Contents

1 projectTypes Theory

Built: 27 December 2018
Parent Theories: indexedLists, patternMatches

1.1 Datatypes

```
commands = PlatoonLeaderCOM platoonLeaderCom | OmniCOM omniCom

omniCom = none | omniNA

output = ContingencyPlan | MoveToORP | ConductORP | FormST
       | ReturnToUnit | Complete | NoActionTaken
       | UnAuthenticated | UnAuthorized

platoonLeaderCom = contingencyPlan | moveToORP | conductORP
                 | formST | returnToUnit | complete

principal = PlatoonLeader | Omni

state = ORP_RECON | CONTINGENCY_PLAN | MOVE_TO_ORP | CONDUCT_ORP
      | FORM_ST | RETURN_TO_UNIT | COMPLETE
```

1.2 Theorems

[commands_distinct_clauses]

```
⊢ ∀a′ a. PlatoonLeaderCOM a ≠ OmniCOM a′
```

[commands_one_one]

```
⊢ (∀a a′.
     (PlatoonLeaderCOM a = PlatoonLeaderCOM a′) ⟺ (a = a′)) ∧
  ∀a a′. (OmniCOM a = OmniCOM a′) ⟺ (a = a′)
```

[omniCom_distinct_clauses]

```
⊢ none ≠ omniNA
```

[output_distinct_clauses]

```
⊢ ContingencyPlan ≠ MoveToORP ∧ ContingencyPlan ≠ ConductORP ∧
  ContingencyPlan ≠ FormST ∧ ContingencyPlan ≠ ReturnToUnit ∧
  ContingencyPlan ≠ Complete ∧
  ContingencyPlan ≠ NoActionTaken ∧
  ContingencyPlan ≠ UnAuthenticated ∧
  ContingencyPlan ≠ UnAuthorized ∧ MoveToORP ≠ ConductORP ∧
  MoveToORP ≠ FormST ∧ MoveToORP ≠ ReturnToUnit ∧
  MoveToORP ≠ Complete ∧ MoveToORP ≠ NoActionTaken ∧
  MoveToORP ≠ UnAuthenticated ∧ MoveToORP ≠ UnAuthorized ∧
  ConductORP ≠ FormST ∧ ConductORP ≠ ReturnToUnit ∧
  ConductORP ≠ Complete ∧ ConductORP ≠ NoActionTaken ∧
  ConductORP ≠ UnAuthenticated ∧ ConductORP ≠ UnAuthorized ∧
  FormST ≠ ReturnToUnit ∧ FormST ≠ Complete ∧
  FormST ≠ NoActionTaken ∧ FormST ≠ UnAuthenticated ∧
  FormST ≠ UnAuthorized ∧ ReturnToUnit ≠ Complete ∧
  ReturnToUnit ≠ NoActionTaken ∧
  ReturnToUnit ≠ UnAuthenticated ∧
  ReturnToUnit ≠ UnAuthorized ∧ Complete ≠ NoActionTaken ∧
  Complete ≠ UnAuthenticated ∧ Complete ≠ UnAuthorized ∧
  NoActionTaken ≠ UnAuthenticated ∧
  NoActionTaken ≠ UnAuthorized ∧ UnAuthenticated ≠ UnAuthorized
```

[platoonLeaderCom_distinct_clauses]

```
⊢ contingencyPlan ≠ moveToORP ∧ contingencyPlan ≠ conductORP ∧
  contingencyPlan ≠ formST ∧ contingencyPlan ≠ returnToUnit ∧
  contingencyPlan ≠ complete ∧ moveToORP ≠ conductORP ∧
  moveToORP ≠ formST ∧ moveToORP ≠ returnToUnit ∧
  moveToORP ≠ complete ∧ conductORP ≠ formST ∧
  conductORP ≠ returnToUnit ∧ conductORP ≠ complete ∧
  formST ≠ returnToUnit ∧ formST ≠ complete ∧
  returnToUnit ≠ complete
```

[principal_distinct_clauses]

```
⊢ PlatoonLeader ≠ Omni
```

[state_distinct_clauses]

```
⊢ ORP_RECON ≠ CONTINGENCY_PLAN ∧ ORP_RECON ≠ MOVE_TO_ORP ∧
  ORP_RECON ≠ CONDUCT_ORP ∧ ORP_RECON ≠ FORM_ST ∧
  ORP_RECON ≠ RETURN_TO_UNIT ∧ ORP_RECON ≠ COMPLETE ∧
  CONTINGENCY_PLAN ≠ MOVE_TO_ORP ∧
  CONTINGENCY_PLAN ≠ CONDUCT_ORP ∧ CONTINGENCY_PLAN ≠ FORM_ST ∧
  CONTINGENCY_PLAN ≠ RETURN_TO_UNIT ∧
  CONTINGENCY_PLAN ≠ COMPLETE ∧ MOVE_TO_ORP ≠ CONDUCT_ORP ∧
  MOVE_TO_ORP ≠ FORM_ST ∧ MOVE_TO_ORP ≠ RETURN_TO_UNIT ∧
  MOVE_TO_ORP ≠ COMPLETE ∧ CONDUCT_ORP ≠ FORM_ST ∧
  CONDUCT_ORP ≠ RETURN_TO_UNIT ∧ CONDUCT_ORP ≠ COMPLETE ∧
  FORM_ST ≠ RETURN_TO_UNIT ∧ FORM_ST ≠ COMPLETE ∧
  RETURN_TO_UNIT ≠ COMPLETE
```

2 projectUtilities Theory

Built: 27 December 2018

Parent Theories: projectTypes, satList

2.1 Theorems

[getOmniCOM_def]

```
⊢ (getOmniCOM [] = NONE) ∧
  (∀ xs cmd.
     getOmniCOM (SOME (OmniCOM cmd)::xs) =
     SOME (OmniCOM cmd)) ∧
  (∀ xs. getOmniCOM (NONE::xs) = getOmniCOM xs) ∧
  ∀ xs v4.
    getOmniCOM (SOME (PlatoonLeaderCOM v4)::xs) = getOmniCOM xs
```

[getOmniCOM_ind]

```
⊢ ∀ P.
    P [] ∧ (∀ cmd xs. P (SOME (OmniCOM cmd)::xs)) ∧
    (∀ xs. P xs ⇒ P (NONE::xs)) ∧
    (∀ v4 xs. P xs ⇒ P (SOME (PlatoonLeaderCOM v4)::xs)) ⇒
    ∀ v. P v
```

[getOmniCOMx_def]

```
⊢ (getOmniCOMx [] = NONE) ∧
  (∀ xs cmd.
     getOmniCOMx
       (Name Omni says prop (SOME (OmniCOM cmd))::xs) =
     SOME (OmniCOM cmd)) ∧
  (∀ xs. getOmniCOMx (TT::xs) = getOmniCOMx xs) ∧
  (∀ xs. getOmniCOMx (FF::xs) = getOmniCOMx xs) ∧
  (∀ xs v2. getOmniCOMx (prop v2::xs) = getOmniCOMx xs) ∧
  (∀ xs v3. getOmniCOMx (notf v3::xs) = getOmniCOMx xs) ∧
  (∀ xs v5 v4. getOmniCOMx (v4 andf v5::xs) = getOmniCOMx xs) ∧
  (∀ xs v7 v6. getOmniCOMx (v6 orf v7::xs) = getOmniCOMx xs) ∧
  (∀ xs v9 v8. getOmniCOMx (v8 impf v9::xs) = getOmniCOMx xs) ∧
  (∀ xs v11 v10.
     getOmniCOMx (v10 eqf v11::xs) = getOmniCOMx xs) ∧
  (∀ xs v12. getOmniCOMx (v12 says TT::xs) = getOmniCOMx xs) ∧
  (∀ xs v12. getOmniCOMx (v12 says FF::xs) = getOmniCOMx xs) ∧
  (∀ xs v134.
     getOmniCOMx (Name v134 says prop NONE::xs) =
     getOmniCOMx xs) ∧
  (∀ xs v144.
     getOmniCOMx
       (Name PlatoonLeader says prop (SOME v144)::xs) =
     getOmniCOMx xs) ∧
  (∀ xs v146.
     getOmniCOMx
       (Name Omni says prop (SOME (PlatoonLeaderCOM v146))::
            xs) =
     getOmniCOMx xs) ∧
  (∀ xs v68 v136 v135.
     getOmniCOMx (v135 meet v136 says prop v68::xs) =
     getOmniCOMx xs) ∧
```

(∀ xs v_{68} $v138$ $v137$.
getOmniCOMx ($v137$ quoting $v138$ says prop v_{68}::xs) =
getOmniCOMx xs) ∧
(∀ xs v_{69} v_{12}.
getOmniCOMx (v_{12} says notf v_{69}::xs) = getOmniCOMx xs) ∧
(∀ xs v_{71} v_{70} v_{12}.
getOmniCOMx (v_{12} says (v_{70} andf v_{71})::xs) =
getOmniCOMx xs) ∧
(∀ xs v_{73} v_{72} v_{12}.
getOmniCOMx (v_{12} says (v_{72} orf v_{73})::xs) =
getOmniCOMx xs) ∧
(∀ xs v_{75} v_{74} v_{12}.
getOmniCOMx (v_{12} says (v_{74} impf v_{75})::xs) =
getOmniCOMx xs) ∧
(∀ xs v_{77} v_{76} v_{12}.
getOmniCOMx (v_{12} says (v_{76} eqf v_{77})::xs) =
getOmniCOMx xs) ∧
(∀ xs v_{79} v_{78} v_{12}.
getOmniCOMx (v_{12} says v_{78} says v_{79}::xs) =
getOmniCOMx xs) ∧
(∀ xs v_{81} v_{80} v_{12}.
getOmniCOMx (v_{12} says v_{80} speaks_for v_{81}::xs) =
getOmniCOMx xs) ∧
(∀ xs v_{83} v_{82} v_{12}.
getOmniCOMx (v_{12} says v_{82} controls v_{83}::xs) =
getOmniCOMx xs) ∧
(∀ xs v_{86} v_{85} v_{84} v_{12}.
getOmniCOMx (v_{12} says reps v_{84} v_{85} v_{86}::xs) =
getOmniCOMx xs) ∧
(∀ xs v_{88} v_{87} v_{12}.
getOmniCOMx (v_{12} says v_{87} domi v_{88}::xs) =
getOmniCOMx xs) ∧
(∀ xs v_{90} v_{89} v_{12}.
getOmniCOMx (v_{12} says v_{89} eqi v_{90}::xs) = getOmniCOMx xs) ∧
(∀ xs v_{92} v_{91} v_{12}.
getOmniCOMx (v_{12} says v_{91} doms v_{92}::xs) =
getOmniCOMx xs) ∧
(∀ xs v_{94} v_{93} v_{12}.
getOmniCOMx (v_{12} says v_{93} eqs v_{94}::xs) = getOmniCOMx xs) ∧
(∀ xs v_{96} v_{95} v_{12}.
getOmniCOMx (v_{12} says v_{95} eqn v_{96}::xs) = getOmniCOMx xs) ∧
(∀ xs v_{98} v_{97} v_{12}.
getOmniCOMx (v_{12} says v_{97} lte v_{98}::xs) = getOmniCOMx xs) ∧
(∀ xs v_{99} v_{12} $v100$.
getOmniCOMx (v_{12} says v_{99} lt $v100$::xs) = getOmniCOMx xs) ∧
(∀ xs v_{15} v_{14}.
getOmniCOMx (v_{14} speaks_for v_{15}::xs) = getOmniCOMx xs) ∧
(∀ xs v_{17} v_{16}.
getOmniCOMx (v_{16} controls v_{17}::xs) = getOmniCOMx xs) ∧

```
(∀xs v20 v19 v18.
   getOmniCOMx (reps v18 v19 v20::xs) = getOmniCOMx xs) ∧
(∀xs v22 v21.
   getOmniCOMx (v21 domi v22::xs) = getOmniCOMx xs) ∧
(∀xs v24 v23.
   getOmniCOMx (v23 eqi v24::xs) = getOmniCOMx xs) ∧
(∀xs v26 v25.
   getOmniCOMx (v25 doms v26::xs) = getOmniCOMx xs) ∧
(∀xs v28 v27.
   getOmniCOMx (v27 eqs v28::xs) = getOmniCOMx xs) ∧
(∀xs v30 v29.
   getOmniCOMx (v29 eqn v30::xs) = getOmniCOMx xs) ∧
(∀xs v32 v31.
   getOmniCOMx (v31 lte v32::xs) = getOmniCOMx xs) ∧
∀xs v34 v33. getOmniCOMx (v33 lt v34::xs) = getOmniCOMx xs
```

[getOmniCOMx_ind]

```
⊢ ∀P.
    P [] ∧
    (∀cmd xs.
       P (Name Omni says prop (SOME (OmniCOM cmd))::xs)) ∧
    (∀xs. P xs ⇒ P (TT::xs)) ∧ (∀xs. P xs ⇒ P (FF::xs)) ∧
    (∀v2 xs. P xs ⇒ P (prop v2::xs)) ∧
    (∀v3 xs. P xs ⇒ P (notf v3::xs)) ∧
    (∀v4 v5 xs. P xs ⇒ P (v4 andf v5::xs)) ∧
    (∀v6 v7 xs. P xs ⇒ P (v6 orf v7::xs)) ∧
    (∀v8 v9 xs. P xs ⇒ P (v8 impf v9::xs)) ∧
    (∀v10 v11 xs. P xs ⇒ P (v10 eqf v11::xs)) ∧
    (∀v12 xs. P xs ⇒ P (v12 says TT::xs)) ∧
    (∀v12 xs. P xs ⇒ P (v12 says FF::xs)) ∧
    (∀v134 xs. P xs ⇒ P (Name v134 says prop NONE::xs)) ∧
    (∀v144 xs.
       P xs ⇒
       P (Name PlatoonLeader says prop (SOME v144)::xs)) ∧
    (∀v146 xs.
       P xs ⇒
       P
         (Name Omni says prop (SOME (PlatoonLeaderCOM v146))::
              xs)) ∧
    (∀v135 v136 v68 xs.
       P xs ⇒ P (v135 meet v136 says prop v68::xs)) ∧
    (∀v137 v138 v68 xs.
       P xs ⇒ P (v137 quoting v138 says prop v68::xs)) ∧
    (∀v12 v69 xs. P xs ⇒ P (v12 says notf v69::xs)) ∧
    (∀v12 v70 v71 xs. P xs ⇒ P (v12 says (v70 andf v71)::xs)) ∧
    (∀v12 v72 v73 xs. P xs ⇒ P (v12 says (v72 orf v73)::xs)) ∧
    (∀v12 v74 v75 xs. P xs ⇒ P (v12 says (v74 impf v75)::xs)) ∧
    (∀v12 v76 v77 xs. P xs ⇒ P (v12 says (v76 eqf v77)::xs)) ∧
    (∀v12 v78 v79 xs. P xs ⇒ P (v12 says v78 says v79::xs)) ∧
```

$$\begin{array}{l}
(\forall v_{12}\ v_{80}\ v_{81}\ xs.\\
\quad P\ xs\ \Rightarrow\ P\ (v_{12}\ \texttt{says}\ v_{80}\ \texttt{speaks\_for}\ v_{81}\texttt{::}xs))\ \land\\
(\forall v_{12}\ v_{82}\ v_{83}\ xs.\\
\quad P\ xs\ \Rightarrow\ P\ (v_{12}\ \texttt{says}\ v_{82}\ \texttt{controls}\ v_{83}\texttt{::}xs))\ \land\\
(\forall v_{12}\ v_{84}\ v_{85}\ v_{86}\ xs.\\
\quad P\ xs\ \Rightarrow\ P\ (v_{12}\ \texttt{says}\ \texttt{reps}\ v_{84}\ v_{85}\ v_{86}\texttt{::}xs))\ \land\\
(\forall v_{12}\ v_{87}\ v_{88}\ xs.\ P\ xs\ \Rightarrow\ P\ (v_{12}\ \texttt{says}\ v_{87}\ \texttt{domi}\ v_{88}\texttt{::}xs))\ \land\\
(\forall v_{12}\ v_{89}\ v_{90}\ xs.\ P\ xs\ \Rightarrow\ P\ (v_{12}\ \texttt{says}\ v_{89}\ \texttt{eqi}\ v_{90}\texttt{::}xs))\ \land\\
(\forall v_{12}\ v_{91}\ v_{92}\ xs.\ P\ xs\ \Rightarrow\ P\ (v_{12}\ \texttt{says}\ v_{91}\ \texttt{doms}\ v_{92}\texttt{::}xs))\ \land\\
(\forall v_{12}\ v_{93}\ v_{94}\ xs.\ P\ xs\ \Rightarrow\ P\ (v_{12}\ \texttt{says}\ v_{93}\ \texttt{eqs}\ v_{94}\texttt{::}xs))\ \land\\
(\forall v_{12}\ v_{95}\ v_{96}\ xs.\ P\ xs\ \Rightarrow\ P\ (v_{12}\ \texttt{says}\ v_{95}\ \texttt{eqn}\ v_{96}\texttt{::}xs))\ \land\\
(\forall v_{12}\ v_{97}\ v_{98}\ xs.\ P\ xs\ \Rightarrow\ P\ (v_{12}\ \texttt{says}\ v_{97}\ \texttt{lte}\ v_{98}\texttt{::}xs))\ \land\\
(\forall v_{12}\ v_{99}\ v100\ xs.\ P\ xs\ \Rightarrow\ P\ (v_{12}\ \texttt{says}\ v_{99}\ \texttt{lt}\ v100\texttt{::}xs))\ \land\\
(\forall v_{14}\ v_{15}\ xs.\ P\ xs\ \Rightarrow\ P\ (v_{14}\ \texttt{speaks\_for}\ v_{15}\texttt{::}xs))\ \land\\
(\forall v_{16}\ v_{17}\ xs.\ P\ xs\ \Rightarrow\ P\ (v_{16}\ \texttt{controls}\ v_{17}\texttt{::}xs))\ \land\\
(\forall v_{18}\ v_{19}\ v_{20}\ xs.\ P\ xs\ \Rightarrow\ P\ (\texttt{reps}\ v_{18}\ v_{19}\ v_{20}\texttt{::}xs))\ \land\\
(\forall v_{21}\ v_{22}\ xs.\ P\ xs\ \Rightarrow\ P\ (v_{21}\ \texttt{domi}\ v_{22}\texttt{::}xs))\ \land\\
(\forall v_{23}\ v_{24}\ xs.\ P\ xs\ \Rightarrow\ P\ (v_{23}\ \texttt{eqi}\ v_{24}\texttt{::}xs))\ \land\\
(\forall v_{25}\ v_{26}\ xs.\ P\ xs\ \Rightarrow\ P\ (v_{25}\ \texttt{doms}\ v_{26}\texttt{::}xs))\ \land\\
(\forall v_{27}\ v_{28}\ xs.\ P\ xs\ \Rightarrow\ P\ (v_{27}\ \texttt{eqs}\ v_{28}\texttt{::}xs))\ \land\\
(\forall v_{29}\ v_{30}\ xs.\ P\ xs\ \Rightarrow\ P\ (v_{29}\ \texttt{eqn}\ v_{30}\texttt{::}xs))\ \land\\
(\forall v_{31}\ v_{32}\ xs.\ P\ xs\ \Rightarrow\ P\ (v_{31}\ \texttt{lte}\ v_{32}\texttt{::}xs))\ \land\\
(\forall v_{33}\ v_{34}\ xs.\ P\ xs\ \Rightarrow\ P\ (v_{33}\ \texttt{lt}\ v_{34}\texttt{::}xs))\ \Rightarrow\\
\forall v.\ P\ v
\end{array}$$

[getPlatoonLeaderCOM_def]

$$\begin{array}{l}
\vdash\ (\texttt{getPlatoonLeaderCOM}\ \texttt{[]}\ =\ \texttt{NONE})\ \land\\
\quad (\forall xs\ cmd.\\
\quad\quad \texttt{getPlatoonLeaderCOM}\ (\texttt{SOME}\ (\texttt{PlatoonLeaderCOM}\ cmd)\texttt{::}xs)\ =\\
\quad\quad \texttt{SOME}\ (\texttt{PlatoonLeaderCOM}\ cmd))\ \land\\
\quad (\forall xs.\\
\quad\quad \texttt{getPlatoonLeaderCOM}\ (\texttt{NONE::}xs)\ =\ \texttt{getPlatoonLeaderCOM}\ xs)\ \land\\
\quad \forall xs\ v_5.\\
\quad\quad \texttt{getPlatoonLeaderCOM}\ (\texttt{SOME}\ (\texttt{OmniCOM}\ v_5)\texttt{::}xs)\ =\\
\quad\quad \texttt{getPlatoonLeaderCOM}\ xs
\end{array}$$

[getPlatoonLeaderCOM_ind]

$$\begin{array}{l}
\vdash\ \forall P.\\
\quad\quad P\ \texttt{[]}\ \land\ (\forall cmd\ xs.\ P\ (\texttt{SOME}\ (\texttt{PlatoonLeaderCOM}\ cmd)\texttt{::}xs))\ \land\\
\quad\quad (\forall xs.\ P\ xs\ \Rightarrow\ P\ (\texttt{NONE::}xs))\ \land\\
\quad\quad (\forall v_5\ xs.\ P\ xs\ \Rightarrow\ P\ (\texttt{SOME}\ (\texttt{OmniCOM}\ v_5)\texttt{::}xs))\ \Rightarrow\\
\quad\quad \forall v.\ P\ v
\end{array}$$

[getPlatoonLeaderCOMx_def]

$$\begin{array}{l}
\vdash\ (\texttt{getPlatoonLeaderCOMx}\ \texttt{[]}\ =\ \texttt{NONE})\ \land\\
\quad (\forall xs\ cmd.\\
\quad\quad \texttt{getPlatoonLeaderCOMx}\\
\quad\quad\quad (\texttt{Name PlatoonLeader says}
\end{array}$$

```
      prop (SOME (PlatoonLeaderCOM cmd))::xs) =
   SOME (PlatoonLeaderCOM cmd)) ∧
(∀ xs.
   getPlatoonLeaderCOMx (TT::xs) = getPlatoonLeaderCOMx xs) ∧
(∀ xs.
   getPlatoonLeaderCOMx (FF::xs) = getPlatoonLeaderCOMx xs) ∧
(∀ xs v2.
   getPlatoonLeaderCOMx (prop v2::xs) =
   getPlatoonLeaderCOMx xs) ∧
(∀ xs v3.
   getPlatoonLeaderCOMx (notf v3::xs) =
   getPlatoonLeaderCOMx xs) ∧
(∀ xs v5 v4.
   getPlatoonLeaderCOMx (v4 andf v5::xs) =
   getPlatoonLeaderCOMx xs) ∧
(∀ xs v7 v6.
   getPlatoonLeaderCOMx (v6 orf v7::xs) =
   getPlatoonLeaderCOMx xs) ∧
(∀ xs v9 v8.
   getPlatoonLeaderCOMx (v8 impf v9::xs) =
   getPlatoonLeaderCOMx xs) ∧
(∀ xs v11 v10.
   getPlatoonLeaderCOMx (v10 eqf v11::xs) =
   getPlatoonLeaderCOMx xs) ∧
(∀ xs v12.
   getPlatoonLeaderCOMx (v12 says TT::xs) =
   getPlatoonLeaderCOMx xs) ∧
(∀ xs v12.
   getPlatoonLeaderCOMx (v12 says FF::xs) =
   getPlatoonLeaderCOMx xs) ∧
(∀ xs v134.
   getPlatoonLeaderCOMx (Name v134 says prop NONE::xs) =
   getPlatoonLeaderCOMx xs) ∧
(∀ xs v147.
   getPlatoonLeaderCOMx
     (Name PlatoonLeader says prop (SOME (OmniCOM v147))::
          xs) =
   getPlatoonLeaderCOMx xs) ∧
(∀ xs v144.
   getPlatoonLeaderCOMx
     (Name Omni says prop (SOME v144)::xs) =
   getPlatoonLeaderCOMx xs) ∧
(∀ xs v68 v136 v135.
   getPlatoonLeaderCOMx (v135 meet v136 says prop v68::xs) =
   getPlatoonLeaderCOMx xs) ∧
(∀ xs v68 v138 v137.
   getPlatoonLeaderCOMx
     (v137 quoting v138 says prop v68::xs) =
   getPlatoonLeaderCOMx xs) ∧
```

```
(∀xs v69 v12.
   getPlatoonLeaderCOMx (v12 says notf v69::xs) =
   getPlatoonLeaderCOMx xs) ∧
(∀xs v71 v70 v12.
   getPlatoonLeaderCOMx (v12 says (v70 andf v71)::xs) =
   getPlatoonLeaderCOMx xs) ∧
(∀xs v73 v72 v12.
   getPlatoonLeaderCOMx (v12 says (v72 orf v73)::xs) =
   getPlatoonLeaderCOMx xs) ∧
(∀xs v75 v74 v12.
   getPlatoonLeaderCOMx (v12 says (v74 impf v75)::xs) =
   getPlatoonLeaderCOMx xs) ∧
(∀xs v77 v76 v12.
   getPlatoonLeaderCOMx (v12 says (v76 eqf v77)::xs) =
   getPlatoonLeaderCOMx xs) ∧
(∀xs v79 v78 v12.
   getPlatoonLeaderCOMx (v12 says v78 says v79::xs) =
   getPlatoonLeaderCOMx xs) ∧
(∀xs v81 v80 v12.
   getPlatoonLeaderCOMx (v12 says v80 speaks_for v81::xs) =
   getPlatoonLeaderCOMx xs) ∧
(∀xs v83 v82 v12.
   getPlatoonLeaderCOMx (v12 says v82 controls v83::xs) =
   getPlatoonLeaderCOMx xs) ∧
(∀xs v86 v85 v84 v12.
   getPlatoonLeaderCOMx (v12 says reps v84 v85 v86::xs) =
   getPlatoonLeaderCOMx xs) ∧
(∀xs v88 v87 v12.
   getPlatoonLeaderCOMx (v12 says v87 domi v88::xs) =
   getPlatoonLeaderCOMx xs) ∧
(∀xs v90 v89 v12.
   getPlatoonLeaderCOMx (v12 says v89 eqi v90::xs) =
   getPlatoonLeaderCOMx xs) ∧
(∀xs v92 v91 v12.
   getPlatoonLeaderCOMx (v12 says v91 doms v92::xs) =
   getPlatoonLeaderCOMx xs) ∧
(∀xs v94 v93 v12.
   getPlatoonLeaderCOMx (v12 says v93 eqs v94::xs) =
   getPlatoonLeaderCOMx xs) ∧
(∀xs v96 v95 v12.
   getPlatoonLeaderCOMx (v12 says v95 eqn v96::xs) =
   getPlatoonLeaderCOMx xs) ∧
(∀xs v98 v97 v12.
   getPlatoonLeaderCOMx (v12 says v97 lte v98::xs) =
   getPlatoonLeaderCOMx xs) ∧
(∀xs v99 v12 v100.
   getPlatoonLeaderCOMx (v12 says v99 lt v100::xs) =
   getPlatoonLeaderCOMx xs) ∧
(∀xs v15 v14.
   getPlatoonLeaderCOMx (v14 speaks_for v15::xs) =
   getPlatoonLeaderCOMx xs) ∧
(∀ xs v17 v16.
   getPlatoonLeaderCOMx (v16 controls v17::xs) =
   getPlatoonLeaderCOMx xs) ∧
(∀ xs v20 v19 v18.
   getPlatoonLeaderCOMx (reps v18 v19 v20::xs) =
   getPlatoonLeaderCOMx xs) ∧
(∀ xs v22 v21.
   getPlatoonLeaderCOMx (v21 domi v22::xs) =
   getPlatoonLeaderCOMx xs) ∧
(∀ xs v24 v23.
   getPlatoonLeaderCOMx (v23 eqi v24::xs) =
   getPlatoonLeaderCOMx xs) ∧
(∀ xs v26 v25.
   getPlatoonLeaderCOMx (v25 doms v26::xs) =
   getPlatoonLeaderCOMx xs) ∧
(∀ xs v28 v27.
   getPlatoonLeaderCOMx (v27 eqs v28::xs) =
   getPlatoonLeaderCOMx xs) ∧
(∀ xs v30 v29.
   getPlatoonLeaderCOMx (v29 eqn v30::xs) =
   getPlatoonLeaderCOMx xs) ∧
(∀ xs v32 v31.
   getPlatoonLeaderCOMx (v31 lte v32::xs) =
   getPlatoonLeaderCOMx xs) ∧
∀ xs v34 v33.
  getPlatoonLeaderCOMx (v33 lt v34::xs) =
  getPlatoonLeaderCOMx xs
```

[getPlatoonLeaderCOMx_ind]

```
 ⊢ ∀ P.
     P [] ∧
     (∀ cmd xs.
        P
          (Name PlatoonLeader says
           prop (SOME (PlatoonLeaderCOM cmd))::xs)) ∧
     (∀ xs. P xs ⇒ P (TT::xs)) ∧ (∀ xs. P xs ⇒ P (FF::xs)) ∧
     (∀ v2 xs. P xs ⇒ P (prop v2::xs)) ∧
     (∀ v3 xs. P xs ⇒ P (notf v3::xs)) ∧
     (∀ v4 v5 xs. P xs ⇒ P (v4 andf v5::xs)) ∧
     (∀ v6 v7 xs. P xs ⇒ P (v6 orf v7::xs)) ∧
     (∀ v8 v9 xs. P xs ⇒ P (v8 impf v9::xs)) ∧
     (∀ v10 v11 xs. P xs ⇒ P (v10 eqf v11::xs)) ∧
     (∀ v12 xs. P xs ⇒ P (v12 says TT::xs)) ∧
     (∀ v12 xs. P xs ⇒ P (v12 says FF::xs)) ∧
     (∀ v134 xs. P xs ⇒ P (Name v134 says prop NONE::xs)) ∧
     (∀ v147 xs.
        P xs ⇒
        P
          (Name PlatoonLeader says prop (SOME (OmniCOM v147))::
             xs)) ∧
     (∀ v144 xs.
        P xs ⇒ P (Name Omni says prop (SOME v144)::xs)) ∧
     (∀ v135 v136 v68 xs.
        P xs ⇒ P (v135 meet v136 says prop v68::xs)) ∧
     (∀ v137 v138 v68 xs.
        P xs ⇒ P (v137 quoting v138 says prop v68::xs)) ∧
     (∀ v12 v69 xs. P xs ⇒ P (v12 says notf v69::xs)) ∧
     (∀ v12 v70 v71 xs. P xs ⇒ P (v12 says (v70 andf v71)::xs)) ∧
     (∀ v12 v72 v73 xs. P xs ⇒ P (v12 says (v72 orf v73)::xs)) ∧
     (∀ v12 v74 v75 xs. P xs ⇒ P (v12 says (v74 impf v75)::xs)) ∧
     (∀ v12 v76 v77 xs. P xs ⇒ P (v12 says (v76 eqf v77)::xs)) ∧
     (∀ v12 v78 v79 xs. P xs ⇒ P (v12 says v78 says v79::xs)) ∧
     (∀ v12 v80 v81 xs.
        P xs ⇒ P (v12 says v80 speaks_for v81::xs)) ∧
     (∀ v12 v82 v83 xs.
        P xs ⇒ P (v12 says v82 controls v83::xs)) ∧
     (∀ v12 v84 v85 v86 xs.
        P xs ⇒ P (v12 says reps v84 v85 v86::xs)) ∧
     (∀ v12 v87 v88 xs. P xs ⇒ P (v12 says v87 domi v88::xs)) ∧
     (∀ v12 v89 v90 xs. P xs ⇒ P (v12 says v89 eqi v90::xs)) ∧
     (∀ v12 v91 v92 xs. P xs ⇒ P (v12 says v91 doms v92::xs)) ∧
     (∀ v12 v93 v94 xs. P xs ⇒ P (v12 says v93 eqs v94::xs)) ∧
     (∀ v12 v95 v96 xs. P xs ⇒ P (v12 says v95 eqn v96::xs)) ∧
     (∀ v12 v97 v98 xs. P xs ⇒ P (v12 says v97 lte v98::xs)) ∧
     (∀ v12 v99 v100 xs. P xs ⇒ P (v12 says v99 lt v100::xs)) ∧
     (∀ v14 v15 xs. P xs ⇒ P (v14 speaks_for v15::xs)) ∧
     (∀ v16 v17 xs. P xs ⇒ P (v16 controls v17::xs)) ∧
     (∀ v18 v19 v20 xs. P xs ⇒ P (reps v18 v19 v20::xs)) ∧
     (∀ v21 v22 xs. P xs ⇒ P (v21 domi v22::xs)) ∧
     (∀ v23 v24 xs. P xs ⇒ P (v23 eqi v24::xs)) ∧
     (∀ v25 v26 xs. P xs ⇒ P (v25 doms v26::xs)) ∧
     (∀ v27 v28 xs. P xs ⇒ P (v27 eqs v28::xs)) ∧
     (∀ v29 v30 xs. P xs ⇒ P (v29 eqn v30::xs)) ∧
     (∀ v31 v32 xs. P xs ⇒ P (v31 lte v32::xs)) ∧
     (∀ v33 v34 xs. P xs ⇒ P (v33 lt v34::xs)) ⇒
     ∀ v. P v
```

3 projectSM Theory

Built: 27 December 2018

Parent Theories: projectUtilities, ssm

3.1 Theorems

[NOut_def]

```
⊢ (NOut ORP_RECON (exec x) =
    if
      getPlatoonLeaderCOM x =
      SOME (PlatoonLeaderCOM contingencyPlan)
    then
      ContingencyPlan
    else NoActionTaken) ∧
   (NOut CONTINGENCY_PLAN (exec x) =
    if
      getPlatoonLeaderCOM x = SOME (PlatoonLeaderCOM moveToORP)
    then
      MoveToORP
    else NoActionTaken) ∧
   (NOut MOVE_TO_ORP (exec x) =
    if
      getPlatoonLeaderCOM x = SOME (PlatoonLeaderCOM conductORP)
    then
      ConductORP
    else NoActionTaken) ∧
   (NOut CONDUCT_ORP (exec x) =
    if
      getPlatoonLeaderCOM x = SOME (PlatoonLeaderCOM formST)
    then
      FormST
    else NoActionTaken) ∧
   (NOut FORM_ST (exec x) =
    if
      getPlatoonLeaderCOM x =
      SOME (PlatoonLeaderCOM returnToUnit)
    then
      ReturnToUnit
    else NoActionTaken) ∧
   (NOut RETURN_TO_UNIT (exec x) =
    if
      getPlatoonLeaderCOM x = SOME (PlatoonLeaderCOM complete)
    then
      Complete
    else NoActionTaken) ∧ (NOut s (trap v0) = UnAuthorized) ∧
   (NOut s (discard v1) = UnAuthenticated)
```

[NOut_ind]

```
⊢ ∀P.
     (∀x. P ORP_RECON (exec x)) ∧
     (∀x. P CONTINGENCY_PLAN (exec x)) ∧
     (∀x. P MOVE_TO_ORP (exec x)) ∧
     (∀x. P CONDUCT_ORP (exec x)) ∧ (∀x. P FORM_ST (exec x)) ∧
     (∀x. P RETURN_TO_UNIT (exec x)) ∧ (∀s v0. P s (trap v0)) ∧
     (∀s v1. P s (discard v1)) ∧ (∀v6. P COMPLETE (exec v6)) ⇒
     ∀v v1. P v v1
```

[NS_def]

```
 ⊢ (NS ORP_RECON (exec x) =
    if
      getPlatoonLeaderCOM x =
      SOME (PlatoonLeaderCOM contingencyPlan)
    then
      CONTINGENCY_PLAN
    else ORP_RECON) ∧
   (NS CONTINGENCY_PLAN (exec x) =
    if
      getPlatoonLeaderCOM x = SOME (PlatoonLeaderCOM moveToORP)
    then
      MOVE_TO_ORP
    else CONTINGENCY_PLAN) ∧
   (NS MOVE_TO_ORP (exec x) =
    if
      getPlatoonLeaderCOM x = SOME (PlatoonLeaderCOM conductORP)
    then
      CONDUCT_ORP
    else MOVE_TO_ORP) ∧
   (NS CONDUCT_ORP (exec x) =
    if
      getPlatoonLeaderCOM x = SOME (PlatoonLeaderCOM formST)
    then
      FORM_ST
    else CONDUCT_ORP) ∧
   (NS FORM_ST (exec x) =
    if
      getPlatoonLeaderCOM x =
      SOME (PlatoonLeaderCOM returnToUnit)
    then
      RETURN_TO_UNIT
    else FORM_ST) ∧
   (NS RETURN_TO_UNIT (exec x) =
    if
      getPlatoonLeaderCOM x = SOME (PlatoonLeaderCOM complete)
    then
      COMPLETE
    else RETURN_TO_UNIT) ∧ (NS s (trap v0) = s) ∧
   (NS s (discard v1) = s)
```

[NS_ind]

```
 ⊢ ∀P.
     (∀x. P ORP_RECON (exec x)) ∧
     (∀x. P CONTINGENCY_PLAN (exec x)) ∧
     (∀x. P MOVE_TO_ORP (exec x)) ∧
```

```
(∀x. P CONDUCT_ORP (exec x)) ∧ (∀x. P FORM_ST (exec x)) ∧
(∀x. P RETURN_TO_UNIT (exec x)) ∧ (∀s v0. P s (trap v0)) ∧
(∀s v1. P s (discard v1)) ∧ (∀v6. P COMPLETE (exec v6)) ⇒
∀v v1. P v v1
```

4 projectSecurity Theory

Built: 27 December 2018

Parent Theories: projectUtilities, ssm

4.1 Definitions

[globalAuth_def]

```
⊢ ∀x. globalAuth x = [TT]
```

[stateAuth_def]

```
⊢ ∀s x.
    stateAuth s x =
    if s = ORP_RECON then
      if
        getPlatoonLeaderCOMx x =
        SOME (PlatoonLeaderCOM contingencyPlan)
      then
        [Name PlatoonLeader controls
         prop (SOME (PlatoonLeaderCOM contingencyPlan))]
      else [prop NONE]
    else if s = CONTINGENCY_PLAN then
      if
        getPlatoonLeaderCOMx x =
        SOME (PlatoonLeaderCOM moveToORP)
      then
        [Name PlatoonLeader controls
         prop (SOME (PlatoonLeaderCOM moveToORP))]
      else [prop NONE]
    else if s = MOVE_TO_ORP then
      if
        getPlatoonLeaderCOMx x =
        SOME (PlatoonLeaderCOM conductORP)
      then
        [Name PlatoonLeader controls
         prop (SOME (PlatoonLeaderCOM conductORP))]
      else [prop NONE]
    else if s = CONDUCT_ORP then
      if
        getPlatoonLeaderCOMx x = SOME (PlatoonLeaderCOM formST)
      then
        [Name PlatoonLeader controls
```

```
         prop (SOME (PlatoonLeaderCOM formST))]
      else [prop NONE]
    else if s = FORM_ST then
      if
        getPlatoonLeaderCOMx x =
        SOME (PlatoonLeaderCOM returnToUnit)
      then
        [Name PlatoonLeader controls
         prop (SOME (PlatoonLeaderCOM returnToUnit))]
      else [prop NONE]
    else if s = RETURN_TO_UNIT then
      if
        getPlatoonLeaderCOMx x =
        SOME (PlatoonLeaderCOM complete)
      then
        [Name PlatoonLeader controls
         prop (SOME (PlatoonLeaderCOM complete))]
      else [prop NONE]
    else [prop NONE]
```

4.2 Theorems

[authentication_def]

```
 ⊢ (authentication
       (Name PlatoonLeader says
        prop (SOME (PlatoonLeaderCOM x′))) ⇔ T) ∧
   (authentication (Name Omni says prop (SOME (OmniCOM x))) ⇔
    T) ∧ (authentication TT ⇔ F) ∧ (authentication FF ⇔ F) ∧
   (authentication (prop v) ⇔ F) ∧
   (authentication (notf v1) ⇔ F) ∧
   (authentication (v2 andf v3) ⇔ F) ∧
   (authentication (v4 orf v5) ⇔ F) ∧
   (authentication (v6 impf v7) ⇔ F) ∧
   (authentication (v8 eqf v9) ⇔ F) ∧
   (authentication (Name v66 says TT) ⇔ F) ∧
   (authentication (Name v66 says FF) ⇔ F) ∧
   (authentication (Name v66 says prop NONE) ⇔ F) ∧
   (authentication
      (Name Omni says prop (SOME (PlatoonLeaderCOM v144))) ⇔
    F) ∧
   (authentication
      (Name PlatoonLeader says prop (SOME (OmniCOM v145))) ⇔
    F) ∧ (authentication (Name v66 says notf v77) ⇔ F) ∧
   (authentication (Name v66 says (v78 andf v79)) ⇔ F) ∧
   (authentication (Name v66 says (v80 orf v81)) ⇔ F) ∧
   (authentication (Name v66 says (v82 impf v83)) ⇔ F) ∧
   (authentication (Name v66 says (v84 eqf v85)) ⇔ F) ∧
   (authentication (Name v66 says v86 says v87) ⇔ F) ∧
   (authentication (Name v66 says v88 speaks_for v89) ⇔ F) ∧
```

```
(authentication (Name v66 says v90 controls v91) ⇔ F) ∧
(authentication (Name v66 says reps v92 v93 v94) ⇔ F) ∧
(authentication (Name v66 says v95 domi v96) ⇔ F) ∧
(authentication (Name v66 says v97 eqi v98) ⇔ F) ∧
(authentication (Name v66 says v99 doms v100) ⇔ F) ∧
(authentication (Name v66 says v101 eqs v102) ⇔ F) ∧
(authentication (Name v66 says v103 eqn v104) ⇔ F) ∧
(authentication (Name v66 says v105 lte v106) ⇔ F) ∧
(authentication (Name v66 says v107 lt v108) ⇔ F) ∧
(authentication (v67 meet v68 says v11) ⇔ F) ∧
(authentication (v69 quoting v70 says v11) ⇔ F) ∧
(authentication (v12 speaks_for v13) ⇔ F) ∧
(authentication (v14 controls v15) ⇔ F) ∧
(authentication (reps v16 v17 v18) ⇔ F) ∧
(authentication (v19 domi v20) ⇔ F) ∧
(authentication (v21 eqi v22) ⇔ F) ∧
(authentication (v23 doms v24) ⇔ F) ∧
(authentication (v25 eqs v26) ⇔ F) ∧
(authentication (v27 eqn v28) ⇔ F) ∧
(authentication (v29 lte v30) ⇔ F) ∧
(authentication (v31 lt v32) ⇔ F)
```

[authentication_ind]

```
⊢ ∀P.
    (∀x.
       P
         (Name PlatoonLeader says
          prop (SOME (PlatoonLeaderCOM x)))) ∧
    (∀x. P (Name Omni says prop (SOME (OmniCOM x)))) ∧ P TT ∧
    P FF ∧ (∀v. P (prop v)) ∧ (∀v1. P (notf v1)) ∧
    (∀v2 v3. P (v2 andf v3)) ∧ (∀v4 v5. P (v4 orf v5)) ∧
    (∀v6 v7. P (v6 impf v7)) ∧ (∀v8 v9. P (v8 eqf v9)) ∧
    (∀v66. P (Name v66 says TT)) ∧
    (∀v66. P (Name v66 says FF)) ∧
    (∀v66. P (Name v66 says prop NONE)) ∧
    (∀v144.
       P
         (Name Omni says
          prop (SOME (PlatoonLeaderCOM v144)))) ∧
    (∀v145.
       P
         (Name PlatoonLeader says
          prop (SOME (OmniCOM v145)))) ∧
    (∀v66 v77. P (Name v66 says notf v77)) ∧
    (∀v66 v78 v79. P (Name v66 says (v78 andf v79))) ∧
    (∀v66 v80 v81. P (Name v66 says (v80 orf v81))) ∧
    (∀v66 v82 v83. P (Name v66 says (v82 impf v83))) ∧
    (∀v66 v84 v85. P (Name v66 says (v84 eqf v85))) ∧
    (∀v66 v86 v87. P (Name v66 says v86 says v87)) ∧
```

```
(∀v66 v88 v89. P (Name v66 says v88 speaks_for v89)) ∧
(∀v66 v90 v91. P (Name v66 says v90 controls v91)) ∧
(∀v66 v92 v93 v94. P (Name v66 says reps v92 v93 v94)) ∧
(∀v66 v95 v96. P (Name v66 says v95 domi v96)) ∧
(∀v66 v97 v98. P (Name v66 says v97 eqi v98)) ∧
(∀v66 v99 v100. P (Name v66 says v99 doms v100)) ∧
(∀v66 v101 v102. P (Name v66 says v101 eqs v102)) ∧
(∀v66 v103 v104. P (Name v66 says v103 eqn v104)) ∧
(∀v66 v105 v106. P (Name v66 says v105 lte v106)) ∧
(∀v66 v107 v108. P (Name v66 says v107 lt v108)) ∧
(∀v67 v68 v11. P (v67 meet v68 says v11)) ∧
(∀v69 v70 v11. P (v69 quoting v70 says v11)) ∧
(∀v12 v13. P (v12 speaks_for v13)) ∧
(∀v14 v15. P (v14 controls v15)) ∧
(∀v16 v17 v18. P (reps v16 v17 v18)) ∧
(∀v19 v20. P (v19 domi v20)) ∧
(∀v21 v22. P (v21 eqi v22)) ∧
(∀v23 v24. P (v23 doms v24)) ∧
(∀v25 v26. P (v25 eqs v26)) ∧ (∀v27 v28. P (v27 eqn v28)) ∧
(∀v29 v30. P (v29 lte v30)) ∧ (∀v31 v32. P (v31 lt v32)) ⇒
∀v. P v
```

5 projectAssuranceExec Theory

Built: 27 December 2018

Parent Theories: projectSecurity

5.1 Theorems

[CONDUCT_ORP_exec_formST_lemma1]

```
⊢ ∀M Oi Os.
    CFGInterpret (M,Oi,Os)
      (CFG authentication stateAuth globalAuth
         ([Name PlatoonLeader says
           prop (SOME (PlatoonLeaderCOM formST))]::ins)
         CONDUCT_ORP outs) ⇒
    (M,Oi,Os) satList
    propCommandList
      [Name PlatoonLeader says
       prop (SOME (PlatoonLeaderCOM formST))]
```

[CONDUCT_ORP_exec_formST_lemma2]

```
⊢ ∀NS Out M Oi Os.
    TR (M,Oi,Os)
      (exec
         (inputList
            [Name PlatoonLeader says
```

```
             prop (SOME (PlatoonLeaderCOM formST))]))
      (CFG authentication stateAuth globalAuth
         ([Name PlatoonLeader says
           prop (SOME (PlatoonLeaderCOM formST))]::ins)
         CONDUCT_ORP outs)
      (CFG authentication stateAuth globalAuth ins
         (NS CONDUCT_ORP
            (exec
               (inputList
                  [Name PlatoonLeader says
                   prop (SOME (PlatoonLeaderCOM formST))])))
         (Out CONDUCT_ORP
            (exec
               (inputList
                  [Name PlatoonLeader says
                   prop (SOME (PlatoonLeaderCOM formST))]))::
              outs)) ⟺
    authenticationTest authentication
      [Name PlatoonLeader says
       prop (SOME (PlatoonLeaderCOM formST))] ∧
    CFGInterpret (M,Oi,Os)
      (CFG authentication stateAuth globalAuth
         ([Name PlatoonLeader says
           prop (SOME (PlatoonLeaderCOM formST))]::ins)
         CONDUCT_ORP outs) ∧
    (M,Oi,Os) satList
    propCommandList
      [Name PlatoonLeader says
       prop (SOME (PlatoonLeaderCOM formST))]
```

[CONDUCT_ORP_exec_formST_thm]

```
⊢ ∀NS Out M Oi Os.
    TR (M,Oi,Os) (exec [SOME (PlatoonLeaderCOM formST)])
      (CFG authentication stateAuth globalAuth
         ([Name PlatoonLeader says
           prop (SOME (PlatoonLeaderCOM formST))]::ins)
         CONDUCT_ORP outs)
      (CFG authentication stateAuth globalAuth ins
         (NS CONDUCT_ORP
            (exec [SOME (PlatoonLeaderCOM formST)]))
         (Out CONDUCT_ORP
            (exec [SOME (PlatoonLeaderCOM formST)])::outs)) ⟺
    authenticationTest authentication
      [Name PlatoonLeader says
       prop (SOME (PlatoonLeaderCOM formST))] ∧
    CFGInterpret (M,Oi,Os)
      (CFG authentication stateAuth globalAuth
         ([Name PlatoonLeader says
           prop (SOME (PlatoonLeaderCOM formST))]::ins)
         CONDUCT_ORP outs) ∧
     (M,Oi,Os) satList [prop (SOME (PlatoonLeaderCOM formST))]
```

[CONTINGENCY_PLAN_exec_moveToORP_lemma1]

```
 ⊢ ∀ M  Oi  Os.
     CFGInterpret (M,Oi,Os)
       (CFG authentication stateAuth globalAuth
          ([Name PlatoonLeader says
            prop (SOME (PlatoonLeaderCOM moveToORP))]::ins)
          CONTINGENCY_PLAN outs) ⇒
     (M,Oi,Os) satList
     propCommandList
       [Name PlatoonLeader says
        prop (SOME (PlatoonLeaderCOM moveToORP))]
```

[CONTINGENCY_PLAN_exec_moveToORP_lemma2]

```
 ⊢ ∀ NS  Out  M  Oi  Os.
     TR (M,Oi,Os)
       (exec
          (inputList
             [Name PlatoonLeader says
              prop (SOME (PlatoonLeaderCOM moveToORP))]))
       (CFG authentication stateAuth globalAuth
          ([Name PlatoonLeader says
            prop (SOME (PlatoonLeaderCOM moveToORP))]::ins)
          CONTINGENCY_PLAN outs)
       (CFG authentication stateAuth globalAuth ins
          (NS CONTINGENCY_PLAN
             (exec
                (inputList
                   [Name PlatoonLeader says
                    prop (SOME (PlatoonLeaderCOM moveToORP))])))
          (Out CONTINGENCY_PLAN
             (exec
                (inputList
                   [Name PlatoonLeader says
                    prop
                      (SOME (PlatoonLeaderCOM moveToORP))]))::
                outs)) ⟺
     authenticationTest authentication
       [Name PlatoonLeader says
        prop (SOME (PlatoonLeaderCOM moveToORP))] ∧
     CFGInterpret (M,Oi,Os)
       (CFG authentication stateAuth globalAuth
          ([Name PlatoonLeader says
            prop (SOME (PlatoonLeaderCOM moveToORP))]::ins)
          CONTINGENCY_PLAN outs) ∧
     (M,Oi,Os) satList
     propCommandList
```

```
        [Name PlatoonLeader says
         prop (SOME (PlatoonLeaderCOM moveToORP))]
```

[CONTINGENCY_PLAN_exec_moveToORP_thm]

```
 ⊢ ∀NS Out M Oi Os.
     TR (M,Oi,Os) (exec [SOME (PlatoonLeaderCOM moveToORP)])
       (CFG authentication stateAuth globalAuth
          ([Name PlatoonLeader says
            prop (SOME (PlatoonLeaderCOM moveToORP))]::ins)
          CONTINGENCY_PLAN outs)
       (CFG authentication stateAuth globalAuth ins
          (NS CONTINGENCY_PLAN
             (exec [SOME (PlatoonLeaderCOM moveToORP)]))
          (Out CONTINGENCY_PLAN
             (exec [SOME (PlatoonLeaderCOM moveToORP)])::
               outs)) ⟺
     authenticationTest authentication
       [Name PlatoonLeader says
        prop (SOME (PlatoonLeaderCOM moveToORP))] ∧
     CFGInterpret (M,Oi,Os)
       (CFG authentication stateAuth globalAuth
          ([Name PlatoonLeader says
            prop (SOME (PlatoonLeaderCOM moveToORP))]::ins)
          CONTINGENCY_PLAN outs) ∧
     (M,Oi,Os) satList
     [prop (SOME (PlatoonLeaderCOM moveToORP))]
```

[FORM_ST_exec_returnToUnit_lemma1]

```
 ⊢ ∀M Oi Os.
     CFGInterpret (M,Oi,Os)
       (CFG authentication stateAuth globalAuth
          ([Name PlatoonLeader says
            prop (SOME (PlatoonLeaderCOM returnToUnit))]::ins)
          FORM_ST outs) ⇒
     (M,Oi,Os) satList
     propCommandList
       [Name PlatoonLeader says
        prop (SOME (PlatoonLeaderCOM returnToUnit))]
```

[FORM_ST_exec_returnToUnit_lemma2]

```
 ⊢ ∀NS Out M Oi Os.
     TR (M,Oi,Os)
       (exec
          (inputList
             [Name PlatoonLeader says
              prop (SOME (PlatoonLeaderCOM returnToUnit))]))
       (CFG authentication stateAuth globalAuth
          ([Name PlatoonLeader says
            prop (SOME (PlatoonLeaderCOM returnToUnit))]::ins)
          FORM_ST outs)
       (CFG authentication stateAuth globalAuth ins
          (NS FORM_ST
             (exec
                (inputList
                   [Name PlatoonLeader says
                    prop
                      (SOME (PlatoonLeaderCOM returnToUnit))])))
          (Out FORM_ST
             (exec
                (inputList
                   [Name PlatoonLeader says
                    prop
                      (SOME
                         (PlatoonLeaderCOM returnToUnit))]))::
               outs)) ⟺
     authenticationTest authentication
       [Name PlatoonLeader says
        prop (SOME (PlatoonLeaderCOM returnToUnit))] ∧
     CFGInterpret (M,Oi,Os)
       (CFG authentication stateAuth globalAuth
          ([Name PlatoonLeader says
            prop (SOME (PlatoonLeaderCOM returnToUnit))]::ins)
          FORM_ST outs) ∧
     (M,Oi,Os) satList
     propCommandList
       [Name PlatoonLeader says
        prop (SOME (PlatoonLeaderCOM returnToUnit))]
```

[FORM_ST_exec_returnToUnit_thm]

```
 ⊢ ∀NS Out M Oi Os.
     TR (M,Oi,Os) (exec [SOME (PlatoonLeaderCOM returnToUnit)])
       (CFG authentication stateAuth globalAuth
          ([Name PlatoonLeader says
            prop (SOME (PlatoonLeaderCOM returnToUnit))]::ins)
          FORM_ST outs)
       (CFG authentication stateAuth globalAuth ins
          (NS FORM_ST
             (exec [SOME (PlatoonLeaderCOM returnToUnit)]))
          (Out FORM_ST
             (exec [SOME (PlatoonLeaderCOM returnToUnit)])::
               outs)) ⟺
     authenticationTest authentication
       [Name PlatoonLeader says
        prop (SOME (PlatoonLeaderCOM returnToUnit))] ∧
     CFGInterpret (M,Oi,Os)
       (CFG authentication stateAuth globalAuth
          ([Name PlatoonLeader says
            prop (SOME (PlatoonLeaderCOM returnToUnit))]::ins)
          FORM_ST outs) ∧
     (M,Oi,Os) satList
     [prop (SOME (PlatoonLeaderCOM returnToUnit))]
```

[MOVE_TO_ORP_exec_conductORP_lemma1]

```
⊢ ∀ M  Oi  Os.
     CFGInterpret (M,Oi,Os)
       (CFG authentication stateAuth globalAuth
          ([Name PlatoonLeader says
            prop (SOME (PlatoonLeaderCOM conductORP))]::ins)
          MOVE_TO_ORP outs) ⇒
     (M,Oi,Os) satList
     propCommandList
       [Name PlatoonLeader says
        prop (SOME (PlatoonLeaderCOM conductORP))]
```

[MOVE_TO_ORP_exec_conductORP_lemma2]

```
⊢ ∀ NS  Out  M  Oi  Os.
     TR (M,Oi,Os)
       (exec
          (inputList
             [Name PlatoonLeader says
              prop (SOME (PlatoonLeaderCOM conductORP))]))
       (CFG authentication stateAuth globalAuth
          ([Name PlatoonLeader says
            prop (SOME (PlatoonLeaderCOM conductORP))]::ins)
          MOVE_TO_ORP outs)
       (CFG authentication stateAuth globalAuth ins
          (NS MOVE_TO_ORP
             (exec
                (inputList
                   [Name PlatoonLeader says
                    prop
                      (SOME (PlatoonLeaderCOM conductORP))])))
          (Out MOVE_TO_ORP
             (exec
                (inputList
                   [Name PlatoonLeader says
                    prop
                      (SOME (PlatoonLeaderCOM conductORP))]))::
           outs)) ⟺
     authenticationTest authentication
       [Name PlatoonLeader says
        prop (SOME (PlatoonLeaderCOM conductORP))] ∧
     CFGInterpret (M,Oi,Os)
       (CFG authentication stateAuth globalAuth
          ([Name PlatoonLeader says
            prop (SOME (PlatoonLeaderCOM conductORP))]::ins)
          MOVE_TO_ORP outs) ∧
     (M,Oi,Os) satList
     propCommandList
       [Name PlatoonLeader says
        prop (SOME (PlatoonLeaderCOM conductORP))]
```

[MOVE_TO_ORP_exec_conductORP_thm]

```
 ⊢ ∀NS Out M Oi Os.
     TR (M,Oi,Os) (exec [SOME (PlatoonLeaderCOM conductORP)])
       (CFG authentication stateAuth globalAuth
          ([Name PlatoonLeader says
            prop (SOME (PlatoonLeaderCOM conductORP))]::ins)
          MOVE_TO_ORP outs)
       (CFG authentication stateAuth globalAuth ins
          (NS MOVE_TO_ORP
             (exec [SOME (PlatoonLeaderCOM conductORP)]))
          (Out MOVE_TO_ORP
             (exec [SOME (PlatoonLeaderCOM conductORP)])::
               outs)) ⟺
     authenticationTest authentication
       [Name PlatoonLeader says
        prop (SOME (PlatoonLeaderCOM conductORP))] ∧
     CFGInterpret (M,Oi,Os)
       (CFG authentication stateAuth globalAuth
          ([Name PlatoonLeader says
            prop (SOME (PlatoonLeaderCOM conductORP))]::ins)
          MOVE_TO_ORP outs) ∧
     (M,Oi,Os) satList
     [prop (SOME (PlatoonLeaderCOM conductORP))]
```

[ORP_RECON_exec_contingencyPlan_lemma1]

```
 ⊢ ∀M Oi Os.
     CFGInterpret (M,Oi,Os)
       (CFG authentication stateAuth globalAuth
          ([Name PlatoonLeader says
            prop (SOME (PlatoonLeaderCOM contingencyPlan))]::
               ins) ORP_RECON outs) ⇒
     (M,Oi,Os) satList
     propCommandList
       [Name PlatoonLeader says
        prop (SOME (PlatoonLeaderCOM contingencyPlan))]
```

[ORP_RECON_exec_contingencyPlan_lemma2]

```
 ⊢ ∀NS Out M Oi Os.
     TR (M,Oi,Os)
       (exec
          (inputList
             [Name PlatoonLeader says
```

```
              prop (SOME (PlatoonLeaderCOM contingencyPlan))]))
       (CFG authentication stateAuth globalAuth
          ([Name PlatoonLeader says
            prop (SOME (PlatoonLeaderCOM contingencyPlan))]::
               ins) ORP_RECON outs)
       (CFG authentication stateAuth globalAuth ins
          (NS ORP_RECON
             (exec
                (inputList
                   [Name PlatoonLeader says
                    prop
                      (SOME
                         (PlatoonLeaderCOM contingencyPlan))])))
          (Out ORP_RECON
             (exec
                (inputList
                   [Name PlatoonLeader says
                    prop
                      (SOME
                         (PlatoonLeaderCOM
                            contingencyPlan))]))::outs)) ⟺
     authenticationTest authentication
       [Name PlatoonLeader says
        prop (SOME (PlatoonLeaderCOM contingencyPlan))] ∧
     CFGInterpret (M,Oi,Os)
       (CFG authentication stateAuth globalAuth
          ([Name PlatoonLeader says
            prop (SOME (PlatoonLeaderCOM contingencyPlan))]::
               ins) ORP_RECON outs) ∧
     (M,Oi,Os) satList
     propCommandList
       [Name PlatoonLeader says
        prop (SOME (PlatoonLeaderCOM contingencyPlan))]
```

[ORP_RECON_exec_contingencyPlan_thm]

```
 ⊢ ∀NS Out M Oi Os.
     TR (M,Oi,Os)
       (exec [SOME (PlatoonLeaderCOM contingencyPlan)])
       (CFG authentication stateAuth globalAuth
          ([Name PlatoonLeader says
            prop (SOME (PlatoonLeaderCOM contingencyPlan))]::
               ins) ORP_RECON outs)
       (CFG authentication stateAuth globalAuth ins
          (NS ORP_RECON
             (exec [SOME (PlatoonLeaderCOM contingencyPlan)]))
          (Out ORP_RECON
             (exec [SOME (PlatoonLeaderCOM contingencyPlan)])::
               outs)) ⟺
     authenticationTest authentication
```

```
        [Name PlatoonLeader says
         prop (SOME (PlatoonLeaderCOM contingencyPlan))] ∧
      CFGInterpret (M,Oi,Os)
        (CFG authentication stateAuth globalAuth
           ([Name PlatoonLeader says
             prop (SOME (PlatoonLeaderCOM contingencyPlan))]::
                ins) ORP_RECON outs) ∧
      (M,Oi,Os) satList
      [prop (SOME (PlatoonLeaderCOM contingencyPlan))]
```

[RETURN_TO_UNIT_exec_complete_lemma1]

```
 ⊢ ∀M Oi Os.
      CFGInterpret (M,Oi,Os)
        (CFG authentication stateAuth globalAuth
           ([Name PlatoonLeader says
             prop (SOME (PlatoonLeaderCOM complete))]::ins)
           RETURN_TO_UNIT outs) ⇒
      (M,Oi,Os) satList
      propCommandList
        [Name PlatoonLeader says
         prop (SOME (PlatoonLeaderCOM complete))]
```

[RETURN_TO_UNIT_exec_complete_lemma2]

```
 ⊢ ∀NS Out M Oi Os.
      TR (M,Oi,Os)
        (exec
           (inputList
              [Name PlatoonLeader says
               prop (SOME (PlatoonLeaderCOM complete))]))
        (CFG authentication stateAuth globalAuth
           ([Name PlatoonLeader says
             prop (SOME (PlatoonLeaderCOM complete))]::ins)
           RETURN_TO_UNIT outs)
        (CFG authentication stateAuth globalAuth ins
           (NS RETURN_TO_UNIT
              (exec
                 (inputList
                    [Name PlatoonLeader says
                     prop (SOME (PlatoonLeaderCOM complete))])))
           (Out RETURN_TO_UNIT
              (exec
                 (inputList
                    [Name PlatoonLeader says
                     prop (SOME (PlatoonLeaderCOM complete))]))::
                outs)) ⟺
      authenticationTest authentication
        [Name PlatoonLeader says
         prop (SOME (PlatoonLeaderCOM complete))] ∧
      CFGInterpret (M,Oi,Os)
```

```
        (CFG authentication stateAuth globalAuth
           ([Name PlatoonLeader says
             prop (SOME (PlatoonLeaderCOM complete))]::ins)
           RETURN_TO_UNIT outs) ∧
      (M,Oi,Os) satList
      propCommandList
        [Name PlatoonLeader says
         prop (SOME (PlatoonLeaderCOM complete))]
```

[RETURN_TO_UNIT_exec_complete_thm]

```
 ⊢ ∀ NS Out M Oi Os.
      TR (M,Oi,Os) (exec [SOME (PlatoonLeaderCOM complete)])
        (CFG authentication stateAuth globalAuth
           ([Name PlatoonLeader says
             prop (SOME (PlatoonLeaderCOM complete))]::ins)
           RETURN_TO_UNIT outs)
        (CFG authentication stateAuth globalAuth ins
           (NS RETURN_TO_UNIT
              (exec [SOME (PlatoonLeaderCOM complete)]))
           (Out RETURN_TO_UNIT
              (exec [SOME (PlatoonLeaderCOM complete)])::
                outs)) ⇔
      authenticationTest authentication
        [Name PlatoonLeader says
         prop (SOME (PlatoonLeaderCOM complete))] ∧
      CFGInterpret (M,Oi,Os)
        (CFG authentication stateAuth globalAuth
           ([Name PlatoonLeader says
             prop (SOME (PlatoonLeaderCOM complete))]::ins)
           RETURN_TO_UNIT outs) ∧
      (M,Oi,Os) satList [prop (SOME (PlatoonLeaderCOM complete))]
```

Index