

## PMCCNTR\_EL0, Performance Monitors Cycle Count Register

The PMCCNTR\_EL0 characteristics are:

### Purpose

Holds the value of the processor Cycle Counter, CCNT, that counts processor clock cycles. See 'Time as measured by the Performance Monitors cycle counter' for more information.

[PMCCFILTR\\_EL0](#) determines the modes and states in which the PMCCNTR\_EL0 can increment.

### Configuration

AArch64 System register PMCCNTR\_EL0 bits [63:0] are architecturally mapped to AArch32 System register [PMCCNTR\[63:0\]](#).

AArch64 System register PMCCNTR\_EL0 bits [63:0] are architecturally mapped to External register [PMU.PMCCNTR\\_EL0\[63:0\]](#).

This register is present only when FEAT\_PMUv3 is implemented. Otherwise, direct accesses to PMCCNTR\_EL0 are undefined.

All counters are subject to any changes in clock frequency, including clock stopping caused by the WFI and WFE instructions. This means that it is constrained unpredictable whether or not PMCCNTR\_EL0 continues to increment when clocks are stopped by WFI and WFE instructions.

### Attributes

PMCCNTR\_EL0 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
CCNT																															
CCNT																															
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

## CCNT, bits [63:0]

Cycle count. Depending on the values of [PMCR\\_EL0](#).{LC,D}, this field increments in one of the following ways:

- Every processor clock cycle.
- Every 64th processor clock cycle.

Writing 1 to [PMCR\\_EL0](#).C sets this field to 0.

The reset behavior of this field is:

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

## Accessing PMCCNTR\_EL0

PMCCNTR\_EL0 reads-as-zero and ignores writes if all of the following are true:

- FEAT\_PMUv3p9 is implemented.
- PSTATE.EL == EL0.
- [PMUSERENR\\_EL0](#).UEN == 1.
- [PMUACR\\_EL1](#).C == 0.

PMCCNTR\_EL0 ignores writes if all of the following are true:

- FEAT\_PMUv3p9 is implemented.
- PSTATE.EL == EL0.
- [PMUSERENR\\_EL0](#).{UEN,CR} == {1,1}.

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

## MRS <Xt>, PMCCNTR\_EL0

op0	op1	CRn	CRm	op2
0b11	0b011	0b1001	0b1101	0b000

```
if PSTATE.EL == EL0 then
    if Halted() && HaveEL(EL3) && EDSCR.SDD == '1'
    && boolean IMPLEMENTATION_DEFINED "EL3 trap priority
when SDD == '1'" && MDCR_EL3.TPM == '1' then
        UNDEFINED;
    elsif PMUSERENR_EL0.<CR,EN> == '00' then
        if EL2Enabled() && HCR_EL2.TGE == '1' then
            AArch64.SystemAccessTrap(EL2, 0x18);
        else
            AArch64.SystemAccessTrap(EL1, 0x18);
    elsif EL2Enabled() && HCR_EL2.<E2H,TGE> != '11'
```

```

&& IsFeatureImplemented(FEAT_FGT) && (!HaveEL(EL3)
|| SCR_EL3.FGTEn == '1') && HDFGRTR_EL2.PMCCNTR_EL0
== '1' then
    AArch64.SystemAccessTrap(EL2, 0x18);
elseif EL2Enabled() && MDCR_EL2.TPM == '1' then
    AArch64.SystemAccessTrap(EL2, 0x18);
elseif HaveEL(EL3) && MDCR_EL3.TPM == '1' then
    if Halted() && EDSCR.SDD == '1' then
        UNDEFINED;
    else
        AArch64.SystemAccessTrap(EL3, 0x18);
    else
        X[t, 64] = PMCCNTR_EL0;
elseif PSTATE.EL == EL1 then
    if Halted() && HaveEL(EL3) && EDSCR.SDD == '1'
&& boolean IMPLEMENTATION_DEFINED "EL3 trap priority
when SDD == '1'" && MDCR_EL3.TPM == '1' then
        UNDEFINED;
    elseif EL2Enabled() &&
IsFeatureImplemented(FEAT_FGT) && (!HaveEL(EL3) ||
SCR_EL3.FGTEn == '1') && HDFGRTR_EL2.PMCCNTR_EL0 ==
'1' then
        AArch64.SystemAccessTrap(EL2, 0x18);
    elseif EL2Enabled() && MDCR_EL2.TPM == '1' then
        AArch64.SystemAccessTrap(EL2, 0x18);
    elseif HaveEL(EL3) && MDCR_EL3.TPM == '1' then
        if Halted() && EDSCR.SDD == '1' then
            UNDEFINED;
        else
            AArch64.SystemAccessTrap(EL3, 0x18);
        else
            X[t, 64] = PMCCNTR_EL0;
elseif PSTATE.EL == EL2 then
    if Halted() && HaveEL(EL3) && EDSCR.SDD == '1'
&& boolean IMPLEMENTATION_DEFINED "EL3 trap priority
when SDD == '1'" && MDCR_EL3.TPM == '1' then
        UNDEFINED;
    elseif HaveEL(EL3) && MDCR_EL3.TPM == '1' then
        if Halted() && EDSCR.SDD == '1' then
            UNDEFINED;
        else
            AArch64.SystemAccessTrap(EL3, 0x18);
        else
            X[t, 64] = PMCCNTR_EL0;
elseif PSTATE.EL == EL3 then
    X[t, 64] = PMCCNTR_EL0;

```

## MSR PMCCNTR\_EL0, <Xt>

op0	op1	CRn	CRm	op2
0b11	0b011	0b1001	0b1101	0b000

```

if PSTATE.EL == EL0 then

```

```

        if Halted() && HaveEL(EL3) && EDSCR.SDD == '1'
        && boolean IMPLEMENTATION_DEFINED "EL3 trap priority
when SDD == '1'" && MDCR_EL3.TPM == '1' then
            UNDEFINED;
        elsif PMUSERENR_EL0.EN == '0' then
            if EL2Enabled() && HCR_EL2.TGE == '1' then
                AArch64.SystemAccessTrap(EL2, 0x18);
            else
                AArch64.SystemAccessTrap(EL1, 0x18);
            elsif EL2Enabled() && HCR_EL2.<E2H,TGE> != '11'
&& IsFeatureImplemented(FEAT_FGT) && (!HaveEL(EL3)
|| SCR_EL3.FGTEn == '1') && HDFGWTR_EL2.PMCCNTR_EL0
== '1' then
                AArch64.SystemAccessTrap(EL2, 0x18);
            elsif EL2Enabled() && MDCR_EL3.TPM == '1' then
                AArch64.SystemAccessTrap(EL2, 0x18);
            elsif HaveEL(EL3) && MDCR_EL3.TPM == '1' then
                if Halted() && EDSCR.SDD == '1' then
                    UNDEFINED;
                else
                    AArch64.SystemAccessTrap(EL3, 0x18);
                else
                    PMCCNTR_EL0 = X[t, 64];
            elsif PSTATE.EL == EL1 then
                if Halted() && HaveEL(EL3) && EDSCR.SDD == '1'
&& boolean IMPLEMENTATION_DEFINED "EL3 trap priority
when SDD == '1'" && MDCR_EL3.TPM == '1' then
                    UNDEFINED;
                elsif EL2Enabled() &&
IsFeatureImplemented(FEAT_FGT) && (!HaveEL(EL3) ||
SCR_EL3.FGTEn == '1') && HDFGWTR_EL2.PMCCNTR_EL0 ==
'1' then
                    AArch64.SystemAccessTrap(EL2, 0x18);
                elsif EL2Enabled() && MDCR_EL3.TPM == '1' then
                    AArch64.SystemAccessTrap(EL2, 0x18);
                elsif HaveEL(EL3) && MDCR_EL3.TPM == '1' then
                    if Halted() && EDSCR.SDD == '1' then
                        UNDEFINED;
                    else
                        AArch64.SystemAccessTrap(EL3, 0x18);
                    else
                        PMCCNTR_EL0 = X[t, 64];
            elsif PSTATE.EL == EL2 then
                if Halted() && HaveEL(EL3) && EDSCR.SDD == '1'
&& boolean IMPLEMENTATION_DEFINED "EL3 trap priority
when SDD == '1'" && MDCR_EL3.TPM == '1' then
                    UNDEFINED;
                elsif HaveEL(EL3) && MDCR_EL3.TPM == '1' then
                    if Halted() && EDSCR.SDD == '1' then
                        UNDEFINED;
                    else
                        AArch64.SystemAccessTrap(EL3, 0x18);
                    else
                        PMCCNTR_EL0 = X[t, 64];
            elsif PSTATE.EL == EL3 then
                PMCCNTR_EL0 = X[t, 64];

```

[AArch32  
Registers](#)

[AArch64  
Registers](#)

[AArch32  
Instructions](#)

[AArch64  
Instructions](#)

[Index by  
Encoding](#)

[External  
Registers](#)

28/03/2023 16:02; 72747e43966d6b97dcbd230a1b3f0421d1ea3d94

Copyright © 2010-2023 Arm Limited or its affiliates. All rights reserved. This document is Non-Confidential.