Contents

1 MoveToORPType Theory

Built: 10 June 2018
Parent Theories: indexedLists, patternMatches

1.1 Datatypes

slCommand = pltForm | pltMove | pltSecureHalt | complete
| incomplete

slOutput = MoveToORP | PLTForm | PLTMove | PLTSecureHalt
| Complete | unAuthorized | unAuthenticated

slState = MOVE_TO_ORP | PLT_FORM | PLT_MOVE | PLT_SECURE_HALT
| COMPLETE

stateRole = PlatoonLeader

1.2 Theorems

[slCommand_distinct_clauses]

⊢ pltForm ≠ pltMove ∧ pltForm ≠ pltSecureHalt ∧
pltForm ≠ complete ∧ pltForm ≠ incomplete ∧
pltMove ≠ pltSecureHalt ∧ pltMove ≠ complete ∧
pltMove ≠ incomplete ∧ pltSecureHalt ≠ complete ∧
pltSecureHalt ≠ incomplete ∧ complete ≠ incomplete

[slOutput_distinct_clauses]

⊢ MoveToORP ≠ PLTForm ∧ MoveToORP ≠ PLTMove ∧
MoveToORP ≠ PLTSecureHalt ∧ MoveToORP ≠ Complete ∧
MoveToORP ≠ unAuthorized ∧ MoveToORP ≠ unAuthenticated ∧
PLTForm ≠ PLTMove ∧ PLTForm ≠ PLTSecureHalt ∧
PLTForm ≠ Complete ∧ PLTForm ≠ unAuthorized ∧
PLTForm ≠ unAuthenticated ∧ PLTMove ≠ PLTSecureHalt ∧
PLTMove ≠ Complete ∧ PLTMove ≠ unAuthorized ∧
PLTMove ≠ unAuthenticated ∧ PLTSecureHalt ≠ Complete ∧
PLTSecureHalt ≠ unAuthorized ∧
PLTSecureHalt ≠ unAuthenticated ∧ Complete ≠ unAuthorized ∧
Complete ≠ unAuthenticated ∧ unAuthorized ≠ unAuthenticated

[slState_distinct_clauses]

⊢ MOVE_TO_ORP ≠ PLT_FORM ∧ MOVE_TO_ORP ≠ PLT_MOVE ∧
MOVE_TO_ORP ≠ PLT_SECURE_HALT ∧ MOVE_TO_ORP ≠ COMPLETE ∧
PLT_FORM ≠ PLT_MOVE ∧ PLT_FORM ≠ PLT_SECURE_HALT ∧
PLT_FORM ≠ COMPLETE ∧ PLT_MOVE ≠ PLT_SECURE_HALT ∧
PLT_MOVE ≠ COMPLETE ∧ PLT_SECURE_HALT ≠ COMPLETE

2 ssmMoveToORP Theory

Built: 10 June 2018
Parent Theories: MoveToORPType, ssm11, OMNIType

2.1 Definitions

[secContextMoveToORP_def]

⊢ ∀ *cmd*.
secContextMoveToORP *cmd* =
[Name PlatoonLeader controls prop (SOME (SLc *cmd*))]

[ssmMoveToORPStateInterp_def]

⊢ ∀ *state*. ssmMoveToORPStateInterp *state* = TT

2.2 Theorems

[authTestMoveToORP_cmd_reject_lemma]

⊢ ∀ *cmd*. ¬authTestMoveToORP (prop (SOME *cmd*))

[authTestMoveToORP_def]

⊢ (authTestMoveToORP (Name PlatoonLeader says prop *cmd*) ⟺ T) ∧
(authTestMoveToORP TT ⟺ F) ∧ (authTestMoveToORP FF ⟺ F) ∧
(authTestMoveToORP (prop *v*) ⟺ F) ∧
(authTestMoveToORP (notf v_1) ⟺ F) ∧
(authTestMoveToORP (v_2 andf v_3) ⟺ F) ∧
(authTestMoveToORP (v_4 orf v_5) ⟺ F) ∧
(authTestMoveToORP (v_6 impf v_7) ⟺ F) ∧
(authTestMoveToORP (v_8 eqf v_9) ⟺ F) ∧
(authTestMoveToORP (v_{10} says TT) ⟺ F) ∧
(authTestMoveToORP (v_{10} says FF) ⟺ F) ∧
(authTestMoveToORP (*v133* meet *v134* says prop v_{66}) ⟺ F) ∧
(authTestMoveToORP (*v135* quoting *v136* says prop v_{66}) ⟺ F) ∧
(authTestMoveToORP (v_{10} says notf v_{67}) ⟺ F) ∧
(authTestMoveToORP (v_{10} says (v_{68} andf v_{69})) ⟺ F) ∧
(authTestMoveToORP (v_{10} says (v_{70} orf v_{71})) ⟺ F) ∧
(authTestMoveToORP (v_{10} says (v_{72} impf v_{73})) ⟺ F) ∧
(authTestMoveToORP (v_{10} says (v_{74} eqf v_{75})) ⟺ F) ∧
(authTestMoveToORP (v_{10} says v_{76} says v_{77}) ⟺ F) ∧
(authTestMoveToORP (v_{10} says v_{78} speaks_for v_{79}) ⟺ F) ∧
(authTestMoveToORP (v_{10} says v_{80} controls v_{81}) ⟺ F) ∧
(authTestMoveToORP (v_{10} says reps v_{82} v_{83} v_{84}) ⟺ F) ∧
(authTestMoveToORP (v_{10} says v_{85} domi v_{86}) ⟺ F) ∧
(authTestMoveToORP (v_{10} says v_{87} eqi v_{88}) ⟺ F) ∧
(authTestMoveToORP (v_{10} says v_{89} doms v_{90}) ⟺ F) ∧
(authTestMoveToORP (v_{10} says v_{91} eqs v_{92}) ⟺ F) ∧
(authTestMoveToORP (v_{10} says v_{93} eqn v_{94}) ⟺ F) ∧

(authTestMoveToORP (v_{10} says v_{95} lte v_{96}) $\iff$ F) $\wedge$
(authTestMoveToORP (v_{10} says v_{97} lt v_{98}) $\iff$ F) $\wedge$
(authTestMoveToORP (v_{12} speaks_for v_{13}) $\iff$ F) $\wedge$
(authTestMoveToORP (v_{14} controls v_{15}) $\iff$ F) $\wedge$
(authTestMoveToORP (reps v_{16} v_{17} v_{18}) $\iff$ F) $\wedge$
(authTestMoveToORP (v_{19} domi v_{20}) $\iff$ F) $\wedge$
(authTestMoveToORP (v_{21} eqi v_{22}) $\iff$ F) $\wedge$
(authTestMoveToORP (v_{23} doms v_{24}) $\iff$ F) $\wedge$
(authTestMoveToORP (v_{25} eqs v_{26}) $\iff$ F) $\wedge$
(authTestMoveToORP (v_{27} eqn v_{28}) $\iff$ F) $\wedge$
(authTestMoveToORP (v_{29} lte v_{30}) $\iff$ F) $\wedge$
(authTestMoveToORP (v_{31} lt v_{32}) $\iff$ F)

[authTestMoveToORP_ind]

$\vdash \forall P.$
(∀ *cmd*. P (Name PlatoonLeader says prop *cmd*)) $\wedge$ P TT $\wedge$
P FF $\wedge$ (∀ v. P (prop v)) $\wedge$ (∀ v_1. P (notf v_1)) $\wedge$
(∀ v_2 v_3. P (v_2 andf v_3)) $\wedge$ (∀ v_4 v_5. P (v_4 orf v_5)) $\wedge$
(∀ v_6 v_7. P (v_6 impf v_7)) $\wedge$ (∀ v_8 v_9. P (v_8 eqf v_9)) $\wedge$
(∀ v_{10}. P (v_{10} says TT)) $\wedge$ (∀ v_{10}. P (v_{10} says FF)) $\wedge$
(∀ *v133* *v134* v_{66}. P (*v133* meet *v134* says prop v_{66})) $\wedge$
(∀ *v135* *v136* v_{66}. P (*v135* quoting *v136* says prop v_{66})) $\wedge$
(∀ v_{10} v_{67}. P (v_{10} says notf v_{67})) $\wedge$
(∀ v_{10} v_{68} v_{69}. P (v_{10} says (v_{68} andf v_{69}))) $\wedge$
(∀ v_{10} v_{70} v_{71}. P (v_{10} says (v_{70} orf v_{71}))) $\wedge$
(∀ v_{10} v_{72} v_{73}. P (v_{10} says (v_{72} impf v_{73}))) $\wedge$
(∀ v_{10} v_{74} v_{75}. P (v_{10} says (v_{74} eqf v_{75}))) $\wedge$
(∀ v_{10} v_{76} v_{77}. P (v_{10} says v_{76} says v_{77})) $\wedge$
(∀ v_{10} v_{78} v_{79}. P (v_{10} says v_{78} speaks_for v_{79})) $\wedge$
(∀ v_{10} v_{80} v_{81}. P (v_{10} says v_{80} controls v_{81})) $\wedge$
(∀ v_{10} v_{82} v_{83} v_{84}. P (v_{10} says reps v_{82} v_{83} v_{84})) $\wedge$
(∀ v_{10} v_{85} v_{86}. P (v_{10} says v_{85} domi v_{86})) $\wedge$
(∀ v_{10} v_{87} v_{88}. P (v_{10} says v_{87} eqi v_{88})) $\wedge$
(∀ v_{10} v_{89} v_{90}. P (v_{10} says v_{89} doms v_{90})) $\wedge$
(∀ v_{10} v_{91} v_{92}. P (v_{10} says v_{91} eqs v_{92})) $\wedge$
(∀ v_{10} v_{93} v_{94}. P (v_{10} says v_{93} eqn v_{94})) $\wedge$
(∀ v_{10} v_{95} v_{96}. P (v_{10} says v_{95} lte v_{96})) $\wedge$
(∀ v_{10} v_{97} v_{98}. P (v_{10} says v_{97} lt v_{98})) $\wedge$
(∀ v_{12} v_{13}. P (v_{12} speaks_for v_{13})) $\wedge$
(∀ v_{14} v_{15}. P (v_{14} controls v_{15})) $\wedge$
(∀ v_{16} v_{17} v_{18}. P (reps v_{16} v_{17} v_{18})) $\wedge$
(∀ v_{19} v_{20}. P (v_{19} domi v_{20})) $\wedge$
(∀ v_{21} v_{22}. P (v_{21} eqi v_{22})) $\wedge$
(∀ v_{23} v_{24}. P (v_{23} doms v_{24})) $\wedge$
(∀ v_{25} v_{26}. P (v_{25} eqs v_{26})) $\wedge$ (∀ v_{27} v_{28}. P (v_{27} eqn v_{28})) $\wedge$
(∀ v_{29} v_{30}. P (v_{29} lte v_{30})) $\wedge$ (∀ v_{31} v_{32}. P (v_{31} lt v_{32})) $\Rightarrow$
∀ v. P v

[moveToORPNS_def]

```
⊢ (moveToORPNS MOVE_TO_ORP (exec (SLc pltForm)) = PLT_FORM) ∧
  (moveToORPNS MOVE_TO_ORP (exec (SLc incomplete)) =
   MOVE_TO_ORP) ∧
  (moveToORPNS PLT_FORM (exec (SLc pltMove)) = PLT_MOVE) ∧
  (moveToORPNS PLT_FORM (exec (SLc incomplete)) = PLT_FORM) ∧
  (moveToORPNS PLT_MOVE (exec (SLc pltSecureHalt)) =
   PLT_SECURE_HALT) ∧
  (moveToORPNS PLT_MOVE (exec (SLc incomplete)) = PLT_MOVE) ∧
  (moveToORPNS PLT_SECURE_HALT (exec (SLc complete)) =
   COMPLETE) ∧
  (moveToORPNS PLT_SECURE_HALT (exec (SLc incomplete)) =
   PLT_SECURE_HALT) ∧ (moveToORPNS s (trap (SLc cmd)) = s) ∧
  (moveToORPNS s (discard (SLc cmd)) = s)
```

[moveToORPNS_ind]

```
⊢ ∀P.
    P MOVE_TO_ORP (exec (SLc pltForm)) ∧
    P MOVE_TO_ORP (exec (SLc incomplete)) ∧
    P PLT_FORM (exec (SLc pltMove)) ∧
    P PLT_FORM (exec (SLc incomplete)) ∧
    P PLT_MOVE (exec (SLc pltSecureHalt)) ∧
    P PLT_MOVE (exec (SLc incomplete)) ∧
    P PLT_SECURE_HALT (exec (SLc complete)) ∧
    P PLT_SECURE_HALT (exec (SLc incomplete)) ∧
    (∀s cmd. P s (trap (SLc cmd))) ∧
    (∀s cmd. P s (discard (SLc cmd))) ∧
    (∀s v6. P s (discard (ESCc v6))) ∧
    (∀s v9. P s (trap (ESCc v9))) ∧
    (∀v12. P MOVE_TO_ORP (exec (ESCc v12))) ∧
    P MOVE_TO_ORP (exec (SLc pltMove)) ∧
    P MOVE_TO_ORP (exec (SLc pltSecureHalt)) ∧
    P MOVE_TO_ORP (exec (SLc complete)) ∧
    (∀v15. P PLT_FORM (exec (ESCc v15))) ∧
    P PLT_FORM (exec (SLc pltForm)) ∧
    P PLT_FORM (exec (SLc pltSecureHalt)) ∧
    P PLT_FORM (exec (SLc complete)) ∧
    (∀v18. P PLT_MOVE (exec (ESCc v18))) ∧
    P PLT_MOVE (exec (SLc pltForm)) ∧
    P PLT_MOVE (exec (SLc pltMove)) ∧
    P PLT_MOVE (exec (SLc complete)) ∧
    (∀v21. P PLT_SECURE_HALT (exec (ESCc v21))) ∧
    P PLT_SECURE_HALT (exec (SLc pltForm)) ∧
    P PLT_SECURE_HALT (exec (SLc pltMove)) ∧
    P PLT_SECURE_HALT (exec (SLc pltSecureHalt)) ∧
    (∀v23. P COMPLETE (exec v23)) ⇒
   ∀v v1. P v v1
```

[moveToORPOut_def]

```
⊢ (moveToORPOut MOVE_TO_ORP (exec (SLc pltForm)) = PLTForm) ∧
  (moveToORPOut MOVE_TO_ORP (exec (SLc incomplete)) =
```

```
    MoveToORP) ∧
   (moveToORPOut PLT_FORM (exec (SLc pltMove)) = PLTMove) ∧
   (moveToORPOut PLT_FORM (exec (SLc incomplete)) = PLTForm) ∧
   (moveToORPOut PLT_MOVE (exec (SLc pltSecureHalt)) =
    PLTSecureHalt) ∧
   (moveToORPOut PLT_MOVE (exec (SLc incomplete)) = PLTMove) ∧
   (moveToORPOut PLT_SECURE_HALT (exec (SLc complete)) =
    Complete) ∧
   (moveToORPOut PLT_SECURE_HALT (exec (SLc incomplete)) =
    PLTSecureHalt) ∧
   (moveToORPOut s (trap (SLc cmd)) = unAuthorized) ∧
   (moveToORPOut s (discard (SLc cmd)) = unAuthenticated)
```

[moveToORPOut_ind]

```
⊢ ∀P.
    P MOVE_TO_ORP (exec (SLc pltForm)) ∧
    P MOVE_TO_ORP (exec (SLc incomplete)) ∧
    P PLT_FORM (exec (SLc pltMove)) ∧
    P PLT_FORM (exec (SLc incomplete)) ∧
    P PLT_MOVE (exec (SLc pltSecureHalt)) ∧
    P PLT_MOVE (exec (SLc incomplete)) ∧
    P PLT_SECURE_HALT (exec (SLc complete)) ∧
    P PLT_SECURE_HALT (exec (SLc incomplete)) ∧
    (∀s cmd. P s (trap (SLc cmd))) ∧
    (∀s cmd. P s (discard (SLc cmd))) ∧
    (∀s v6. P s (discard (ESCc v6))) ∧
    (∀s v9. P s (trap (ESCc v9))) ∧
    (∀v12. P MOVE_TO_ORP (exec (ESCc v12))) ∧
    P MOVE_TO_ORP (exec (SLc pltMove)) ∧
    P MOVE_TO_ORP (exec (SLc pltSecureHalt)) ∧
    P MOVE_TO_ORP (exec (SLc complete)) ∧
    (∀v15. P PLT_FORM (exec (ESCc v15))) ∧
    P PLT_FORM (exec (SLc pltForm)) ∧
    P PLT_FORM (exec (SLc pltSecureHalt)) ∧
    P PLT_FORM (exec (SLc complete)) ∧
    (∀v18. P PLT_MOVE (exec (ESCc v18))) ∧
    P PLT_MOVE (exec (SLc pltForm)) ∧
    P PLT_MOVE (exec (SLc pltMove)) ∧
    P PLT_MOVE (exec (SLc complete)) ∧
    (∀v21. P PLT_SECURE_HALT (exec (ESCc v21))) ∧
    P PLT_SECURE_HALT (exec (SLc pltForm)) ∧
    P PLT_SECURE_HALT (exec (SLc pltMove)) ∧
    P PLT_SECURE_HALT (exec (SLc pltSecureHalt)) ∧
    (∀v23. P COMPLETE (exec v23)) ⇒
    ∀v v1. P v v1
```

[PlatoonLeader_exec_slCommand_justified_thm]

```
⊢ ∀NS Out M Oi Os.
    TR (M,Oi,Os) (exec (SLc slCommand))
```

```
      (CFG authTestMoveToORP ssmMoveToORPStateInterp
         (secContextMoveToORP slCommand)
         (Name PlatoonLeader says prop (SOME (SLc slCommand))::
              ins) s outs)
      (CFG authTestMoveToORP ssmMoveToORPStateInterp
         (secContextMoveToORP slCommand) ins
         (NS s (exec (SLc slCommand)))
         (Out s (exec (SLc slCommand))::outs)) ⟺
    authTestMoveToORP
      (Name PlatoonLeader says prop (SOME (SLc slCommand))) ∧
    CFGInterpret (M,Oi,Os)
      (CFG authTestMoveToORP ssmMoveToORPStateInterp
         (secContextMoveToORP slCommand)
         (Name PlatoonLeader says prop (SOME (SLc slCommand))::
              ins) s outs) ∧
    (M,Oi,Os) sat prop (SOME (SLc slCommand))
```

[PlatoonLeader_slCommand_lemma]

```
⊢ CFGInterpret (M,Oi,Os)
    (CFG authTestMoveToORP ssmMoveToORPStateInterp
       (secContextMoveToORP slCommand)
       (Name PlatoonLeader says prop (SOME (SLc slCommand))::
            ins) s outs) ⇒
  (M,Oi,Os) sat prop (SOME (SLc slCommand))
```

Index