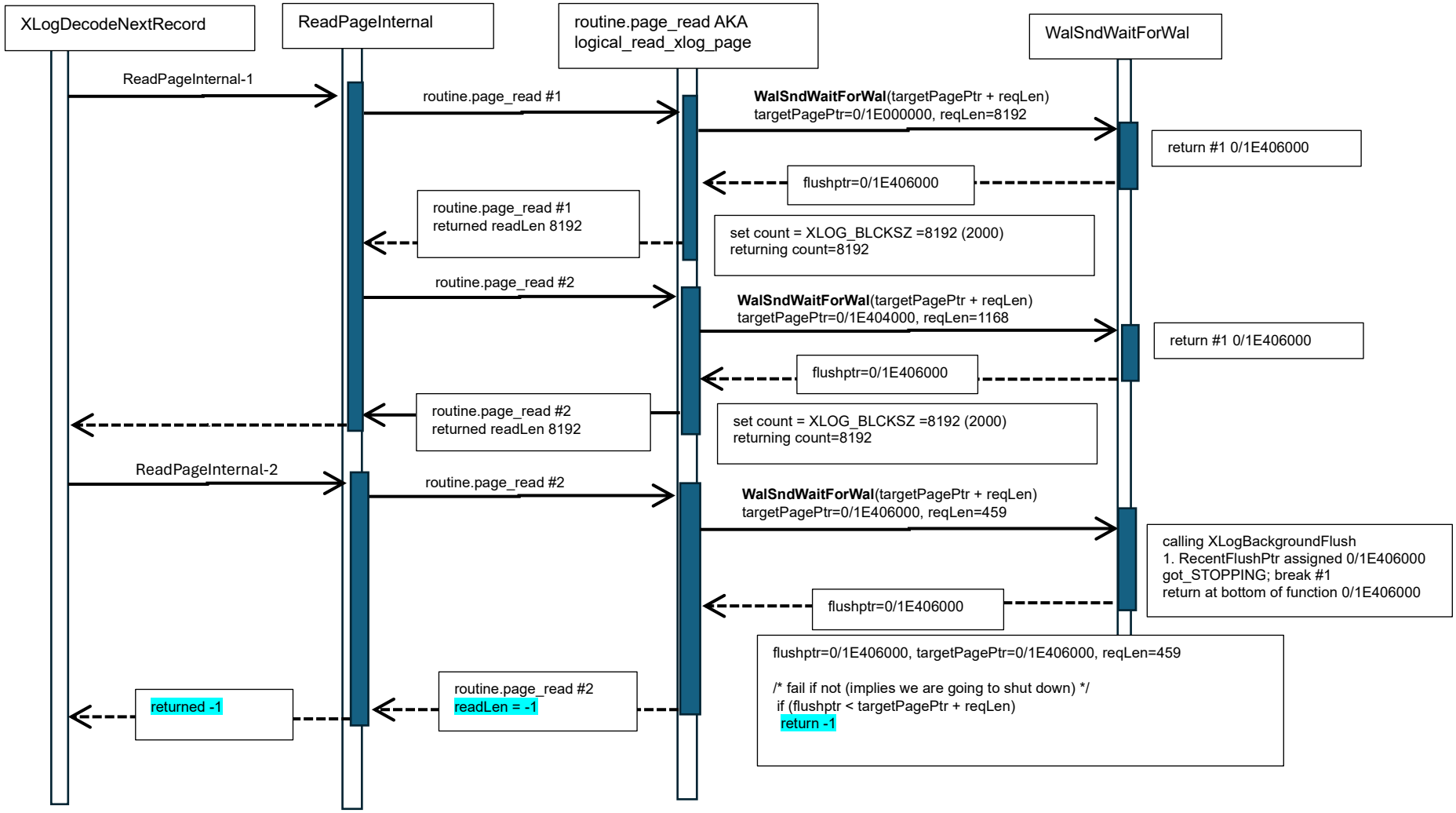
LOGGING EVIDENCE - bad8

Here is some logging my “bad8” logs…. When the test fails it gets into this repeating cycle of similar logs where the EndRecPtr and the flushptr cease to move. So, nothing happens and everything just hang until the TAP test times-out and dies.

```
CYCLE
2024-06-06 09:28:46.860 AEST [4988] tap_sub LOG:  >>> XLogDecodeNextRecord: before ReadPageInternal-1
2024-06-06 09:28:46.860 AEST [4988] tap_sub LOG:  >>> ReadPageInternal: before calling routine.page_read #1
2024-06-06 09:28:46.860 AEST [4988] tap_sub LOG:  >>> logical_read_xlog_page: calling WalSndWaitForWal
2024-06-06 09:28:46.860 AEST [4988] tap_sub LOG:  >>> WalSndWaitForWal: return #1 0/1E406000
2024-06-06 09:28:46.860 AEST [4988] tap_sub LOG:  >>> logical_read_xlog_page: just called WalSndWaitForWal. flushptr=0/1E406000, targetPagePtr=0/1E000000, reqLen=8192
2024-06-06 09:28:46.860 AEST [4988] tap_sub LOG:  >>> logical_read_xlog_page: set count = XLOG_BLCKSZ =8192 (2000)
2024-06-06 09:28:46.860 AEST [4988] tap_sub LOG:  >>> logical_read_xlog_page: returning count=8192
2024-06-06 09:28:46.860 AEST [4988] tap_sub LOG:  >>> ReadPageInternal: routine.page_read #1 returned readLen 8192
2024-06-06 09:28:46.860 AEST [4988] tap_sub LOG:  >>> ReadPageInternal: before calling routine.page_read #2
2024-06-06 09:28:46.860 AEST [4988] tap_sub LOG:  >>> logical_read_xlog_page: calling WalSndWaitForWal
2024-06-06 09:28:46.860 AEST [4988] tap_sub LOG:  >>> WalSndWaitForWal: return #1 0/1E406000
2024-06-06 09:28:46.860 AEST [4988] tap_sub LOG:  >>> logical_read_xlog_page: just called WalSndWaitForWal. flushptr=0/1E406000, targetPagePtr=0/1E404000, reqLen=1168
2024-06-06 09:28:46.860 AEST [4988] tap_sub LOG:  >>> logical_read_xlog_page: set count = XLOG_BLCKSZ =8192 (2000)
2024-06-06 09:28:46.860 AEST [4988] tap_sub LOG:  >>> logical_read_xlog_page: returning count=8192
2024-06-06 09:28:46.860 AEST [4988] tap_sub LOG:  >>> ReadPageInternal: routine.page_read #2 returned readLen 8192
2024-06-06 09:28:46.860 AEST [4988] tap_sub LOG:  >>> XLogDecodeNextRecord: before ReadPageInternal-2
2024-06-06 09:28:46.860 AEST [4988] tap_sub LOG:  >>> ReadPageInternal: before calling routine.page_read #2
2024-06-06 09:28:46.860 AEST [4988] tap_sub LOG:  >>> logical_read_xlog_page: calling WalSndWaitForWal
2024-06-06 09:28:46.860 AEST [4988] tap_sub LOG:  >>> WalSndWaitForWal: calling XLogBackgroundFlush
2024-06-06 09:28:46.860 AEST [4988] tap_sub LOG:  >>> WalSndWaitForWal: 1. RecentFlushPtr assigned 0/1E406000
2024-06-06 09:28:46.860 AEST [4988] tap_sub LOG:  >>> WalSndWaitForWal: got_STOPPING; break #1
2024-06-06 09:28:46.860 AEST [4988] tap_sub LOG:  >>> WalSndWaitForWal: return at bottom of function 0/1E406000
2024-06-06 09:28:46.860 AEST [4988] tap_sub LOG:  >>> logical_read_xlog_page: just called WalSndWaitForWal. flushptr=0/1E406000, targetPagePtr=0/1E406000, reqLen=459
2024-06-06 09:28:46.860 AEST [4988] tap_sub LOG:  >>> logical_read_xlog_page: returning -1. flushptr=0/1E406000, targetPagePtr=0/1E406000, reqLen=459
2024-06-06 09:28:46.860 AEST [4988] tap_sub LOG:  >>> ReadPageInternal: routine.page_read #2 returned readLen -1
2024-06-06 09:28:46.860 AEST [4988] tap_sub LOG:  >>> XLogDecodeNextRecord: called ReadPageInternal-2, which returned -1

NOTICE that EndRecPtr (0/1E404478) and flushPtr (0/1E406000) values below are unchanged at every cycle:
2024-06-06 09:28:46.860 AEST [4988] tap_sub LOG:  >>> XLogSendLogical: EndRecPtr is 0/1E404478, flushPtr is 0/1E406000, WalSndCaughtUp=0
2024-06-06 09:28:46.860 AEST [4988] tap_sub LOG:  >>> XLogSendLogical:
```

Diagram


XLogDecodeNextRecord
ReadPageInternal
routine.page_read AKA
logical_read_xlog_page
WalSndWaitForWal
ReadPageInternal-1
routine.page_read #1
WalSndWaitForWal(targetPagePtr + reqLen)
targetPagePtr=0/1E000000, reqLen=8192
return #1 0/1E406000
flushptr=0/1E406000
routine.page_read #1
returned readLen 8192
set count = XLOG_BLCKSZ =8192 (2000)
returning count=8192
routine.page_read #2
WalSndWaitForWal(targetPagePtr + reqLen)
targetPagePtr=0/1E404000, reqLen=1168
return #1 0/1E406000
flushptr=0/1E406000
routine.page_read #2
returned readLen 8192
set count = XLOG_BLCKSZ =8192 (2000)
returning count=8192
ReadPageInternal-2
routine.page_read #2
WalSndWaitForWal(targetPagePtr + reqLen)
targetPagePtr=0/1E406000, reqLen=459
calling XLogBackgroundFlush
1. RecentFlushPtr assigned 0/1E406000
got_STOPPING; break #1
return at bottom of function 0/1E406000
flushptr=0/1E406000
flushptr=0/1E406000, targetPagePtr=0/1E406000, reqLen=459
/* fail if not (implies we are going to shut down) */
if (flushptr < targetPagePtr + reqLen)
return -1
routine.page_read #2
readLen = -1
returned -1