Addendum for TriCore Arch Manual, Vol2, V1.3.8, pages 493/494

UPDFL
Update Flags

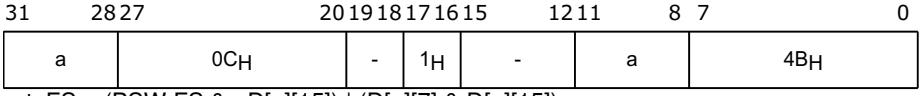
Description

The UPDFL instruction takes two 8-bit data fields from D[a], and uses them to update the PSW user flag bits (PSW [31:24]) that the FPU uses to store its exception flags and rounding mode in. D[a][15:8] is the update mask field; a '1' in a given bit position indicates that the corresponding PSW user flag bit is to be updated. D[a][7:0] is the update value field. These bits supply the values to be written to the PSW user flags bits, in the positions specified by the mask field.

Example: Changing the current PSW[25:24] (Rounding mode) to round toward +∞, without modifying any of the current exception flag settings, can be accomplished by loading the literal value 0301_H into register D[0], and issuing the instruction, UPDFL D[0].

UPDFL can be used to create exceptions that cause asynchronous traps to be taken.

UPDFL **D[a] (RR)**



set_FS = (PSW.FS & ~D[a][15]) | (D[a][7] & D[a][15]);

set_FI = (PSW.FI & ~D[a][14]) | (D[a][6] & D[a][14]);

set_FV = (PSW.FV & ~D[a][13]) | (D[a][5] & D[a][13]);

set_FZ = (PSW.FZ & ~D[a][12]) | (D[a][4] & D[a][12]);

set_FU = (PSW.FU & ~D[a][11]) | (D[a][3] & D[a][11]);

set_FX = (PSW.FX & ~D[a][10]) | (D[a][2] & D[a][10]);

set_RM = (PSW.RM & ~D[a][9:8]) | (D[a][1:0] & D[a][9:8]);

PSW.[31:24] = {set_FS, set_FI, set_FV, set_FZ, set_FU, set_FX, set_RM};

Exception Flags

FS	PSW.FS = set_FS;
FI	PSW.FI = set_FI;
FV	PSW.FV = set_FV;
FZ	PSW.FZ = set_FZ;
FU	PSW.FU = set_FU;
FX	PSW.FX = set_FX;

Addendum for TriCore Arch Manual, Vol2, V1.3.8, pages 493/494

Examples

```
updfl      d1
```

See Also

-