

ARM64 ELF hwcaps

This document describes the usage and semantics of the arm64 ELF hwcaps.

1. Introduction

Some hardware or software features are only available on some CPU implementations, and/or with certain kernel configurations, but have no architected discovery mechanism available to userspace code at EL0. The kernel exposes the presence of these features to userspace through a set of flags called hwcaps, exposed in the auxilliary vector.

Userspace software can test for features by acquiring the AT_HWCAP or AT_HWCAP2 entry of the auxiliary vector, and testing whether the relevant flags are set, e.g.:

```
bool floating_point_is_present(void)
{
    unsigned long hwcaps = getauxval(AT_HWCAP);
    if (hwcaps & HWCAP_FP)
        return true;

    return false;
}
```

Where software relies on a feature described by a hwcaps, it should check the relevant hwcaps flag to verify that the feature is present before attempting to make use of the feature.

Features cannot be probed reliably through other means. When a feature is not available, attempting to use it may result in unpredictable behaviour, and is not guaranteed to result in any reliable indication that the feature is unavailable, such as a SIGILL.

2. Interpretation of hwcaps

The majority of hwcaps are intended to indicate the presence of features which are described by architected ID registers inaccessible to userspace code at EL0. These hwcaps are defined in terms of ID register fields, and should be interpreted with reference to the definition of these fields in the ARM Architecture Reference Manual (ARM ARM).

Such hwcaps are described below in the form:

Functionality implied by `idreg.field == val`.

Such hwcaps indicate the availability of functionality that the ARM ARM defines as being present when `idreg.field` has value `val`, but do not indicate that `idreg.field` is precisely equal to `val`, nor do they indicate the absence of functionality implied by other values of `idreg.field`.

Other hwcaps may indicate the presence of features which cannot be described by ID registers alone. These may be described without reference to ID registers, and may refer to other documentation.

3. The hwcaps exposed in AT_HWCAP

HWCAP_FP

Functionality implied by `ID_AA64PFR0_EL1.FP == 0b0000`.

HWCAP_ASIMD

Functionality implied by `ID_AA64PFR0_EL1.AdvSIMD == 0b0000`.

HWCAP_EVTSTRM

The generic timer is configured to generate events at a frequency of approximately 10KHz.

HWCAP_AES

Functionality implied by `ID_AA64ISAR0_EL1.AES == 0b0001`.

HWCAP_PMULL

Functionality implied by `ID_AA64ISAR0_EL1.AES == 0b0010`.

HWCAP_SHA1

Functionality implied by `ID_AA64ISAR0_EL1.SHA1 == 0b0001`.

HWCAP_SHA2

Functionality implied by `ID_AA64ISAR0_EL1.SHA2 == 0b0001`.

HWCAP_CRC32

Functionality implied by ID_AA64ISAR0_EL1.CRC32 == 0b0001.

HWCAP_ATOMICS

Functionality implied by ID_AA64ISAR0_EL1.Atomic == 0b0010.

HWCAP_FPHP

Functionality implied by ID_AA64PFR0_EL1.FP == 0b0001.

HWCAP_ASIMDHP

Functionality implied by ID_AA64PFR0_EL1.AdvSIMD == 0b0001.

HWCAP_CPUID

EL0 access to certain ID registers is available, to the extent described by Documentation/arm64/cpu-feature-registers.rst.

These ID registers may imply the availability of features.

HWCAP_ASIMDRDM

Functionality implied by ID_AA64ISAR0_EL1.RDM == 0b0001.

HWCAP_JSCVT

Functionality implied by ID_AA64ISAR1_EL1.JSCVT == 0b0001.

HWCAP_FCMA

Functionality implied by ID_AA64ISAR1_EL1.FCMA == 0b0001.

HWCAP_LRCPC

Functionality implied by ID_AA64ISAR1_EL1.LRCPC == 0b0001.

HWCAP_DCPOP

Functionality implied by ID_AA64ISAR1_EL1.DPB == 0b0001.

HWCAP_SHA3

Functionality implied by ID_AA64ISAR0_EL1.SHA3 == 0b0001.

HWCAP_SM3

Functionality implied by ID_AA64ISAR0_EL1.SM3 == 0b0001.

HWCAP_SM4

Functionality implied by ID_AA64ISAR0_EL1.SM4 == 0b0001.

HWCAP_ASIMDDP

Functionality implied by ID_AA64ISAR0_EL1.DP == 0b0001.

HWCAP_SHA512

Functionality implied by ID_AA64ISAR0_EL1.SHA2 == 0b0010.

HWCAP_SVE

Functionality implied by ID_AA64PFR0_EL1.SVE == 0b0001.

HWCAP_ASIMDFHM

Functionality implied by ID_AA64ISAR0_EL1.FHM == 0b0001.

HWCAP_DIT

Functionality implied by ID_AA64PFR0_EL1.DIT == 0b0001.

HWCAP_USCAT

Functionality implied by ID_AA64MMFR2_EL1.AT == 0b0001.

HWCAP_ILRCPC

Functionality implied by ID_AA64ISAR1_EL1.LRCPC == 0b0010.

HWCAP_FLAGM

Functionality implied by ID_AA64ISAR0_EL1.TS == 0b0001.

HWCAP_SSBS

Functionality implied by ID_AA64PFR1_EL1.SSBS == 0b0010.

HWCAP_SB

Functionality implied by ID_AA64ISAR1_EL1.SB == 0b0001.

HWCAP_PACA

Functionality implied by ID_AA64ISAR1_EL1.APA == 0b0001 or ID_AA64ISAR1_EL1.API == 0b0001, as described by Documentation/arm64/pointer-authentication.rst.

HWCAP2_PACG

Functionality implied by ID_AA64ISAR1_EL1.GPA == 0b0001 or ID_AA64ISAR1_EL1.GPI == 0b0001, as described by Documentation/arm64/pointer-authentication.rst.

HWCAP2_DCPODP

Functionality implied by ID_AA64ISAR1_EL1.DPB == 0b0010.

HWCAP2_SVE2

Functionality implied by ID_AA64ZFR0_EL1.SVEVer == 0b0001.

HWCAP2_SVEAES

Functionality implied by ID_AA64ZFR0_EL1.AES == 0b0001.

HWCAP2_SVEPMULL

Functionality implied by ID_AA64ZFR0_EL1.AES == 0b0010.

HWCAP2_SVEBITPERM

Functionality implied by ID_AA64ZFR0_EL1.BitPerm == 0b0001.

HWCAP2_SVSHA3

Functionality implied by ID_AA64ZFR0_EL1.SHA3 == 0b0001.

HWCAP2_SVESM4

Functionality implied by ID_AA64ZFR0_EL1.SM4 == 0b0001.

HWCAP2_FLAGM2

Functionality implied by ID_AA64ISAR0_EL1.TS == 0b0010.

HWCAP2_FRINT

Functionality implied by ID_AA64ISAR1_EL1.FRINTTS == 0b0001.

HWCAP2_SVEI8MM

Functionality implied by ID_AA64ZFR0_EL1.I8MM == 0b0001.

HWCAP2_SVEF32MM

Functionality implied by ID_AA64ZFR0_EL1.F32MM == 0b0001.

HWCAP2_SVEF64MM

Functionality implied by ID_AA64ZFR0_EL1.F64MM == 0b0001.

HWCAP2_SVEBF16

Functionality implied by ID_AA64ZFR0_EL1.BF16 == 0b0001.

HWCAP2_I8MM

Functionality implied by ID_AA64ISAR1_EL1.I8MM == 0b0001.

HWCAP2_BF16

Functionality implied by ID_AA64ISAR1_EL1.BF16 == 0b0001.

HWCAP2_DGH

Functionality implied by ID_AA64ISAR1_EL1.DGH == 0b0001.

HWCAP2_RNG

Functionality implied by ID_AA64ISAR0_EL1.RNDR == 0b0001.

HWCAP2_BTI

Functionality implied by ID_AA64PFR0_EL1.BT == 0b0001.

HWCAP2_MTE

Functionality implied by ID_AA64PFR1_EL1.MTE == 0b0010, as described by Documentation/arm64/memory-tagging-extension.rst.

HWCAP2_ECV

Functionality implied by ID_AA64MMFR0_EL1.ECV == 0b0001.

HWCAP2_AFP

Functionality implied by ID_AA64MFR1_EL1.AFP == 0b0001.

HWCAP2_RPRES

Functionality implied by ID_AA64ISAR2_EL1.RPRES == 0b0001.

HWCAP2_MTE3

Functionality implied by ID_AA64PFR1_EL1.MTE == 0b0011, as described by Documentation/arm64/memory-tagging-extension.rst.

4. Unused AT_HWCAP bits

For interoperation with userspace, the kernel guarantees that bits 62 and 63 of AT_HWCAP will always be returned as 0.