

Published: 07/12/67

Identification

Corrigenda, BD.9.01  
R. M. Graham, M. A. Padlipsky

- passim Error return label arguments are no longer used. In all cases in which they appeared, they are replaced by non-zero values in the error code arguments if errors arose in the called procedures. A zero value in an error code indicates successful completion.
- p.3 The second sentence of step 1, currently ending "and 'validation level'", should end  
  
return location (from oldsp|20), and "validation level".
- p.3 References to the Segment Housekeeping Module should be taken as references to the Segment Management Module.
- p.3 The last two sentences of step 1 should be replaced by  
  
Each entry in <rtn\_stk> is of variable length, in order to allow the Fault Interceptor to use it for secure storage of machine conditions; a diagram of <rtn\_stk> is to be found in Figure 6. The index of the last (most recent) entry is kept in <rtn\_stk>|0; this index is known as the "invocation number" (see BD.9.00).
- p.4 The portion of step 2 at the top of the page should read  
  
call get\_ring(address,ring,new\_ring,type,  
error\_code);  
  
where address is a pointer to the location being transferred to, ring is the ring number of the faulting procedure, new\_ring is a return argument which will be set to the ring number of the target procedure, type is a return argument which indicates whether address is a gate or a door (cf. BD.9.00,BD.9.05)

and `err_code` is a return argument which will be set to a non-zero value if the attempted crossing is illegal - the specific value indicating the specific kind of illegality. (After determining that the ring relationships are permissible, `get_ring` also checks that the specific transfer at hand is directed at a legitimate entry point - or "gate". The file system maintains lists of gates for segments; see also BG.9.00.)

p.5 The following should be appended to the first paragraph of step c:

However, in order to prevent an access fault during the return sequence when the bases are restored, the old stack pointer must not be preserved in the new frame. (It points to a segment which will in general be inaccessible from the new ring.) Therefore, `newsp|6` and `newsp|7` are overwritten, the new value being `newsp`.

p.11 An updated calling sequence for `appendb` will be found in BG.8.02.

p.11 In the first line of the last paragraph on the page, "set to 8" should be "set to point to `<stack_n>|8`".

p.17 Add Figure 6, attached.

Figure 6: Structure of <rtn\_stk>

