Operation of 8-byte stack alignment

On an exception entry when CCR.STKALIGN is set to 1, the exception entry sequence ensures that the stack pointer in use before the exception entry has 8-byte alignment, by adjusting its alignment if necessary. When the processor pushes the PSR value to the stack it uses bit[9] of the stacked PSR value to indicate whether it realigned the stack.

Note

In normal operation, PSR[9] is reserved.

Figure B1-3 shows the frame of information pushed onto the stack on exception entry, and how the processor reserves an additional word on the stack, if necessary, to obtain 8-byte stack alignment.

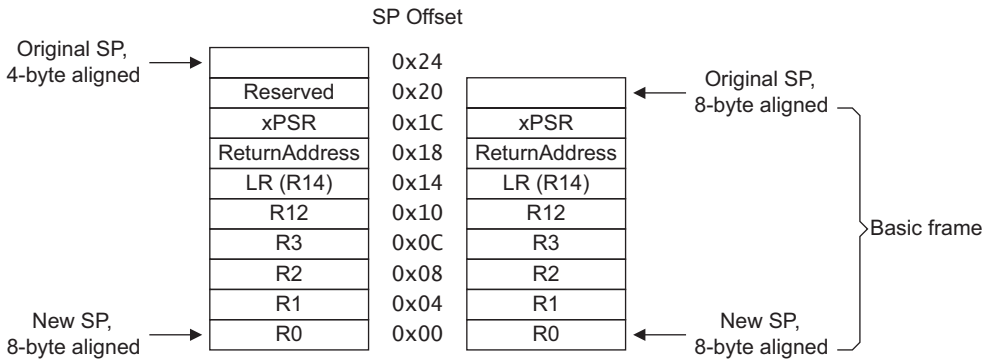

Figure B1-3 Alignment options when stacking the basic frame

When a processor implements the FP extension, on an exception entry it can push a 26-word frame onto the stack, to include FP state information, see *Context state stacking on exception entry with the FP extension* on page B1-650. Therefore:

- an 8-word context state frame pushed onto the stack on exception entry on a processor that does not include the FP extension is called a Basic frame
- a 26-word context state frame that can be pushed onto the stack on exception entry on a processor that includes the FP extension is called an Extended frame.

On an exception return when CCR.STKALIGN is set to 1, the processor uses the value of bit [9] of the PSR value popped from the stack to determine whether it must adjust the stack pointer alignment. This reverses any forced stack alignment performed on the exception entry.

The pseudocode in *Exception entry behavior* on page B1-643 and *Exception return behavior* on page B1-652 describes the effect of the CCR.STKALIGN bit value on exception entry and exception return.