

NAME

iconv_open - allocate descriptor for character set conversion

SYNOPSIS

```
#include <iconv.h>
```

```
iconv_t iconv_open (const char* tocode, const char* fromcode);
```

DESCRIPTION

The **iconv_open** function allocates a conversion descriptor suitable for converting byte sequences from character encoding *fromcode* to character encoding *tocode*.

The values permitted for *fromcode* and *tocode* and the supported combinations are system dependent. For the libiconv library, the following encodings are supported, in all combinations.

European languages

ASCII, ISO-8859-{1,2,3,4,5,7,9,10,13,14,15,16}, KOI8-R, KOI8-U, KOI8-RU, CP{1250,1251,1252,1253,1254,1257}, CP{850,866,1131}, Mac{Roman,CentralEurope,Iceland,Croatian,Romania}, Mac{Cyrillic,Ukraine,Greek,Turkish}, Macintosh

Semitic languages

ISO-8859-{6,8}, CP{1255,1256}, CP862, Mac{Hebrew,Arabic}

Japanese

EUC-JP, SHIFT_JIS, CP932, ISO-2022-JP, ISO-2022-JP-2, ISO-2022-JP-1, ISO-2022-JP-MS

Chinese

EUC-CN, HZ, GBK, CP936, GB18030, EUC-TW, BIG5, CP950, BIG5-HKSCS, BIG5-HKSCS:2004, BIG5-HKSCS:2001, BIG5-HKSCS:1999, ISO-2022-CN, ISO-2022-CN-EXT

Korean

EUC-KR, CP949, ISO-2022-KR, JOHAB

Armenian

ARMSCII-8

Georgian

Georgian-Academy, Georgian-PS

Tajik

KOI8-T

Kazakh

PT154, RK1048

Thai

TIS-620, CP874, MacThai

Laotian

MuleLao-1, CP1133

Vietnamese

VISCII, TCVN, CP1258

Platform specifics

HP-ROMAN8, NEXTSTEP

Full Unicode

UTF-8
 UCS-2, UCS-2BE, UCS-2LE
 UCS-4, UCS-4BE, UCS-4LE
 UTF-16, UTF-16BE, UTF-16LE
 UTF-32, UTF-32BE, UTF-32LE

UTF-7
C99, JAVA

Full Unicode, in terms of **uint16_t** or **uint32_t**
(with machine dependent endianness and alignment)
UCS-2-INTERNAL, UCS-4-INTERNAL

Locale dependent, in terms of **char** or **wchar_t**
(with machine dependent endianness and alignment, and with semantics depending on the OS and the current LC_CTYPE locale facet)
char, **wchar_t**

When configured with the option **--enable-extra-encodings**, it also provides support for a few extra encodings:

European languages

CP{437,737,775,852,853,855,857,858,860,861,863,865,869,1125}

Semitic languages

CP864

Japanese

EUC-JISX0213, Shift_JISX0213, ISO-2022-JP-3

Chinese

BIG5-2003 (experimental)

Turkmen

TDS565

Platform specifics

ATARIST, RISCOS-LATIN1

The empty encoding name is equivalent to **char**: it denotes the locale dependent character encoding.

When the string **//TRANSLIT** is appended to *to_code*, transliteration is activated. This means that when a character cannot be represented in the target character set, it can be approximated through one or several characters that look similar to the original character.

When the string **//IGNORE** is appended to *to_code*, characters that cannot be represented in the target character set will be silently discarded.

The resulting conversion descriptor can be used with **iconv** any number of times. It remains valid until deallocated using **iconv_close**.

A conversion descriptor contains a conversion state. After creation using **iconv_open**, the state is in the initial state. Using **iconv** modifies the descriptor's conversion state. (This implies that a conversion descriptor can not be used in multiple threads simultaneously.) To bring the state back to the initial state, use **iconv** with **NULL** as *inbuf* argument.

RETURN VALUE

The **iconv_open** function returns a freshly allocated conversion descriptor. In case of error, it sets **errno** and returns **(iconv_t)(-1)**.

ERRORS

The following error can occur, among others:

EINVAL

The conversion from *from_code* to *to_code* is not supported by the implementation.

CONFORMING TO

POSIX:2001

SEE ALSO

[iconv\(3\)](#) [iconvctl\(3\)](#) [iconv_close\(3\)](#)