

共有メモリ上の領域と数値

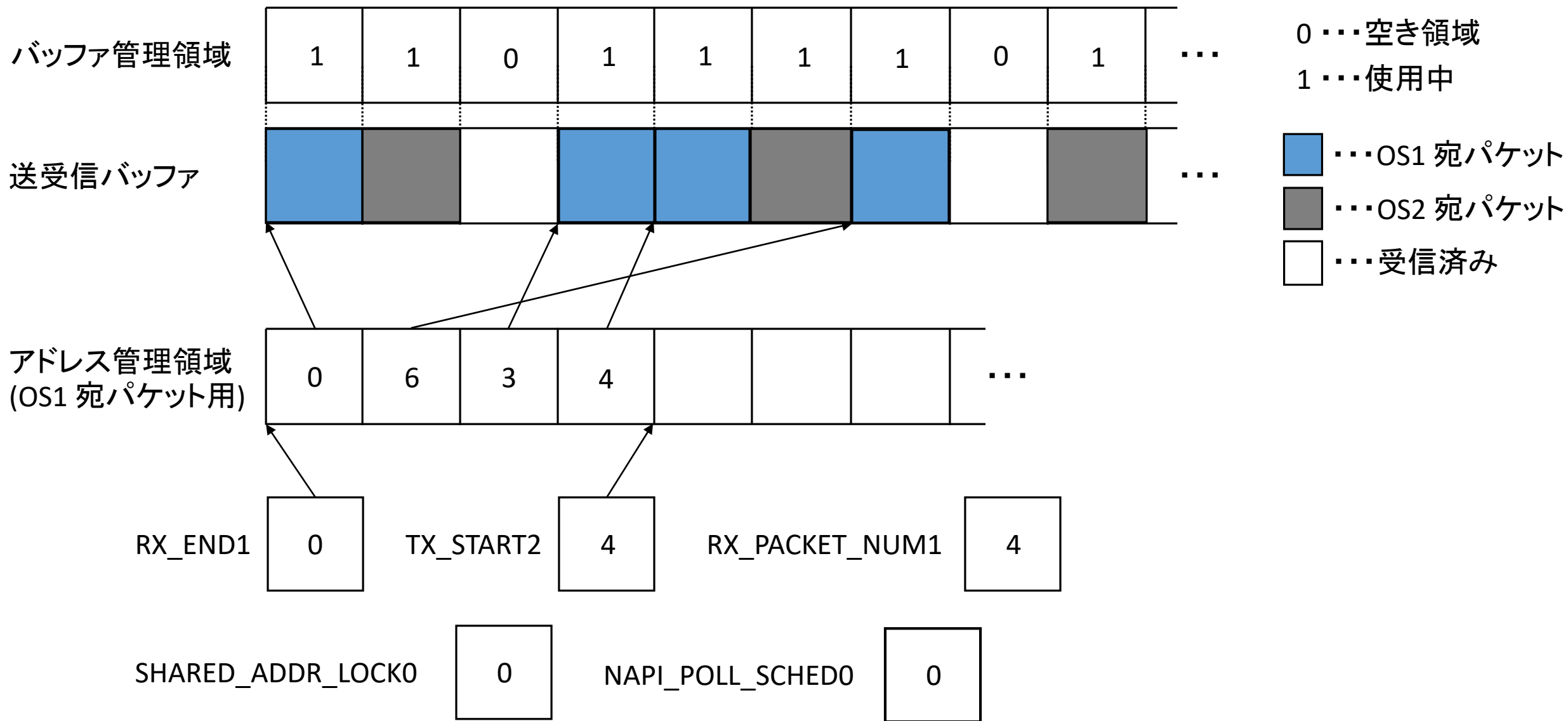
領域

- バッファ管理領域: 送受信バッファの使用状況を管理する領域
- 送受信バッファ: パケットを格納する領域
- アドレス管理領域: 対応する OS 宛のパケットを指示するポインタを保存する領域

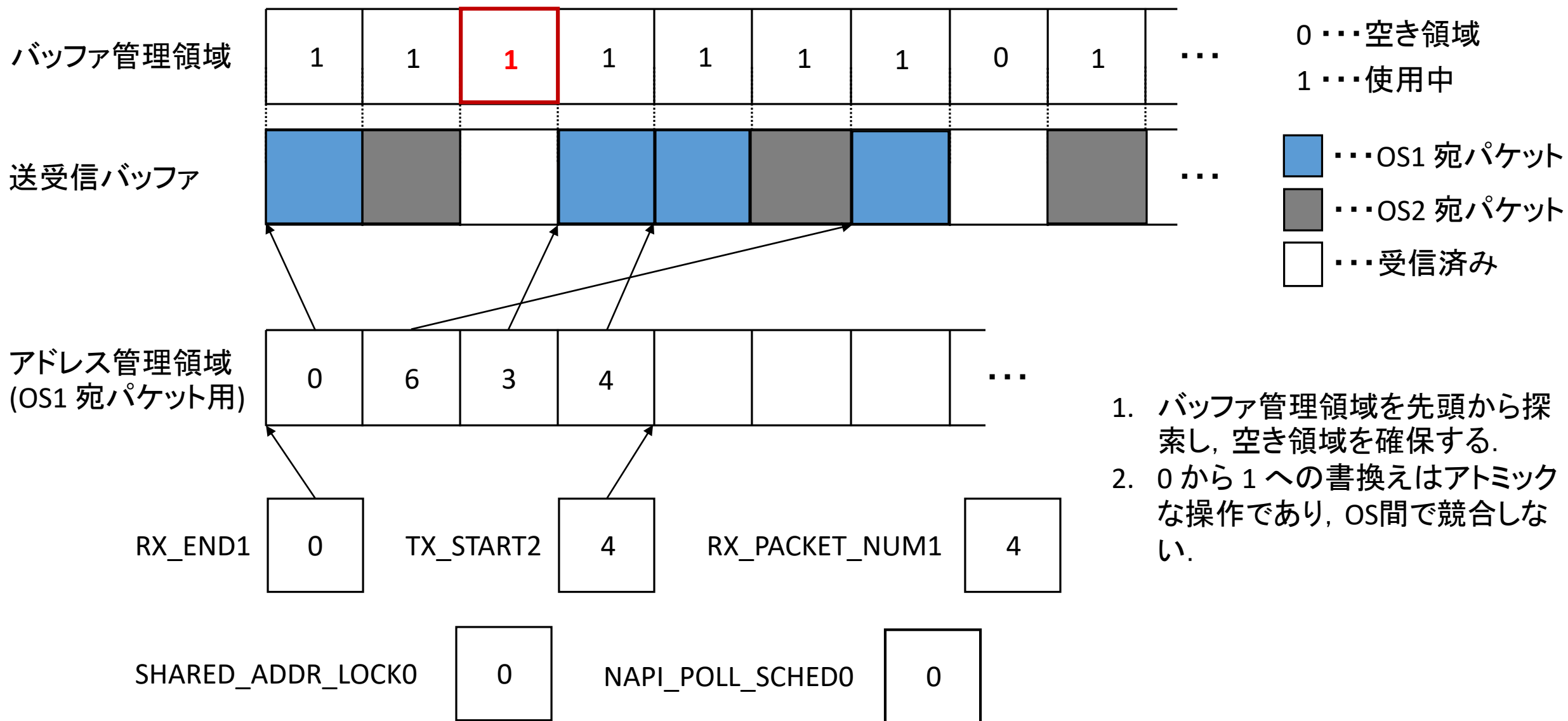
数値

- RX_END1: 次に OS1 の受信するパケットのアドレスへのポインタ
- TX_START2: 次に OS1 へ送信されるパケットのアドレスを書き込む場所へのポインタ
- RX_PACKET_NUM1: OS1 宛ての未受信パケット数を示す値
- SHARED_ADDR_LOCK0: ロック変数
- NAPI_POLL_SCHEDULE0: OS1 における受信ルーチンのスケジューリング状況を示す値

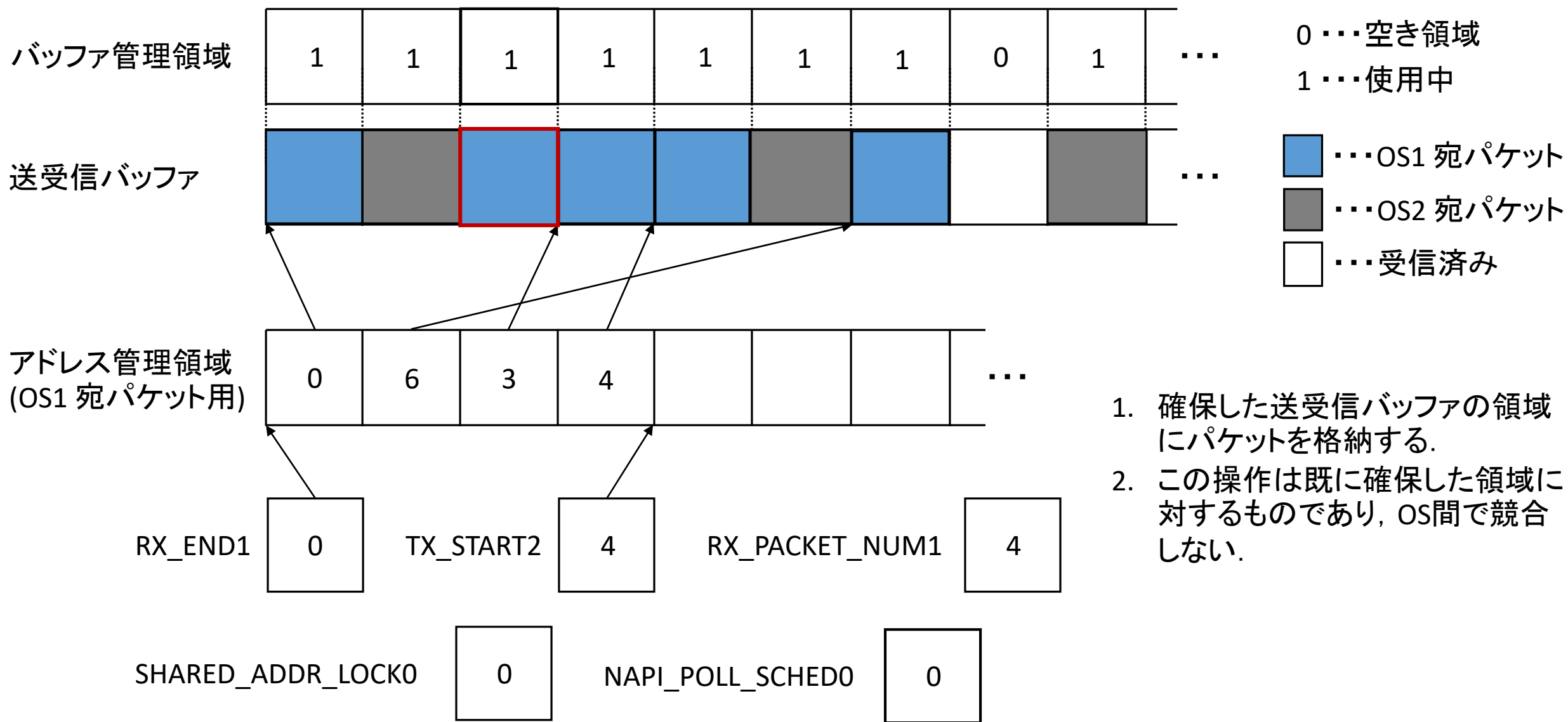
OS2からOS1宛のパケットを共有メモリに配置する様子(1/9)



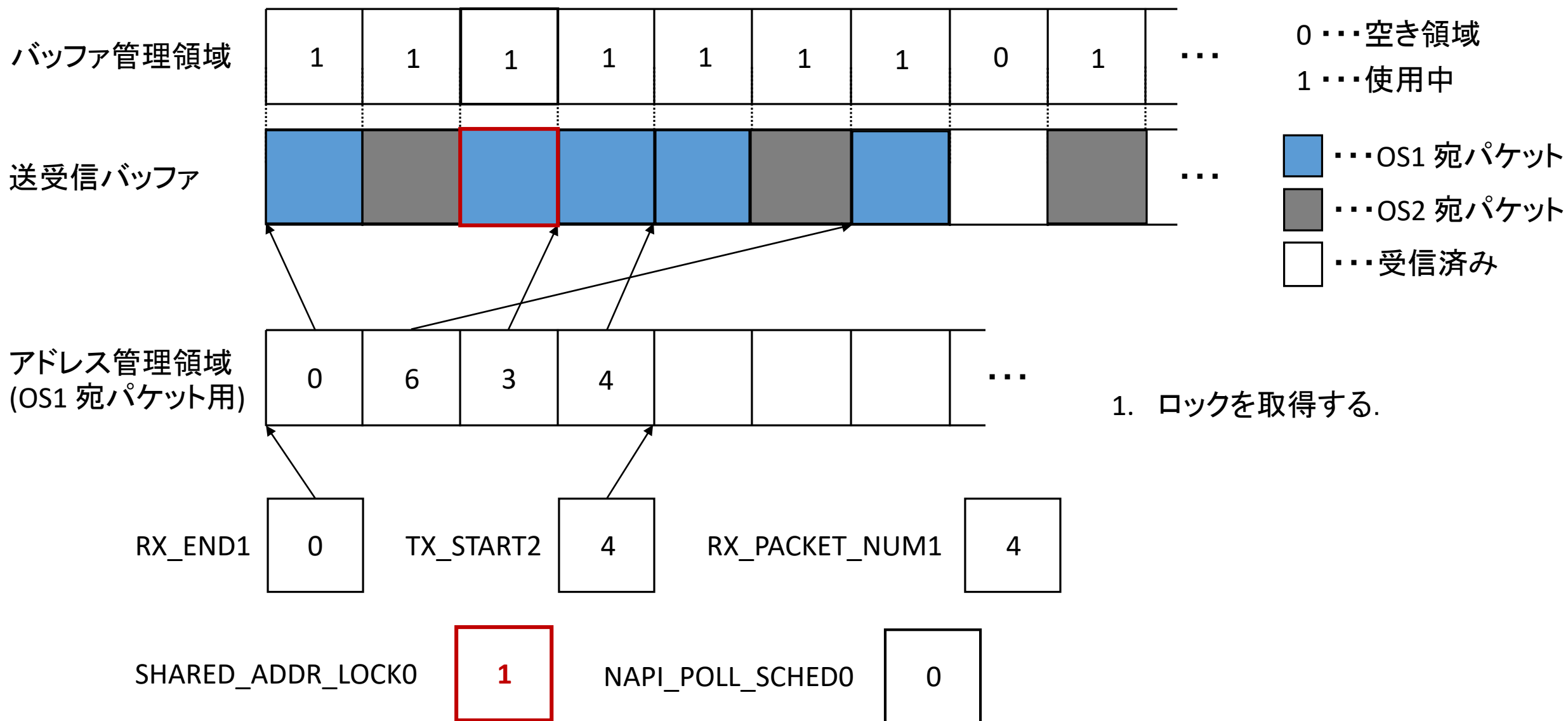
OS2からOS1宛のパケットを共有メモリに配置する様子(2/9)



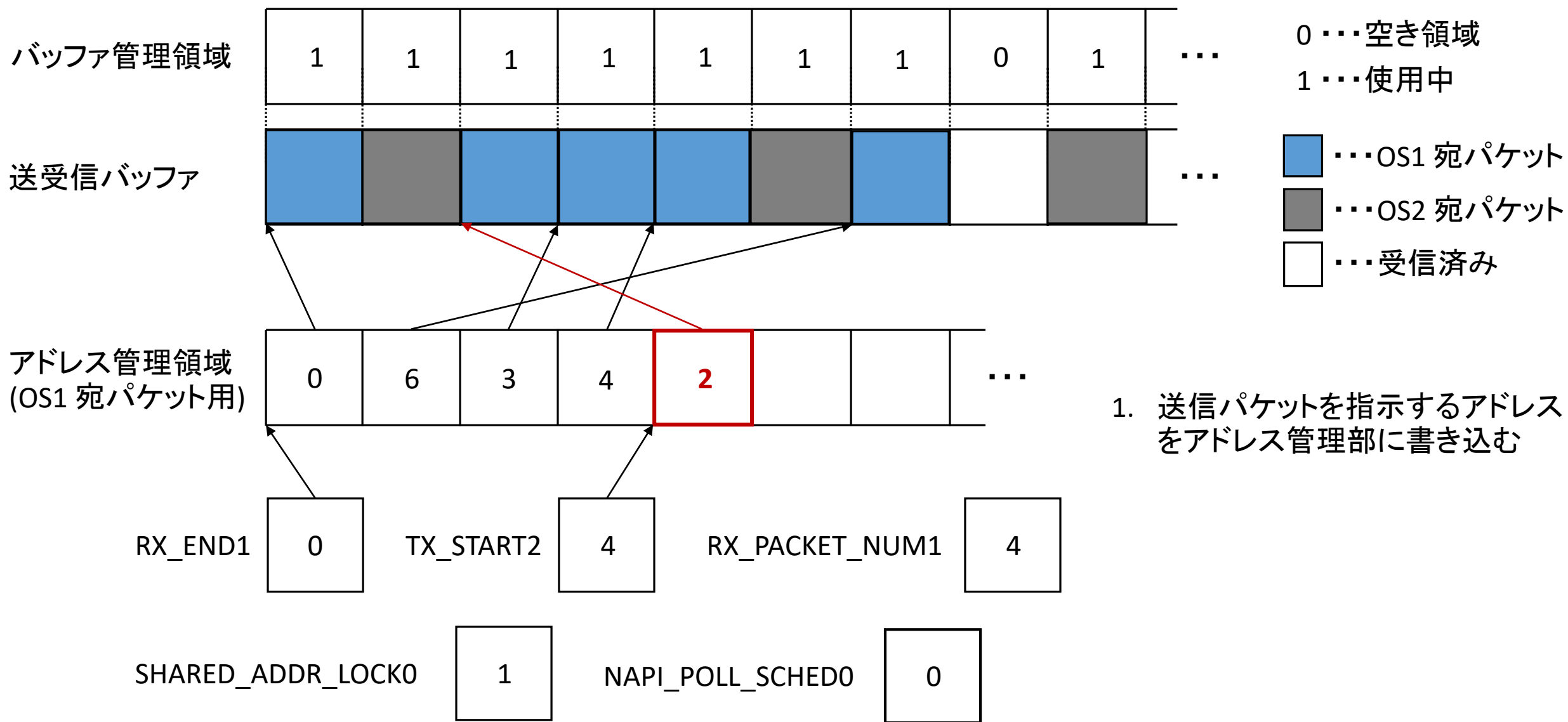
OS2からOS1宛のパケットを共有メモリに配置する様子(3/9)



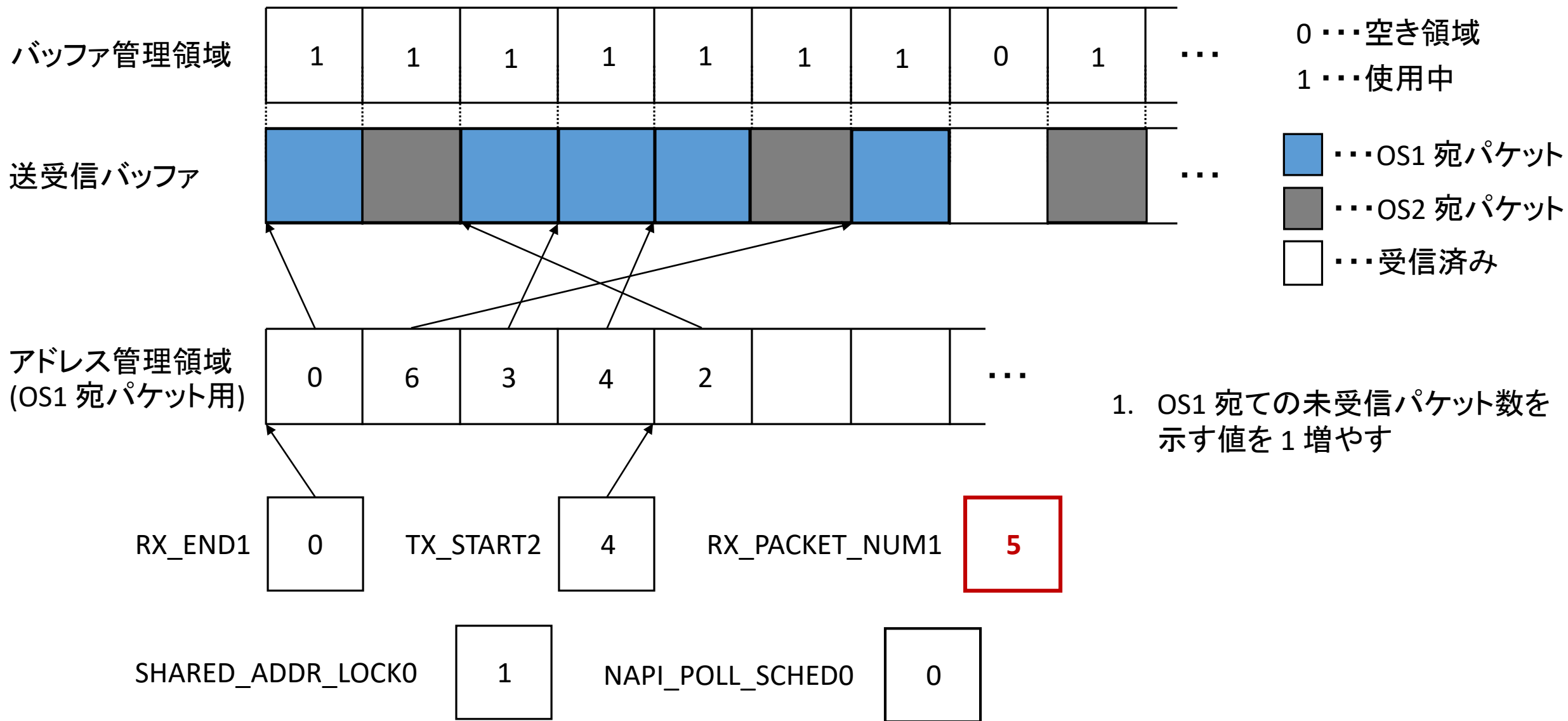
OS2からOS1宛のパケットを共有メモリに配置する様子(4/9)



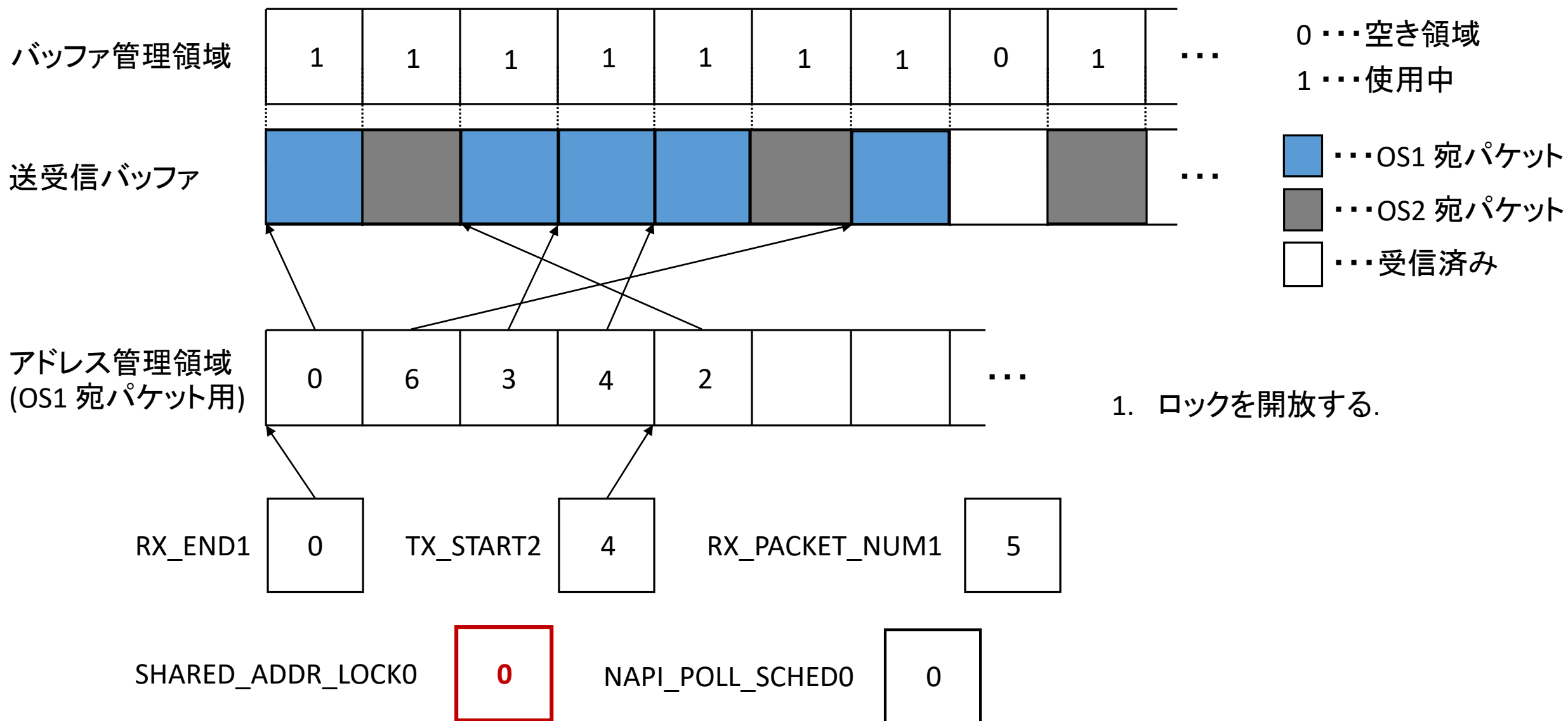
OS2からOS1宛のパケットを共有メモリに配置する様子(5/9)



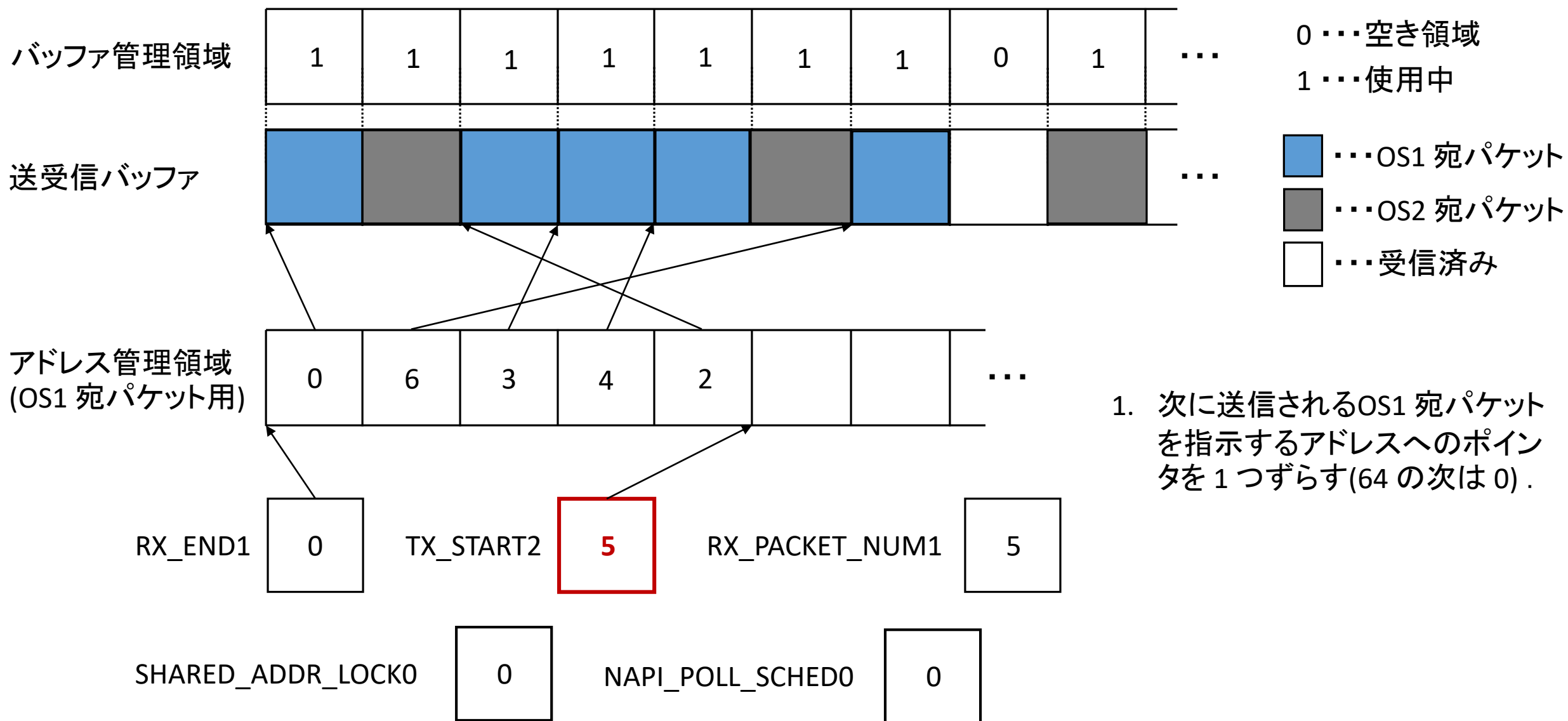
OS2からOS1宛のパケットを共有メモリに配置する様子(6/9)



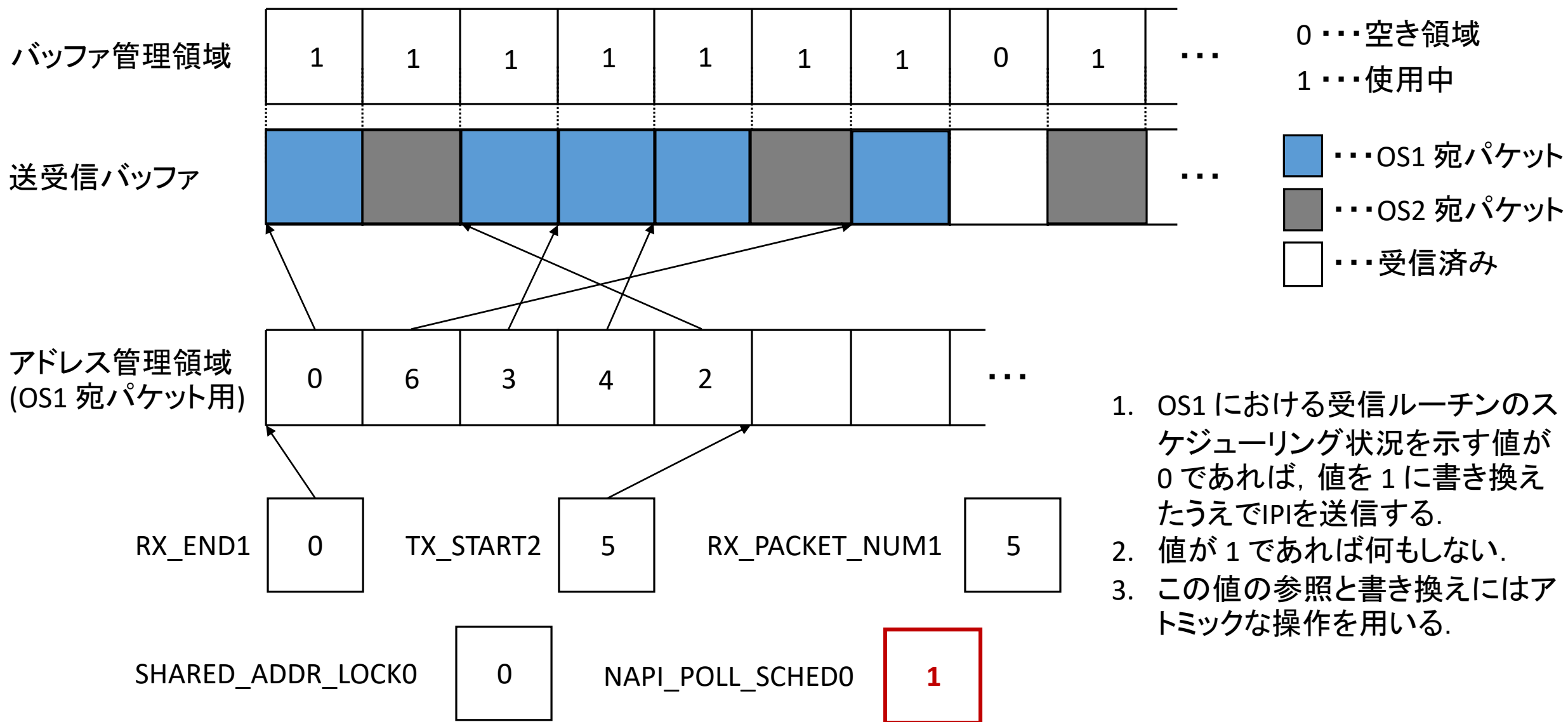
OS2からOS1宛のパケットを共有メモリに配置する様子(7/9)



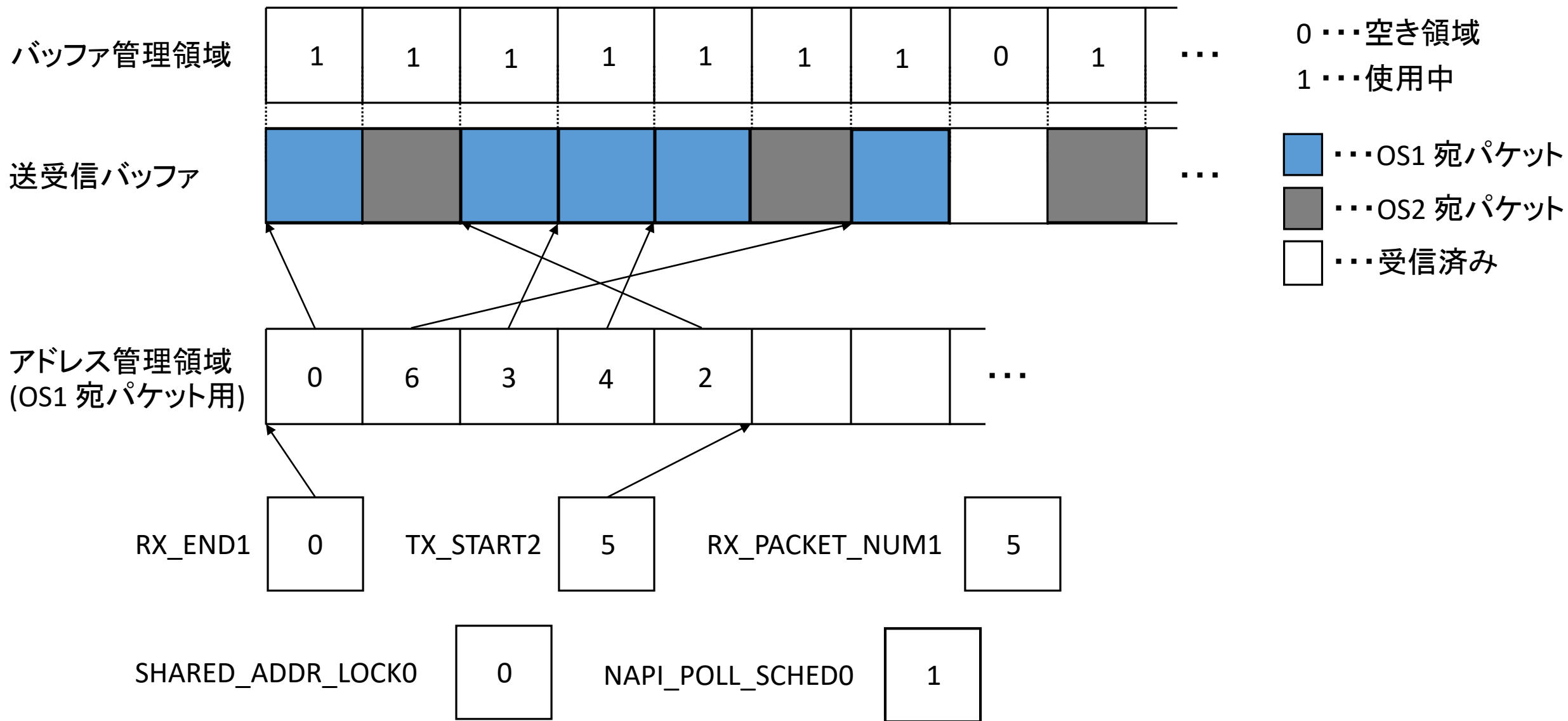
OS2からOS1宛のパケットを共有メモリに配置する様子(8/9)



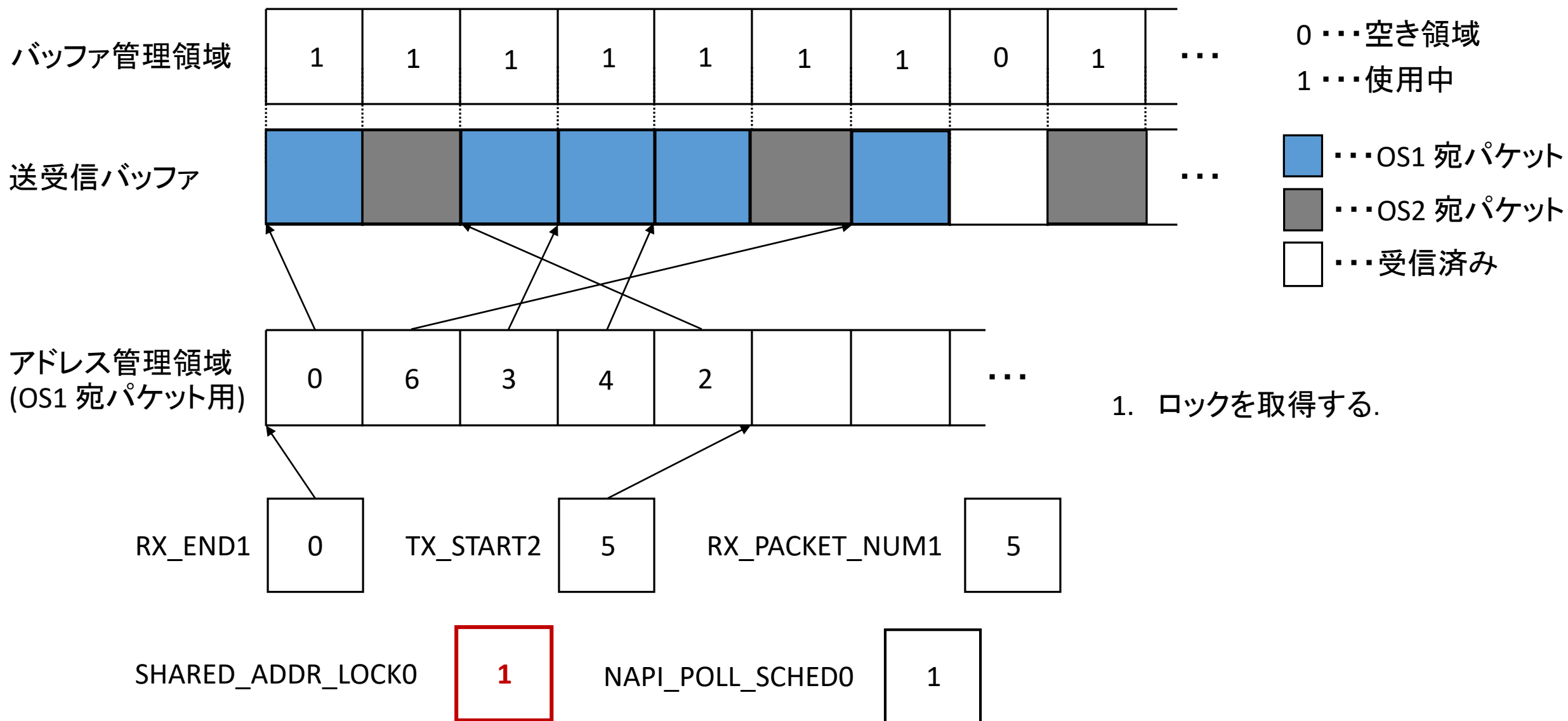
OS2からOS1宛のパケットを共有メモリに配置する様子(9/9)



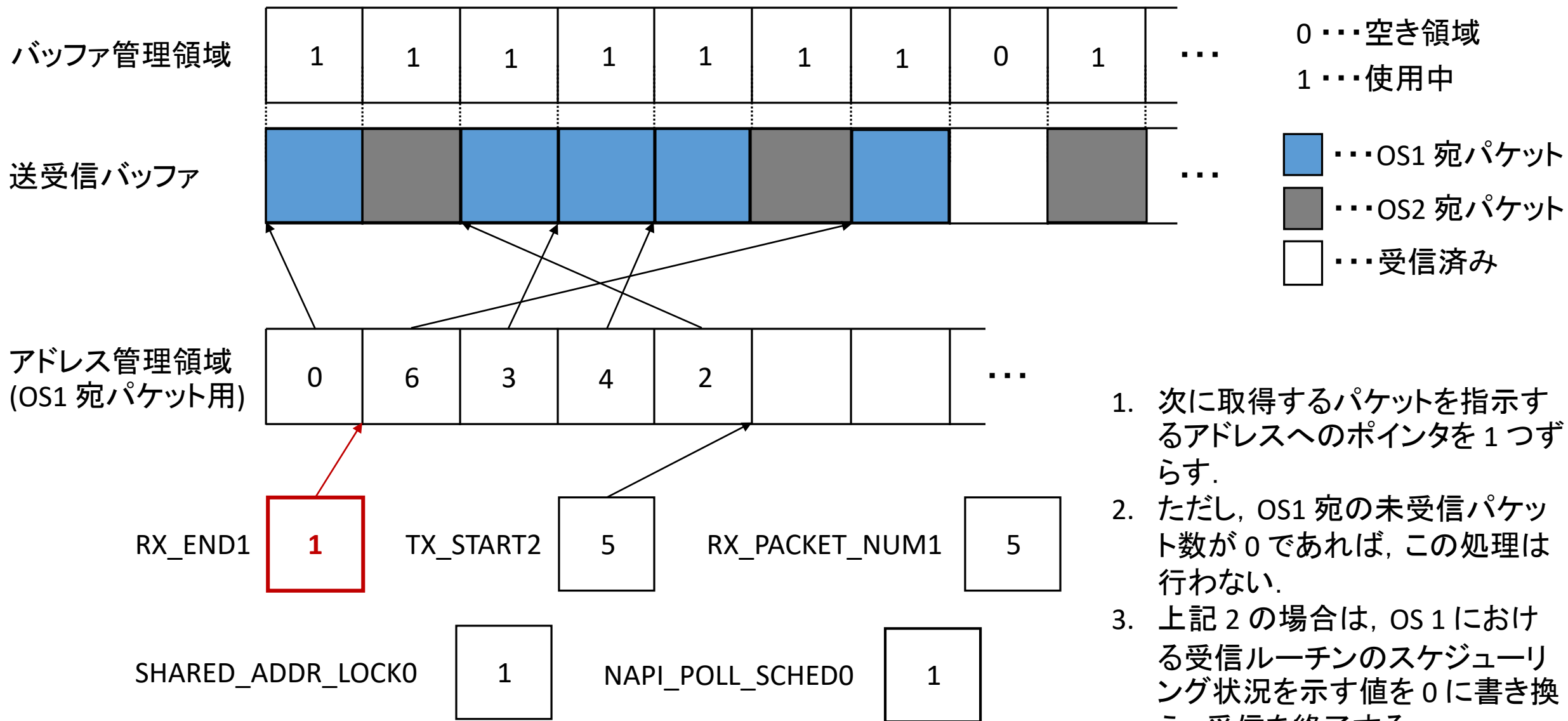
OS2からOS1宛のパケットを共有メモリから取得する様子(1/7)



OS2からOS1宛のパケットを共有メモリから取得する様子(2/7)

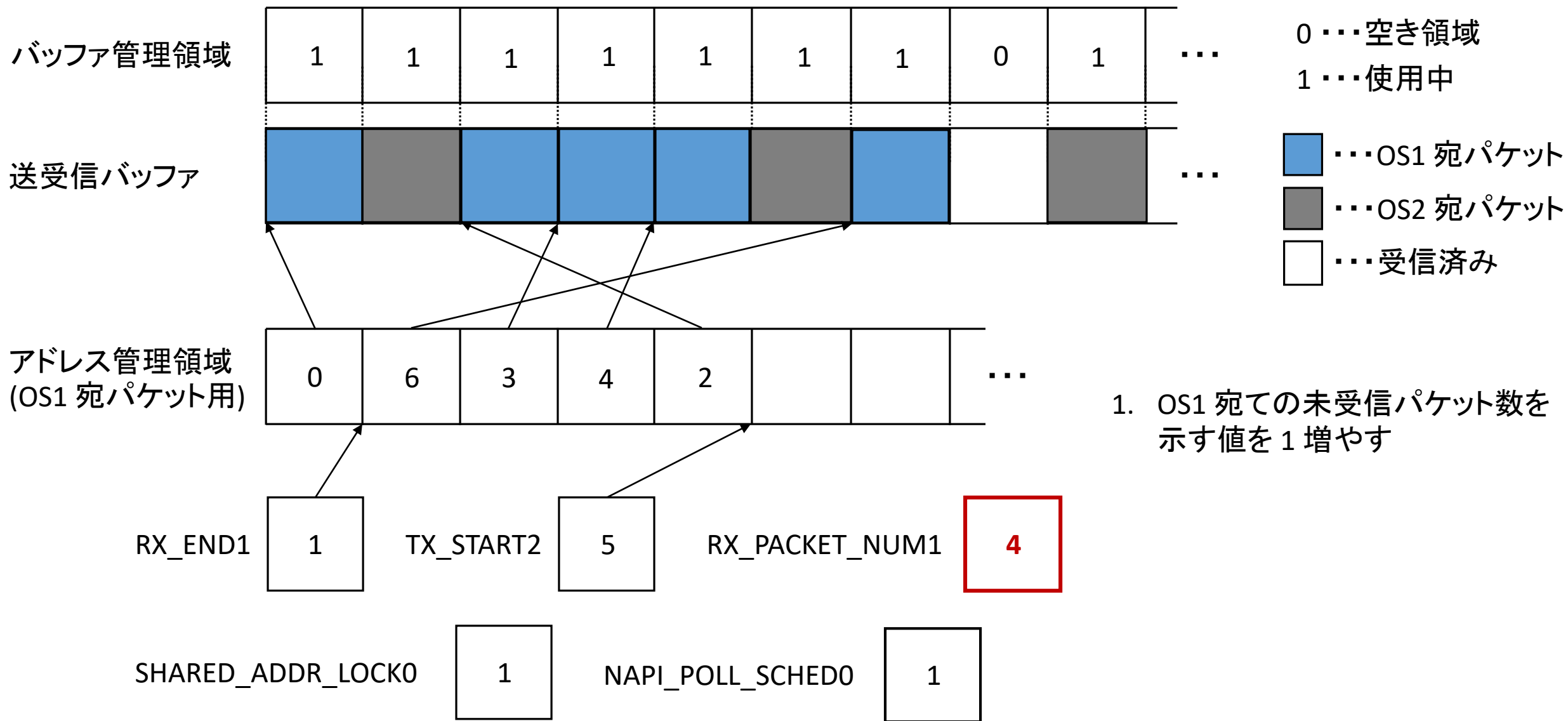


OS2からOS1宛のパケットを共有メモリから取得する様子(3/7)

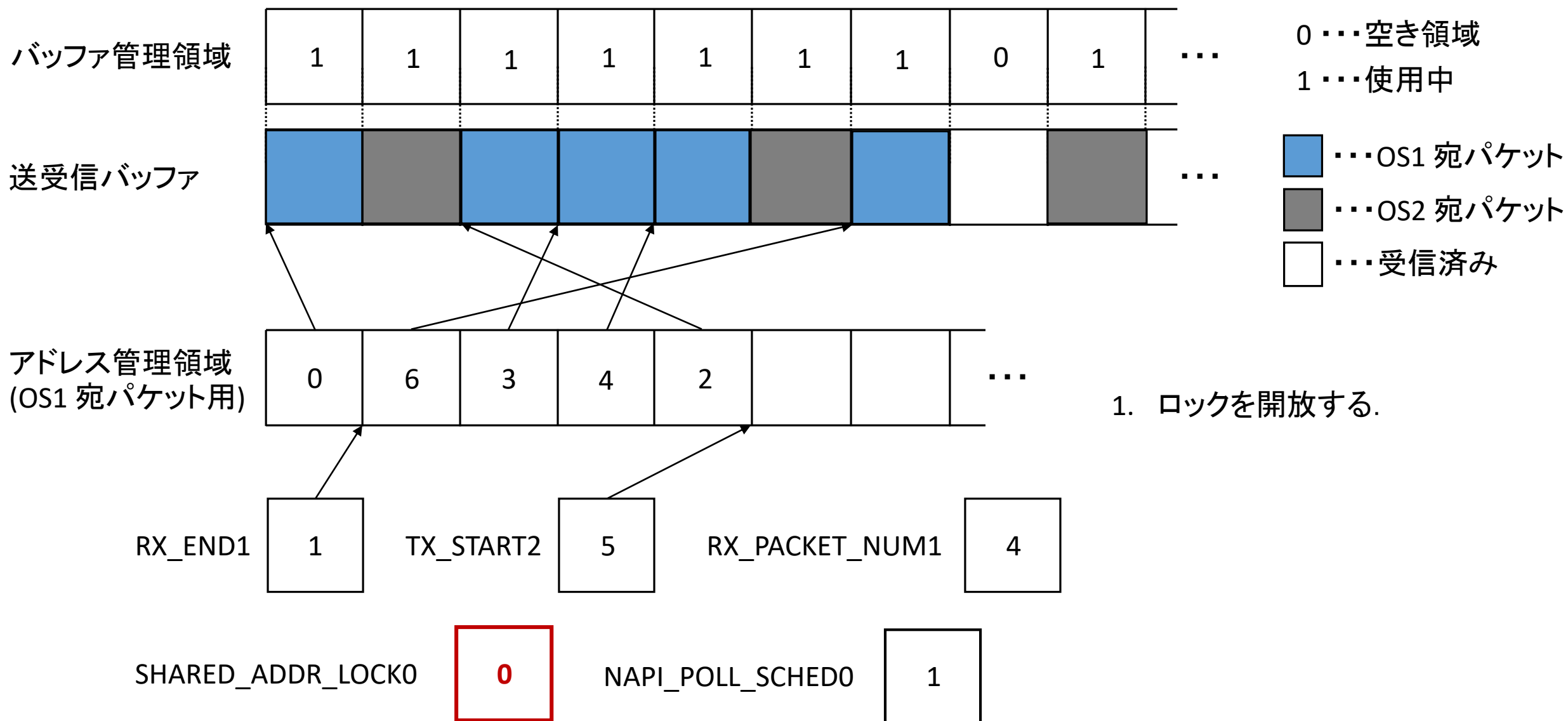


1. 次に取得するパケットを指示するアドレスへのポインタを1つずらす。
2. ただし、OS1 宛の未受信パケット数が0であれば、この処理は行わない。
3. 上記2の場合は、OS1における受信ルーチンのスケジューリング状況を示す値を0に書き換え、受信を終了する。

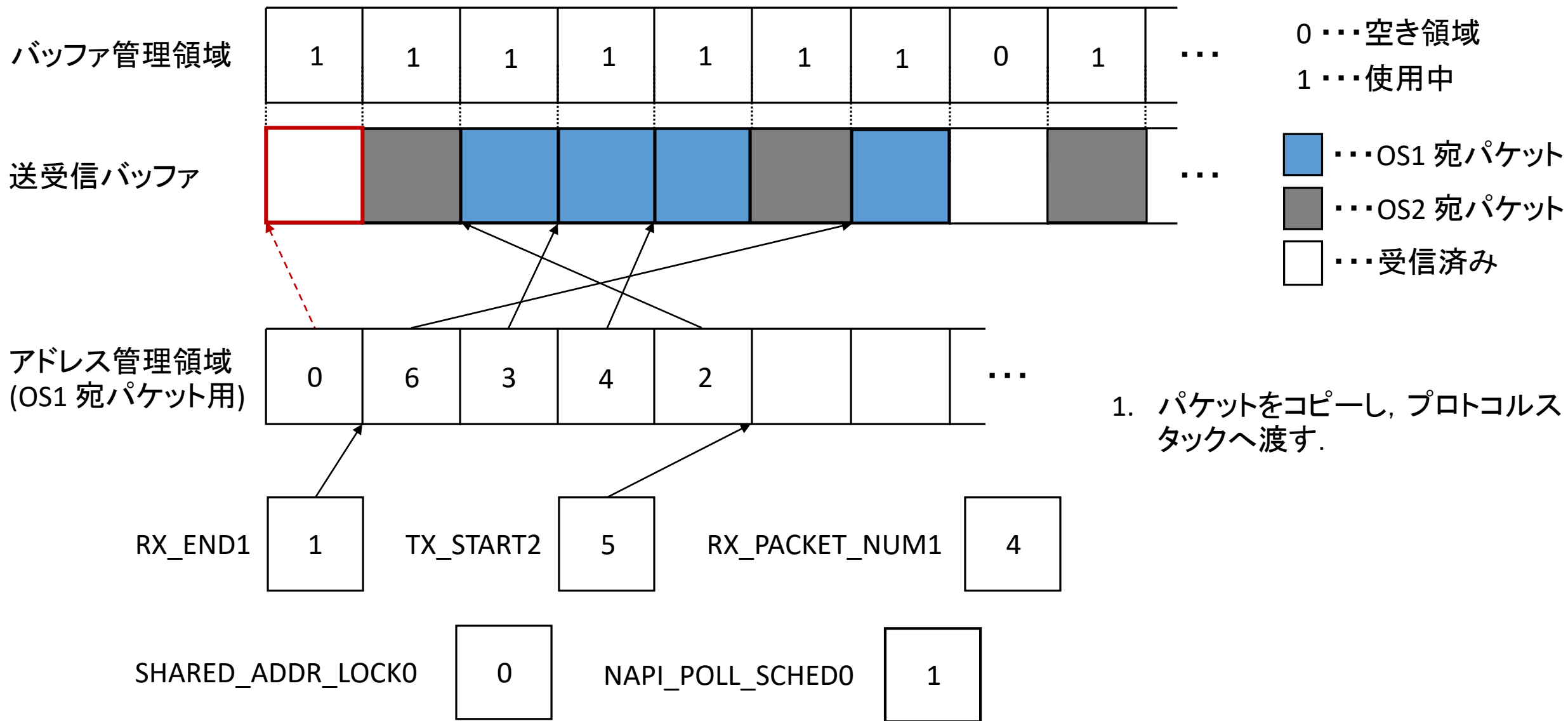
OS2からOS1宛のパケットを共有メモリから取得する様子(4/7)



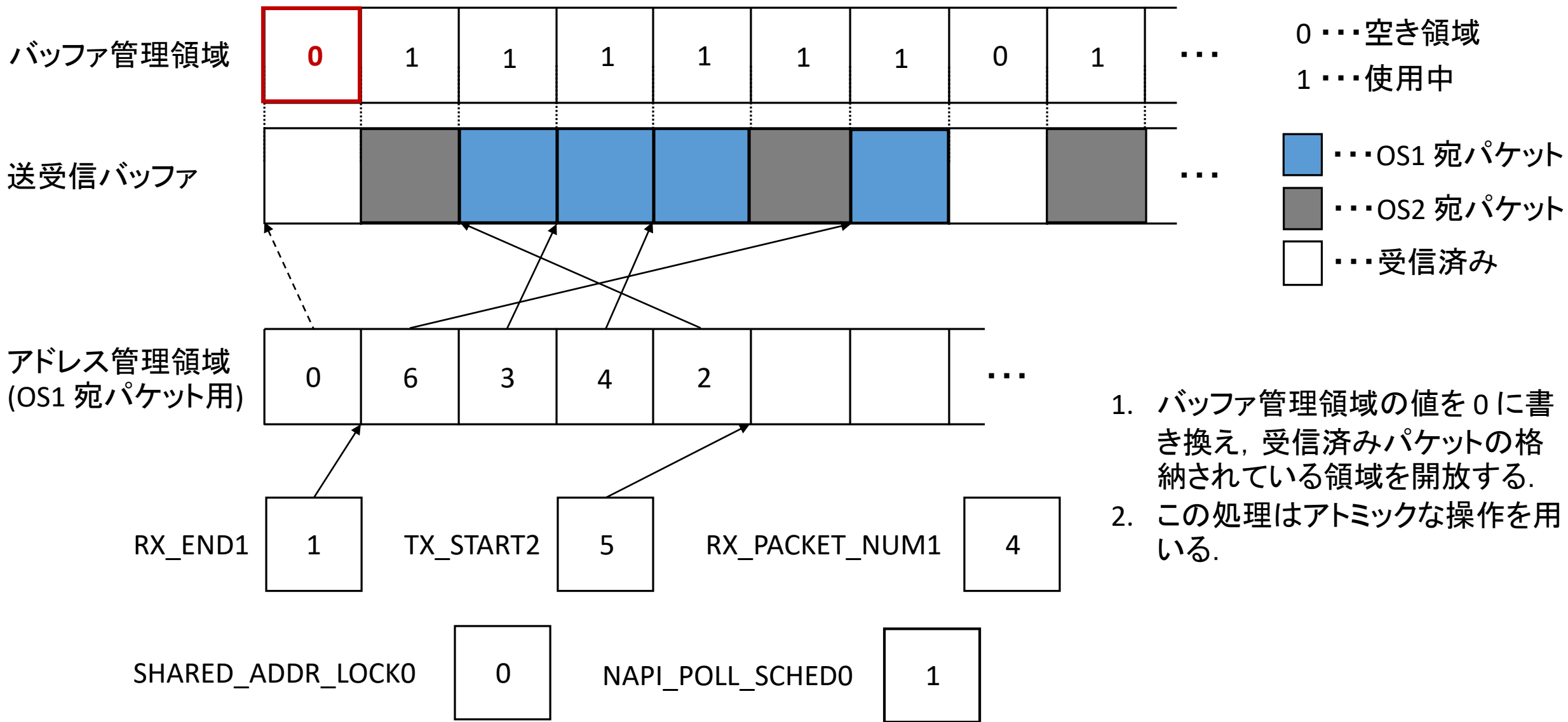
OS2からOS1宛のパケットを共有メモリから取得する様子(5/7)



OS2からOS1宛のパケットを共有メモリから取得する様子(6/7)



OS2からOS1宛のパケットを共有メモリから取得する様子(7/7)



1. バッファ管理領域の値を 0 に書き換え, 受信済みパケットの格納されている領域を開放する.
2. この処理はアトミックな操作を用いる.