

**NAME**

**a64l**, **l64a**, **l64a\_r** - convert between a long integer and a base-64 ASCII string

**LIBRARY**

Standard C Library (libc, -lc)

**SYNOPSIS**

```
#include <stdlib.h>
```

*long*

```
a64l(const char *s);
```

*char \**

```
l64a(long int l);
```

*int*

```
l64a_r(long int l, char *buffer, int buflen);
```

**DESCRIPTION**

These functions are used to maintain numbers stored in radix-64 ASCII characters. This is a notation by which 32-bit integers can be represented by up to six characters; each character represents a digit in radix-64 notation. If the type long contains more than 32 bits, only the low-order 32 bits are used for these operations.

The characters used to represent "digits" are '.' for 0, '/' for 1, '0' - '9' for 2 - 11, 'A' - 'Z' for 12 - 37, and 'a' - 'z' for 38 - 63.

The **a64l**() function takes a pointer to a radix-64 representation, in which the first digit is the least significant, and returns a corresponding *long* value. If the string pointed to by *s* contains more than six characters, **a64l**() uses the first six. If the first six characters of the string contain a null terminator, **a64l**() uses only characters preceding the null terminator. The **a64l**() function scans the character string from left to right with the least significant digit on the left, decoding each character as a 6-bit radix-64 number. If the type long contains more than 32 bits, the resulting value is sign-extended. The behavior of **a64l**() is unspecified if *s* is a null pointer or the string pointed to by *s* was not generated by a previous call to **l64a**().

The **l64a**() function takes a *long* argument and returns a pointer to the corresponding radix-64 representation. The behavior of **l64a**() is unspecified if value is negative.

The value returned by **l64a**() is a pointer into a static buffer. Subsequent calls to **l64a**() may overwrite

the buffer.

The **l64a\_r()** function performs a conversion identical to that of **l64a()** and stores the resulting representation in the memory area pointed to by *buffer*, consuming at most *buflen* characters including the terminating NUL character.

## RETURN VALUES

On successful completion, **a64l()** returns the *long* value resulting from conversion of the input string. If a string pointed to by *s* is an empty string, **a64l()** returns 0.

The **l64a()** function returns a pointer to the radix-64 representation. If value is 0, **l64a()** returns a pointer to an empty string.

## SEE ALSO

[strtoul\(3\)](#)

## HISTORY

The **a64l()**, **l64a()**, and **l64a\_r()** functions are derived from NetBSD with modifications. They appeared in FreeBSD 6.1.

## AUTHORS

The **a64l()**, **l64a()**, and **l64a\_r()** functions were added to FreeBSD by Tom Rhodes <[trhodes@FreeBSD.org](mailto:trhodes@FreeBSD.org)>. Almost all of this manual page came from the POSIX standard.