
MODULE *SPMDRoot*

A Single Program Multi Data root finish See *FinishState.RootFinishSPMD* for the actual implementation

EXTENDS *Sequences, Integers*

VARIABLES *fid, fstates, msgs, thrds, mseq, p0adoptSet*
 CONSTANTS *PLACE, MXFINISHES, PROG_HOME, MXTHREADS, NBLOCKS, MXSTMTS*
 INSTANCE *Commons*

$Alloc(type, here, parent, root) \triangleq$ parent not used here
 $\wedge fstates[fid].status = \text{"unused"}$
 $\wedge fstates' = [fstates \text{ EXCEPT } ![fid].id = fid,$
 $\phantom{\wedge fstates' = [fstates \text{ EXCEPT } } ![fid].count = 1,$
 $\phantom{\wedge fstates' = [fstates \text{ EXCEPT } } ![fid].status = \text{"waiting"},$
 $\phantom{\wedge fstates' = [fstates \text{ EXCEPT } } ![fid].type = type,$
 $\phantom{\wedge fstates' = [fstates \text{ EXCEPT } } ![fid].here = here,$
 $\phantom{\wedge fstates' = [fstates \text{ EXCEPT } } ![fid].root = root]$

$NotifySubActivitySpawn(dst) \triangleq$
 $\wedge fstates' = [fstates \text{ EXCEPT } ![fid].count = @ + 1]$

$NotifySubActivitySpawnError(dst) \triangleq \text{FALSE}$

$NotifyRemoteActivityCreation(src, activity, inMsg) \triangleq$
 $\wedge fstates' = fstates$ always true in SPMD finish
 $\wedge RecvMsg(inMsg)$

$NotifyLocalActivitySpawnAndCreation(here, activity) \triangleq$
 $\wedge fstates' = [fstates \text{ EXCEPT } ![fid].count = @ + 1]$

$LastActivity \triangleq$
 $\wedge fstates[fid].count = 1$

$NotifyActivityTermination \triangleq$
 $\wedge fstates[fid].count > 0$
 $\wedge \text{IF } LastActivity = \text{TRUE}$
 $\phantom{\wedge \text{IF } LastActivity = \text{TRUE}} \text{ THEN } fstates' = [fstates \text{ EXCEPT } ![fid].count = @ - 1,$
 $\phantom{\wedge \text{IF } LastActivity = \text{TRUE}} \phantom{\text{ THEN } } ![fid].status = \text{"finished"}]$
 $\phantom{\wedge \text{IF } LastActivity = \text{TRUE}} \text{ ELSE } fstates' = [fstates \text{ EXCEPT } ![fid].count = @ - 1]$

same as *NotifyActivityTermination* but saves the exceptions too

$NotifyActivityTerminationExcs(excs) \triangleq$
 $\wedge fstates[fid].count > 0$
 $\wedge \text{IF } LastActivity = \text{TRUE}$
 $\phantom{\wedge \text{IF } LastActivity = \text{TRUE}} \text{ THEN } fstates' = [fstates \text{ EXCEPT } ![fid].count = @ - 1,$
 $\phantom{\wedge \text{IF } LastActivity = \text{TRUE}} \phantom{\text{ THEN } } ![fid].status = \text{"finished"},$
 $\phantom{\wedge \text{IF } LastActivity = \text{TRUE}} \phantom{\text{ THEN } } ![fid].excs = @ \circ excs]$

ELSE $fstates' = [fstates \text{ EXCEPT } ![fid].count = @ - 1,$
 $![fid].excxs = @ \circ excxs]$

$PushException(e) \triangleq$
 $\wedge fstates' = [fstates \text{ EXCEPT } ![fid].excxs = Append(@, e)]$

$SendTermMsg \triangleq$
 $\wedge fstates' = [fstates \text{ EXCEPT } ![fid].status = \text{"forgotten"}]$
 $\wedge msgs' = msgs$
 $\wedge mseq' = mseq$

$ProcessChildTermMsg(msg) \triangleq$
 LET $here \triangleq fstates[fid].here$
 $src \triangleq msg.src$
 $excxs \triangleq msg.excxs$
 IN $\wedge NotifyActivityTerminationExcxs(excxs)$
 $\wedge RecvMsg(msg)$

\ * Modification History
 \ * Last modified *Mon Nov 06 19:13:59 AEDT 2017* by *u5482878*
 \ * Created *Wed Sep 13 12:16:33 AEST 2017* by *u5482878*