ARC ABI addendum

# Build attributes

Build attributes record data that a linker uses to determine the compatibility, or incompatibility, of a set of relocatable files. Other tools that consume relocatable files may also benefit from the data. Build attributes are designed to reflect ARC CPU configuration options defined in ARC Programmer Reference Manual.

The main uses of the attributes are to

1. ensure compatibility and/or check compatibility between two distinct tool chains, and
2. within a toolchain, to allow the linker, assembler or disassembler to diagnose incompatibility, or enforce compatibility.

The attributes values are based on the user intentions at compile time, which are also important at link time.

Build attributes are intended to check two types of compatibility:

1. The compatibility of binary code with a target hardware configuration;
2. The procedure-call compatibility between toolchains ABI variations.

## Use cases

Build attributes can be used to:

1. Link two objects, one produced by GNU and the other produced by MWDT;
2. Pass to the disassembler exactly the machine configuration we want to disassemble for;
3. Pass to the assembler the exact machine the compiler compiled for;
4. Pass to the simulator (ARC nSIM) the exact machine configuration;
5. check the object attributes against the running machine when loading a dynamic module,.

# Representing build attributes in ELF files

## Encoding

Build attributes are encoded in a section of type *SHT\_ARC\_ATTRIBUTES* (0x70000001), with a name of *.ARC.attributes*.

The content of the section is a stream of bytes. An attribute is encoded in a *<tag, value*>pair. Both tags and numerical values are encoded using unsigned LEB128 encoding (ULEB128), DWARF-3 style which will allow values in the range 0-127. String values are encoded using NULL-terminated byte strings (NTBS).

An attribute section contains a sequence of subsections. Each one is either:

1. Defined by this ABI and public to all tools that process that file. This subsection is defined by the “ARC” pseudo-vector.
2. Private to a tool vendor’s tools. This information may be safely ignored if it is not understood.

Attributes can apply to:

* A whole translation unit;
* A section
* A function (symbol of type STT\_FUNC).

### Syntactic structure

The overall syntactic structure of an attribute section is:

<format-version: ‘A’>

[<uint32: subsection-length> <NTBS: vendor-name>

[<file-tag: 0x01> <uint32: byte-size> <attribute>\*

| <section-tag: 0x02> <uint32: byte-size> <section-number>\* 0 <attribute>\*

|<symbol-tag: 0x03> <uint32: byte-size> <symbol-number>\* 0 <attribute>\*

]+

]\*

A public subsection contains any number of sub-subsections. Each records attributes relating to:

* The whole relocatable file. These sub-subsections contain just a list of attributes. They are identified by a leading Tag\_File (=1) byte.
* A set of sections within the relocatable file. These sub-subsections contain a list of ULEB128 section numbers followed by a list of attributes. They are identified by a leading Tag\_Section (=2) byte.
* A set of (defined) symbols in the relocatable file. These sub-subsections contain a list of ULEB128 symbol numbers followed by a list of attributes. They are identified by a leading Tag\_Symbol (=3) byte.

In each case, byte-size is a 4-byte unsigned integer in the byte order of the ELF file. Byte-size includes the initial tag byte, the size field itself, and the sub-subsection content. That is, it is the byte offset from the start of this subsubsection to the start of the next sub-subsection. Both section indexes and defined symbol indexes are non-zero, so a NULL byte ends a string and a list of indexes without ambiguity

## Overview of public ARC attributes

|  |  |  |  |
| --- | --- | --- | --- |
| Value | Tag | Visibility | Parameter type |
| 4 | Tag\_ARC\_PCS\_config | Public | uleb128 |
| 5 | Tag\_ARC\_CPU\_base | Public | uleb128 |
| 6 | Tag\_ARC\_CPU\_variation | Public | uleb128 |
| 7 | Tag\_ARC\_CPU\_name | Public | NTBS |
| 8 | Tag\_ARC\_ABI\_rf16 | Public | uleb128 |
| 9 | Tag\_ARC\_ABI\_osver | Public | uleb128 |
| 10 | Tag\_ARC\_ABI\_sda | Public | uleb128 |
| 11 | Tag\_ARC\_ABI\_pic | Public | uleb128 |
| 12 | Tag\_ARC\_ABI\_tls | Public | uleb128 |
| 13 | Tag\_ARC\_ABI\_enumsize | Public | uleb128 |
| 14 | Tag\_ARC\_ABI\_exceptions | Public | uleb128 |
| 15 | Tag\_ARC\_ABI\_double\_size | Public | uleb128 |
| 16 | Tag\_ARC\_ISA\_config | Public | NTBS |
| 17 | Tag\_ARC\_ISA\_apex | Public | NTBS |
| 18 | Tag\_ARC\_ISA\_mpy\_option | Public | uleb128 |
| 19 | Tag\_ARC\_ISA\_lpc\_size | Public | uleb128 |
| 20 | Tag\_ARC\_ATR\_version | Public | uleb128 |
| 21 | Tag\_ARC\_ABI\_pack\_struct | Public | uleb128 |

## ARC Platform configuration.

### Tag\_ARC\_PCS\_config

Defines thehe intended use of the produced object. This attribute is required. An absent value will cause errors when linking with anything else than absent/non standard.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Value | Attribute name | Default | Allowed Values | Meaning |
| 4 | Tag\_ARC\_PCS\_config | Default | 0 | Absent/Non standard |
|  |  |  | 1 | Bare-metal/mwdt |
|  |  |  | 2 | Bare-metal/newlib |
|  |  |  | 3 | Linux/uclibc |
|  |  |  | 4 | Linux/glibc |

## ARC target related attributes







































































### Tag\_ARC\_CPU\_base

Defines the intended target CPU. This attribute is mandatory. It can be derived from .cpu pseudo op.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Value | Attribute name | Default | Allowed Values | Meaning |
| 5 | Tag\_ARC\_CPU\_base | Default | 0 | Absent/legacy |
|  |  |  | 1 | ARC6xx |
|  |  |  | 2 | ARC7xx |
|  |  |  | 3 | ARCEM |
|  |  |  | 4 | ARCHS |

### Tag\_ARC\_CPU\_variation

Defines additional information to CPU\_base. This attribute is optional and can be omitted.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Value | Attribute name | Default | Allowed Values | Meaning |
| 6 | Tag\_ARC\_CPU\_variation | Default | 0 | Absent/Default/Core0 |
|  |  |  | 1-15 | Core1-Core14 |

### Tag\_ARC\_CPU\_name

Defines name of the CPU, which can be the name of a specific manufacturer, a generic name, or any other name. This attribute can be omitted.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Value | Attribute name | Default | Allowed Values | Meaning |
| 7 | Tag\_ARC\_CPU\_name | “” | name | CPU name |

### Tag\_ARC\_ISA\_config

Comma separated list of ISA extensions. This attribute is optional and can be omitted. The names accepted are defined.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Value | Attribute name | Default | Allowed Values | Meaning |
| 16 | Tag\_ARC\_ISA\_config | “” | <NTBS> | Comma separated list of ISA extensions. |

#### Recognized ISA name extensions

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Name | Extension | A6xx | A7xx | HS | EM |
| BITSCAN | Bit scan | Optional | Optional | Optional | Optional |
| bs | Barrel shifter | Optional | Optional | Default | Optional |
| swap | Swap ops | Optional | Optional | Default | Optional |
| DIV\_REM | Division/remainder | N.A. | N.A. | Default | Optional |
| NPS400 | NPS400 | N.A. | Optional | N.A. | N.A. |
| cd | Code density | N.A. | N.A. | Default | Optional |
| QUARKSE | QuarkSE-EM | N.A. | N.A. | N.A. | Optional |
| spfp | FPX single precision | Optional | Optional | N.A. | Optional |
| dpfp | FPX double precision | Optional | Optional | N.A. | Optional |
| FPUDA | FP double assist | N.A. | N.A. | N.A. | Optional |
| FPUS | FPU single precision | N.A. | N.A. | Optional | Optional |
| FPUD | FPU double precision | N.A. | N.A. | Optional | N.A. |
| SA | Shift assist | N.A. | N.A. | Default | Optional |

### Tag\_ARC\_ISA\_apex

Defines list of APEX extensions present. This attribute is optional and can be omitted.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Value | Attribute name | Default | Allowed Values | Meaning |
| 17 | Tag\_ARC\_ISA\_apex | “” | <NTBS> | Comma separated list of APEX extensions. |

### Tag\_ARC\_ISA\_mpy\_option

Defines MPY configuration option. This attribute is optional and can be omitted.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Value | Attribute name | Default | Allowed Values | Meaning |
| 18 | Tag\_ARC\_ISA\_mpy\_option | “” | <NTBS> | Comma separated list of APEX extensions. |

### Tag\_ARC\_ISA\_lpc\_size

Defines the number of bits in the LP\_COUNT register. This attribute is optional and can be omitted.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Value | Attribute name | Default | Allowed Values | Meaning |
| 19 | Tag\_ARC\_ISA\_lpc\_size | 32 | 8, 16, 24, 32 | Number of bits. |

## ARC ABI related attributes

### Tag\_ARC\_ABI\_rf16

Indicates whether CPU has a reduced register set. This attribute is mandatory. If not specified, Default value is assumed and selected.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Value | Attribute name | Default | Allowed Values | Meaning |
| 8 | Tag\_ARC\_ABI\_rf16 | Default | 0 | Absent/Full register file |
|  |  |  | 1 | Reduced register file |



























### Tag\_ARC\_ABI\_osver

Defines ABI version. This attribute also controls the corresponding eflag value. This attribute is optional and can be omitted.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Value | Attribute name | Default | Allowed Values | Meaning |
| 9 | Tag\_ARC\_ABI\_osver |  | 0 | Unset/Not available |
|  |  |  | 1 | Reserved |
|  |  |  | 2 | OSABI v2 |
|  |  |  | 3 | OSABI v3 |
|  |  | Default | 4 | OSABI v4 |

### Tag\_ARC\_ABI\_sda

Indicates whether small data implementation is present. This attribute is mandatory. If not specified, Default value is assumed and selected.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Value | Attribute name | Default | Allowed Values | Meaning |
| 10 | Tag\_ARC\_ABI\_sda | Default | 0 | Absent |
|  |  |  | 1 | MWDT specific |
|  |  |  | 2 | GNU specific |

### Tag\_ARC\_ABI\_pic

Indicates whether pic implementation is present. This attribute is mandatory.. If not specified, Default value is assumed and selected.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Value | Attribute name | Default | Allowed Values | Meaning |
| 11 | Tag\_ARC\_ABI\_pic | Default | 0 | Absent |
|  |  |  | 1 | MWDT specific |
|  |  |  | 2 | GNU specific |

### Tag\_ARC\_ABI\_tls

Indicates whether R25 is used as Thread pointer or not. This attribute is mandatory. If not specified Default value is assumed and selected.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Value | Attribute name | Default | Allowed Values | Meaning |
| 12 | Tag\_ARC\_ABI\_tls | Default | 0 | Absent/not used |
|  |  |  | 1 | Use R25 as thread pointer |

### Tag\_ARC\_ABI\_enumsize

Defines the enum size. This attribute is mandatory. If not specified, Default value is assumed and selected.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Value | Attribute name | Default | Allowed Values | Meaning |
| 13 | Tag\_ARC\_ABI\_enumsize | Default | 0 | Default/32-bit container |
|  |  |  | 1 | Smallest container |

### Tag\_ARC\_ABI\_exceptions

Indicates whether libgcc OPTFP library is used or any other ABI exception. This library uses a non-standard ABI calling convention hence extra care is needed when linking. This attribute is mandatory. If not specified Default value is assumed and selected.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Value | Attribute name | Default | Allowed Values | Meaning |
| 14 | Tag\_ARC\_ABI\_exceptions | Default | 0 | Absent |
|  |  |  | 1 | Libgcc OPTFP library |

### Tag\_ARC\_ATR\_version

Shows the attribute version. If set to 1 indicates the attribute section is encoded using old MWDT encoding.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Value | Attribute name | Default | Allowed Values | Meaning |
| 20 | Tag\_ARC\_ATR\_version | Default | 0 | Absent/GNU |
|  |  |  | 1 | MWDT compatible |

### Tag\_ARC\_ABI\_pack\_struct

Indicates the value used by GCC compatible option “-fpack-struct=n” that defines the maximum alignment of struct members. In addition, we relax things so that if the user specifies 8, then 8-byte integers and double are 8-byte aligned.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Value | Attribute name | Default | Allowed Values | Meaning |
| 21 | Tag\_ARC\_ABI\_pack\_struct | Default | 0 | Absent |
|  |  |  | n | Maximum alignment of struct members |

## References

https://sourceware.org/binutils/docs/as/Object-Attributes.html#Object-Attributes