NAME

unw_is_signal_frame -- check if current frame is a signal frame

SYNOPSIS

```
#include <libunwind.h>

int unw_is_signal_frame(unw_cursor_t *cp);
```

DESCRIPTION

The unw_is_signal_frame() routine returns a positive value if the current frame identified by cp is a signal frame, and a value of 0 otherwise. For the purpose of this discussion, a signal frame is a frame that was created in response to a potentially asynchronous interruption. For UNIX and UNIX-like platforms, such frames are normally created by the kernel when delivering a signal. In a kernel-environment, a signal frame might, for example, correspond to a frame created in response to a device interrupt.

Signal frames are somewhat unusual because the asynchronous nature of the events that create them require storing the contents of registers that are normally treated as scratch (‘‘caller-saved’’) registers.

RETURN VALUE

On successful completion, unw_is_signal_frame() returns a positive value if the current frame is a signal frame, or 0 if it is not. Otherwise, a negative value of one of the error-codes below is returned.

THREAD AND SIGNAL SAFETY

unw_is_signal_frame() is thread-safe as well as safe to use from a signal handler.

ERRORS

UNW_ENOINFO

Libunwind is unable to determine whether or not the current frame is a signal frame.

SEE ALSO

libunwind(3), unw_get_reg(3), unw_set_reg(3), unw_get_fpreg(3), unw_set_fpreg(3)

AUTHOR

David Mosberger-Tang
Email: **dmosberger@gmail.com**
WWW: **http://www.nongnu.org/libunwind/**.