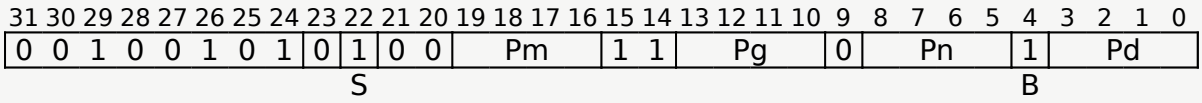
## BRKPBS

Break before first true condition, propagating from previous partition and setting the condition flags

If the last active element of the first source predicate is false then set the destination predicate to all-false. Otherwise sets destination predicate elements up to but not including the first active and true source element to true, then sets subsequent elements to false. Inactive elements in the destination predicate register are set to zero. Sets the first (N), none (Z), ! last (C) condition flags based on the predicate result, and the V flag to zero.




**BRKPBS <Pd>.B, <Pg>/Z, <Pn>.B, <Pm>.B**

```
if !HaveSVE() && !HaveSME() then UNDEFINED;
constant integer esize = 8;
integer g = UInt(Pg);
integer n = UInt(Pn);
integer m = UInt(Pm);
integer d = UInt(Pd);
boolean setflags = TRUE;
```

### Assembler Symbols

| | |
|---|---|
| <Pd> | Is the name of the destination scalable predicate register, encoded in the "Pd" field. |
| <Pg> | Is the name of the governing scalable predicate register, encoded in the "Pg" field. |
| <Pn> | Is the name of the first source scalable predicate register, encoded in the "Pn" field. |
| <Pm> | Is the name of the second source scalable predicate register, encoded in the "Pm" field. |

### Operation

```
CheckSVEEnabled();
constant integer VL = CurrentVL;
constant integer PL = VL DIV 8;
constant integer elements = VL DIV esize;
bits(PL) mask = P[g, PL];
bits(PL) operand1 = P[n, PL];
bits(PL) operand2 = P[m, PL];
bits(PL) result;
boolean last = (LastActive(mask, operand1, 8) == '1');
```

```
for e = 0 to elements-1
    if ActivePredicateElement(mask, e, 8) then
        last = last && (!ActivePredicateElement(operand2, e, 8));
        bit pbit = if last then '1' else '0';
        Elem[result, e, 1] = ZeroExtend(pbit, 1);
    else
        Elem[result, e, 1] = ZeroExtend('0', 1);

if setflags then
    PSTATE.<N,Z,C,V> = PredTest(mask, result, esize);
P[d, PL] = result;
```

**Operational information**

If FEAT_SME is implemented and the PE is in Streaming SVE mode, then any subsequent instruction which is dependent on the NZCV condition flags written by this instruction might be significantly delayed.

---



Internal version only: isa v33.64, AdvSIMD v29.12, pseudocode no_diffs_2023_09_RC2, sve v2023-06_rel ; Build timestamp: 2023-09-18T17:56