halted() && EDSCR.SDD == '1' then
            UNDEFINED;
        else
            AArch64.SystemAccessTrap(EL3, 0x14);
    else
        AArch64.ImpDefSysRegRead128(op0, op1, CRn,
CRm, op2, t, t + 1);
elsif PSTATE.EL == EL3 then
    AArch64.ImpDefSysRegRead128(op0, op1, CRn, CRm,
op2, t, t + 1);
```

When FEAT_SYSREG128 is implemented

MSRR S3_<op1>_C<Cn>_C<Cm>_<op2>, <Xt+1>, <Xt>

op0	op1	CRn	CRm	op2
0b11	op1[2:0]	0b1x11	Cm[3:0]	op2[2:0]

```
if PSTATE.EL == EL0 then
    if Halted() && HaveEL(EL3) && EDSCR.SDD == '1'
&& boolean IMPLEMENTATION_DEFINED "EL3 trap priority
when SDD == '1'" && SCR_EL3.EnIDCP128 == '0' then
        UNDEFINED;
    elsif !(EL2Enabled() && HCR_EL2.<E2H,TGE> ==
'11') && SCTLR_EL1.TIDCP == '1' then
        if EL2Enabled() && HCR_EL2.TGE == '1' then
            AArch64.SystemAccessTrap(EL2, 0x14);
        else
            AArch64.SystemAccessTrap(EL1, 0x14);
    elsif EL2Enabled() && HCR_EL2.<E2H,TGE> == '11'
&& SCTLR_EL2.TIDCP == '1' then
        AArch64.SystemAccessTrap(EL2, 0x14);
    elsif !(EL2Enabled() && HCR_EL2.<E2H,TGE> ==
'11') && (!IsSCTLR2EL1Enabled() ||
SCTLR2_EL1.EnIDCP128 == '0') then
        if EL2Enabled() && HCR_EL2.TGE == '1' then
            AArch64.SystemAccessTrap(EL2, 0x14);
        else
            AArch64.SystemAccessTrap(EL1, 0x14);
    elsif EL2Enabled() && HCR_EL2.<E2H,TGE> == '11'
&& (!IsSCTLR2EL2Enabled() || SCTLR2_EL2.EnIDCP128 ==
'0') then
        AArch64.SystemAccessTrap(EL2, 0x14);
    elsif EL2Enabled() && HCR_EL2.<E2H,TGE> != '11'
&& (!IsHCRXEL2Enabled() || HCRX_EL2.EnIDCP128 ==
'0') then
        AArch64.SystemAccessTrap(EL2, 0x14);
    elsif HaveEL(EL3) && SCR_EL3.EnIDCP128 == '0'
then
        if Halted() && EDSCR.SDD == '1' then
            UNDEFINED;
        else
            AArch64.SystemAccessTrap(EL3, 0x14);
    else
        AArch64.ImpDefSysRegWrite128(op0, op1, CRn,
CRm, op2, t, t + 1);
elsif PSTATE.EL == EL1 then
    if Halted() && HaveEL(EL3) && EDSCR.SDD == '1'
&& boolean IMPLEMENTATION_DEFINED "EL3 trap priority
when SDD == '1'" && SCR_EL3.EnIDCP128 == '0' then
        UNDEFINED;
    elsif EL2Enabled() && HCR_EL2.TIDCP == '1' then
        AArch64.SystemAccessTrap(EL2, 0x14);
    elsif EL2Enabled() && (!IsHCRXEL2Enabled() ||
HCRX_EL2.EnIDCP128 == '0') then
        AArch64.SystemAccessTrap(EL2, 0x14);
    elsif HaveEL(EL3) && SCR_EL3.EnIDCP128 == '0'
then
        if Halted() && EDSCR.SDD == '1' then
            UNDEFINED;
        else
            AArch64.SystemAccessTrap(EL3, 0x14);
    else
        AArch64.ImpDefSysRegWrite128(op0, op1, CRn,
CRm, op2, t, t + 1);
elsif PSTATE.EL == EL2 then
    if Halted() && HaveEL(EL3) && EDSCR.SDD == '1'
&& boolean IMPLEMENTATION_DEFINED "EL3 trap priority
when SDD == '1'" && SCR_EL3.EnIDCP128 == '0' then
        UNDEFINED;
    elsif HaveEL(EL3) && SCR_EL3.EnIDCP128 == '0'
then
        if Halted() && EDSCR.SDD == '1' then
            UNDEFINED;
        else
            AArch64.SystemAccessTrap(EL3, 0x14);
    else
        AArch64.ImpDefSysRegWrite128(op0, op1, CRn,
CRm, op2, t, t + 1);
elsif PSTATE.EL == EL3 then
    AArch64.ImpDefSysRegWrite128(op0, op1, CRn, CRm,
op2, t, t + 1);
```

28/03/2023 16:02; 72747e43966d6b97dcbd230a1b3f0421d1ea3d94