Contents

1 PlanPBType Theory

Built: 10 June 2018

Parent Theories: indexedLists, patternMatches

1.1 Datatypes

```
plCommand = receiveMission | warno | tentativePlan | recon
          | report1 | completePlan | opoid | supervise | report2
          | complete | plIncomplete | invalidPlCommand

psgCommand = initiateMovement | psgIncomplete
           | invalidPsgCommand

slCommand = PL plCommand | PSG psgCommand

slOutput = PlanPB | ReceiveMission | Warno | TentativePlan
         | InitiateMovement | Recon | Report1 | CompletePlan
         | Opoid | Supervise | Report2 | Complete
         | unAuthenticated | unAuthorized

slState = PLAN_PB | RECEIVE_MISSION | WARNO | TENTATIVE_PLAN
        | INITIATE_MOVEMENT | RECON | REPORT1 | COMPLETE_PLAN
        | OPOID | SUPERVISE | REPORT2 | COMPLETE

stateRole = PlatoonLeader | PlatoonSergeant
```

1.2 Theorems

[plCommand_distinct_clauses]

```
⊢ receiveMission ≠ warno ∧ receiveMission ≠ tentativePlan ∧
  receiveMission ≠ recon ∧ receiveMission ≠ report1 ∧
  receiveMission ≠ completePlan ∧ receiveMission ≠ opoid ∧
  receiveMission ≠ supervise ∧ receiveMission ≠ report2 ∧
  receiveMission ≠ complete ∧ receiveMission ≠ plIncomplete ∧
  receiveMission ≠ invalidPlCommand ∧ warno ≠ tentativePlan ∧
  warno ≠ recon ∧ warno ≠ report1 ∧ warno ≠ completePlan ∧
  warno ≠ opoid ∧ warno ≠ supervise ∧ warno ≠ report2 ∧
  warno ≠ complete ∧ warno ≠ plIncomplete ∧
  warno ≠ invalidPlCommand ∧ tentativePlan ≠ recon ∧
  tentativePlan ≠ report1 ∧ tentativePlan ≠ completePlan ∧
  tentativePlan ≠ opoid ∧ tentativePlan ≠ supervise ∧
  tentativePlan ≠ report2 ∧ tentativePlan ≠ complete ∧
  tentativePlan ≠ plIncomplete ∧
  tentativePlan ≠ invalidPlCommand ∧ recon ≠ report1 ∧
  recon ≠ completePlan ∧ recon ≠ opoid ∧ recon ≠ supervise ∧
  recon ≠ report2 ∧ recon ≠ complete ∧ recon ≠ plIncomplete ∧
  recon ≠ invalidPlCommand ∧ report1 ≠ completePlan ∧
```

```
report1 ≠ opoid ∧ report1 ≠ supervise ∧ report1 ≠ report2 ∧
report1 ≠ complete ∧ report1 ≠ plIncomplete ∧
report1 ≠ invalidPlCommand ∧ completePlan ≠ opoid ∧
completePlan ≠ supervise ∧ completePlan ≠ report2 ∧
completePlan ≠ complete ∧ completePlan ≠ plIncomplete ∧
completePlan ≠ invalidPlCommand ∧ opoid ≠ supervise ∧
opoid ≠ report2 ∧ opoid ≠ complete ∧ opoid ≠ plIncomplete ∧
opoid ≠ invalidPlCommand ∧ supervise ≠ report2 ∧
supervise ≠ complete ∧ supervise ≠ plIncomplete ∧
supervise ≠ invalidPlCommand ∧ report2 ≠ complete ∧
report2 ≠ plIncomplete ∧ report2 ≠ invalidPlCommand ∧
complete ≠ plIncomplete ∧ complete ≠ invalidPlCommand ∧
plIncomplete ≠ invalidPlCommand
```

[psgCommand_distinct_clauses]

```
⊢ initiateMovement ≠ psgIncomplete ∧
  initiateMovement ≠ invalidPsgCommand ∧
  psgIncomplete ≠ invalidPsgCommand
```

[slCommand_distinct_clauses]

```
⊢ ∀a′ a. PL a ≠ PSG a′
```

[slCommand_one_one]

```
⊢ (∀a a′. (PL a = PL a′) ⟺ (a = a′)) ∧
  ∀a a′. (PSG a = PSG a′) ⟺ (a = a′)
```

[slOutput_distinct_clauses]

```
⊢ PlanPB ≠ ReceiveMission ∧ PlanPB ≠ Warno ∧
  PlanPB ≠ TentativePlan ∧ PlanPB ≠ InitiateMovement ∧
  PlanPB ≠ Recon ∧ PlanPB ≠ Report1 ∧ PlanPB ≠ CompletePlan ∧
  PlanPB ≠ Opoid ∧ PlanPB ≠ Supervise ∧ PlanPB ≠ Report2 ∧
  PlanPB ≠ Complete ∧ PlanPB ≠ unAuthenticated ∧
  PlanPB ≠ unAuthorized ∧ ReceiveMission ≠ Warno ∧
  ReceiveMission ≠ TentativePlan ∧
  ReceiveMission ≠ InitiateMovement ∧ ReceiveMission ≠ Recon ∧
  ReceiveMission ≠ Report1 ∧ ReceiveMission ≠ CompletePlan ∧
  ReceiveMission ≠ Opoid ∧ ReceiveMission ≠ Supervise ∧
  ReceiveMission ≠ Report2 ∧ ReceiveMission ≠ Complete ∧
  ReceiveMission ≠ unAuthenticated ∧
  ReceiveMission ≠ unAuthorized ∧ Warno ≠ TentativePlan ∧
  Warno ≠ InitiateMovement ∧ Warno ≠ Recon ∧ Warno ≠ Report1 ∧
  Warno ≠ CompletePlan ∧ Warno ≠ Opoid ∧ Warno ≠ Supervise ∧
  Warno ≠ Report2 ∧ Warno ≠ Complete ∧
  Warno ≠ unAuthenticated ∧ Warno ≠ unAuthorized ∧
  TentativePlan ≠ InitiateMovement ∧ TentativePlan ≠ Recon ∧
  TentativePlan ≠ Report1 ∧ TentativePlan ≠ CompletePlan ∧
  TentativePlan ≠ Opoid ∧ TentativePlan ≠ Supervise ∧
  TentativePlan ≠ Report2 ∧ TentativePlan ≠ Complete ∧
```

```
   TentativePlan ≠ unAuthenticated ∧
   TentativePlan ≠ unAuthorized ∧ InitiateMovement ≠ Recon ∧
   InitiateMovement ≠ Report1 ∧
   InitiateMovement ≠ CompletePlan ∧ InitiateMovement ≠ Opoid ∧
   InitiateMovement ≠ Supervise ∧ InitiateMovement ≠ Report2 ∧
   InitiateMovement ≠ Complete ∧
   InitiateMovement ≠ unAuthenticated ∧
   InitiateMovement ≠ unAuthorized ∧ Recon ≠ Report1 ∧
   Recon ≠ CompletePlan ∧ Recon ≠ Opoid ∧ Recon ≠ Supervise ∧
   Recon ≠ Report2 ∧ Recon ≠ Complete ∧
   Recon ≠ unAuthenticated ∧ Recon ≠ unAuthorized ∧
   Report1 ≠ CompletePlan ∧ Report1 ≠ Opoid ∧
   Report1 ≠ Supervise ∧ Report1 ≠ Report2 ∧
   Report1 ≠ Complete ∧ Report1 ≠ unAuthenticated ∧
   Report1 ≠ unAuthorized ∧ CompletePlan ≠ Opoid ∧
   CompletePlan ≠ Supervise ∧ CompletePlan ≠ Report2 ∧
   CompletePlan ≠ Complete ∧ CompletePlan ≠ unAuthenticated ∧
   CompletePlan ≠ unAuthorized ∧ Opoid ≠ Supervise ∧
   Opoid ≠ Report2 ∧ Opoid ≠ Complete ∧
   Opoid ≠ unAuthenticated ∧ Opoid ≠ unAuthorized ∧
   Supervise ≠ Report2 ∧ Supervise ≠ Complete ∧
   Supervise ≠ unAuthenticated ∧ Supervise ≠ unAuthorized ∧
   Report2 ≠ Complete ∧ Report2 ≠ unAuthenticated ∧
   Report2 ≠ unAuthorized ∧ Complete ≠ unAuthenticated ∧
   Complete ≠ unAuthorized ∧ unAuthenticated ≠ unAuthorized
```

[slRole_distinct_clauses]

```
 ⊢ PlatoonLeader ≠ PlatoonSergeant
```

[slState_distinct_clauses]

```
 ⊢ PLAN_PB ≠ RECEIVE_MISSION ∧ PLAN_PB ≠ WARNO ∧
   PLAN_PB ≠ TENTATIVE_PLAN ∧ PLAN_PB ≠ INITIATE_MOVEMENT ∧
   PLAN_PB ≠ RECON ∧ PLAN_PB ≠ REPORT1 ∧
   PLAN_PB ≠ COMPLETE_PLAN ∧ PLAN_PB ≠ OPOID ∧
   PLAN_PB ≠ SUPERVISE ∧ PLAN_PB ≠ REPORT2 ∧
   PLAN_PB ≠ COMPLETE ∧ RECEIVE_MISSION ≠ WARNO ∧
   RECEIVE_MISSION ≠ TENTATIVE_PLAN ∧
   RECEIVE_MISSION ≠ INITIATE_MOVEMENT ∧
   RECEIVE_MISSION ≠ RECON ∧ RECEIVE_MISSION ≠ REPORT1 ∧
   RECEIVE_MISSION ≠ COMPLETE_PLAN ∧ RECEIVE_MISSION ≠ OPOID ∧
   RECEIVE_MISSION ≠ SUPERVISE ∧ RECEIVE_MISSION ≠ REPORT2 ∧
   RECEIVE_MISSION ≠ COMPLETE ∧ WARNO ≠ TENTATIVE_PLAN ∧
   WARNO ≠ INITIATE_MOVEMENT ∧ WARNO ≠ RECON ∧ WARNO ≠ REPORT1 ∧
   WARNO ≠ COMPLETE_PLAN ∧ WARNO ≠ OPOID ∧ WARNO ≠ SUPERVISE ∧
   WARNO ≠ REPORT2 ∧ WARNO ≠ COMPLETE ∧
   TENTATIVE_PLAN ≠ INITIATE_MOVEMENT ∧ TENTATIVE_PLAN ≠ RECON ∧
   TENTATIVE_PLAN ≠ REPORT1 ∧ TENTATIVE_PLAN ≠ COMPLETE_PLAN ∧
   TENTATIVE_PLAN ≠ OPOID ∧ TENTATIVE_PLAN ≠ SUPERVISE ∧
   TENTATIVE_PLAN ≠ REPORT2 ∧ TENTATIVE_PLAN ≠ COMPLETE ∧
```

```
INITIATE_MOVEMENT ≠ RECON ∧ INITIATE_MOVEMENT ≠ REPORT1 ∧
INITIATE_MOVEMENT ≠ COMPLETE_PLAN ∧
INITIATE_MOVEMENT ≠ OPOID ∧ INITIATE_MOVEMENT ≠ SUPERVISE ∧
INITIATE_MOVEMENT ≠ REPORT2 ∧ INITIATE_MOVEMENT ≠ COMPLETE ∧
RECON ≠ REPORT1 ∧ RECON ≠ COMPLETE_PLAN ∧ RECON ≠ OPOID ∧
RECON ≠ SUPERVISE ∧ RECON ≠ REPORT2 ∧ RECON ≠ COMPLETE ∧
REPORT1 ≠ COMPLETE_PLAN ∧ REPORT1 ≠ OPOID ∧
REPORT1 ≠ SUPERVISE ∧ REPORT1 ≠ REPORT2 ∧
REPORT1 ≠ COMPLETE ∧ COMPLETE_PLAN ≠ OPOID ∧
COMPLETE_PLAN ≠ SUPERVISE ∧ COMPLETE_PLAN ≠ REPORT2 ∧
COMPLETE_PLAN ≠ COMPLETE ∧ OPOID ≠ SUPERVISE ∧
OPOID ≠ REPORT2 ∧ OPOID ≠ COMPLETE ∧ SUPERVISE ≠ REPORT2 ∧
SUPERVISE ≠ COMPLETE ∧ REPORT2 ≠ COMPLETE
```

2 ssmPlanPB Theory

Built: 10 June 2018

Parent Theories: PlanPBDef, ssm

2.1 Theorems

[inputOK_def]

```
⊢ (inputOK (Name PlatoonLeader says prop cmd) ⇔ T) ∧
  (inputOK (Name PlatoonSergeant says prop cmd) ⇔ T) ∧
  (inputOK TT ⇔ F) ∧ (inputOK FF ⇔ F) ∧
  (inputOK (prop v) ⇔ F) ∧ (inputOK (notf v1) ⇔ F) ∧
  (inputOK (v2 andf v3) ⇔ F) ∧ (inputOK (v4 orf v5) ⇔ F) ∧
  (inputOK (v6 impf v7) ⇔ F) ∧ (inputOK (v8 eqf v9) ⇔ F) ∧
  (inputOK (v10 says TT) ⇔ F) ∧ (inputOK (v10 says FF) ⇔ F) ∧
  (inputOK (v133 meet v134 says prop v66) ⇔ F) ∧
  (inputOK (v135 quoting v136 says prop v66) ⇔ F) ∧
  (inputOK (v10 says notf v67) ⇔ F) ∧
  (inputOK (v10 says (v68 andf v69)) ⇔ F) ∧
  (inputOK (v10 says (v70 orf v71)) ⇔ F) ∧
  (inputOK (v10 says (v72 impf v73)) ⇔ F) ∧
  (inputOK (v10 says (v74 eqf v75)) ⇔ F) ∧
  (inputOK (v10 says v76 says v77) ⇔ F) ∧
  (inputOK (v10 says v78 speaks_for v79) ⇔ F) ∧
  (inputOK (v10 says v80 controls v81) ⇔ F) ∧
  (inputOK (v10 says reps v82 v83 v84) ⇔ F) ∧
  (inputOK (v10 says v85 domi v86) ⇔ F) ∧
  (inputOK (v10 says v87 eqi v88) ⇔ F) ∧
  (inputOK (v10 says v89 doms v90) ⇔ F) ∧
  (inputOK (v10 says v91 eqs v92) ⇔ F) ∧
  (inputOK (v10 says v93 eqn v94) ⇔ F) ∧
  (inputOK (v10 says v95 lte v96) ⇔ F) ∧
  (inputOK (v10 says v97 lt v98) ⇔ F) ∧
```

```
(inputOK (v12 speaks_for v13) ⟺ F) ∧
(inputOK (v14 controls v15) ⟺ F) ∧
(inputOK (reps v16 v17 v18) ⟺ F) ∧
(inputOK (v19 domi v20) ⟺ F) ∧
(inputOK (v21 eqi v22) ⟺ F) ∧
(inputOK (v23 doms v24) ⟺ F) ∧
(inputOK (v25 eqs v26) ⟺ F) ∧ (inputOK (v27 eqn v28) ⟺ F) ∧
(inputOK (v29 lte v30) ⟺ F) ∧ (inputOK (v31 lt v32) ⟺ F)
```

[inputOK_ind]

```
⊢ ∀P.
    (∀cmd. P (Name PlatoonLeader says prop cmd)) ∧
    (∀cmd. P (Name PlatoonSergeant says prop cmd)) ∧ 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)) ∧
    (∀v10. P (v10 says TT)) ∧ (∀v10. P (v10 says FF)) ∧
    (∀v133 v134 v66. P (v133 meet v134 says prop v66)) ∧
    (∀v135 v136 v66. P (v135 quoting v136 says prop v66)) ∧
    (∀v10 v67. P (v10 says notf v67)) ∧
    (∀v10 v68 v69. P (v10 says (v68 andf v69))) ∧
    (∀v10 v70 v71. P (v10 says (v70 orf v71))) ∧
    (∀v10 v72 v73. P (v10 says (v72 impf v73))) ∧
    (∀v10 v74 v75. P (v10 says (v74 eqf v75))) ∧
    (∀v10 v76 v77. P (v10 says v76 says v77)) ∧
    (∀v10 v78 v79. P (v10 says v78 speaks_for v79)) ∧
    (∀v10 v80 v81. P (v10 says v80 controls v81)) ∧
    (∀v10 v82 v83 v84. P (v10 says reps v82 v83 v84)) ∧
    (∀v10 v85 v86. P (v10 says v85 domi v86)) ∧
    (∀v10 v87 v88. P (v10 says v87 eqi v88)) ∧
    (∀v10 v89 v90. P (v10 says v89 doms v90)) ∧
    (∀v10 v91 v92. P (v10 says v91 eqs v92)) ∧
    (∀v10 v93 v94. P (v10 says v93 eqn v94)) ∧
    (∀v10 v95 v96. P (v10 says v95 lte v96)) ∧
    (∀v10 v97 v98. P (v10 says v97 lt v98)) ∧
    (∀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
```

[planPBNS_def]

```
⊢ (planPBNS WARNO (exec x) =
   if
     (getRecon x = [SOME (SLc (PL recon))]) ∧
```

```
        (getTenativePlan x = [SOME (SLc (PL tentativePlan))]) ∧
        (getReport x = [SOME (SLc (PL report1))]) ∧
        (getInitMove x = [SOME (SLc (PSG initiateMovement))])
     then
        REPORT1
     else WARNO) ∧
    (planPBNS PLAN_PB (exec x) =
     if getPlCom x = receiveMission then RECEIVE_MISSION
     else PLAN_PB) ∧
    (planPBNS RECEIVE_MISSION (exec x) =
     if getPlCom x = warno then WARNO else RECEIVE_MISSION) ∧
    (planPBNS REPORT1 (exec x) =
     if getPlCom x = completePlan then COMPLETE_PLAN
     else REPORT1) ∧
    (planPBNS COMPLETE_PLAN (exec x) =
     if getPlCom x = opoid then OPOID else COMPLETE_PLAN) ∧
    (planPBNS OPOID (exec x) =
     if getPlCom x = supervise then SUPERVISE else OPOID) ∧
    (planPBNS SUPERVISE (exec x) =
     if getPlCom x = report2 then REPORT2 else SUPERVISE) ∧
    (planPBNS REPORT2 (exec x) =
     if getPlCom x = complete then COMPLETE else REPORT2) ∧
    (planPBNS s (trap v0) = s) ∧ (planPBNS s (discard v1) = s)
```

[planPBNS_ind]

```
 ⊢ ∀P.
     (∀x. P WARNO (exec x)) ∧ (∀x. P PLAN_PB (exec x)) ∧
     (∀x. P RECEIVE_MISSION (exec x)) ∧
     (∀x. P REPORT1 (exec x)) ∧ (∀x. P COMPLETE_PLAN (exec x)) ∧
     (∀x. P OPOID (exec x)) ∧ (∀x. P SUPERVISE (exec x)) ∧
     (∀x. P REPORT2 (exec x)) ∧ (∀s v0. P s (trap v0)) ∧
     (∀s v1. P s (discard v1)) ∧
     (∀v6. P TENTATIVE_PLAN (exec v6)) ∧
     (∀v7. P INITIATE_MOVEMENT (exec v7)) ∧
     (∀v8. P RECON (exec v8)) ∧ (∀v9. P COMPLETE (exec v9)) ⇒
     ∀v v1. P v v1
```

[planPBOut_def]

```
 ⊢ (planPBOut WARNO (exec x) =
     if
        (getRecon x = [SOME (SLc (PL recon))]) ∧
        (getTenativePlan x = [SOME (SLc (PL tentativePlan))]) ∧
        (getReport x = [SOME (SLc (PL report1))]) ∧
        (getInitMove x = [SOME (SLc (PSG initiateMovement))])
     then
        Report1
     else unAuthorized) ∧
    (planPBOut PLAN_PB (exec x) =
     if getPlCom x = receiveMission then ReceiveMission
     else unAuthorized) ∧
    (planPBOut RECEIVE_MISSION (exec x) =
     if getPlCom x = warno then Warno else unAuthorized) ∧
    (planPBOut REPORT1 (exec x) =
     if getPlCom x = completePlan then CompletePlan
     else unAuthorized) ∧
    (planPBOut COMPLETE_PLAN (exec x) =
     if getPlCom x = opoid then Opoid else unAuthorized) ∧
    (planPBOut OPOID (exec x) =
     if getPlCom x = supervise then Supervise
     else unAuthorized) ∧
    (planPBOut SUPERVISE (exec x) =
     if getPlCom x = report2 then Report2 else unAuthorized) ∧
    (planPBOut REPORT2 (exec x) =
     if getPlCom x = complete then Complete else unAuthorized) ∧
    (planPBOut s (trap v0) = unAuthorized) ∧
    (planPBOut s (discard v1) = unAuthenticated)
```

[planPBOut_ind]

```
 ⊢ ∀P.
      (∀x. P WARNO (exec x)) ∧ (∀x. P PLAN_PB (exec x)) ∧
      (∀x. P RECEIVE_MISSION (exec x)) ∧
      (∀x. P REPORT1 (exec x)) ∧ (∀x. P COMPLETE_PLAN (exec x)) ∧
      (∀x. P OPOID (exec x)) ∧ (∀x. P SUPERVISE (exec x)) ∧
      (∀x. P REPORT2 (exec x)) ∧ (∀s v0. P s (trap v0)) ∧
      (∀s v1. P s (discard v1)) ∧
      (∀v6. P TENTATIVE_PLAN (exec v6)) ∧
      (∀v7. P INITIATE_MOVEMENT (exec v7)) ∧
      (∀v8. P RECON (exec v8)) ∧ (∀v9. P COMPLETE (exec v9)) ⇒
      ∀v v1. P v v1
```

[PlatoonLeader_notWARNO_notreport1_exec_plCommand_justified_lemma]

```
 ⊢ s ≠ WARNO ⇒
   plCommand ≠ invalidPlCommand ⇒
   plCommand ≠ report1 ⇒
   ∀NS Out M Oi Os.
     TR (M,Oi,Os)
       (exec
          (inputList
             [Name PlatoonLeader says
              prop (SOME (SLc (PL plCommand)))]))
       (CFG inputOK secContext secContextNull
          ([Name PlatoonLeader says
            prop (SOME (SLc (PL plCommand)))]::ins) s outs)
       (CFG inputOK secContext secContextNull ins
          (NS s
             (exec
                (inputList
                   [Name PlatoonLeader says
```

```
                     prop (SOME (SLc (PL plCommand)))])))
           (Out s
              (exec
                 (inputList
                    [Name PlatoonLeader says
                     prop (SOME (SLc (PL plCommand)))])))::
                outs)) ⟺
      authenticationTest inputOK
        [Name PlatoonLeader says
         prop (SOME (SLc (PL plCommand)))] ∧
      CFGInterpret (M,Oi,Os)
        (CFG inputOK secContext secContextNull
           ([Name PlatoonLeader says
             prop (SOME (SLc (PL plCommand)))]::ins) s outs) ∧
      (M,Oi,Os) satList
      propCommandList
        [Name PlatoonLeader says
         prop (SOME (SLc (PL plCommand)))]
```

[PlatoonLeader_notWARNO_notreport1_exec_plCommand_justified_thm]

```
 ⊢ s ≠ WARNO ⇒
   plCommand ≠ invalidPlCommand ⇒
   plCommand ≠ report1 ⇒
   ∀NS Out M Oi Os.
     TR (M,Oi,Os) (exec [SOME (SLc (PL plCommand))])
       (CFG inputOK secContext secContextNull
          ([Name PlatoonLeader says
            prop (SOME (SLc (PL plCommand)))]::ins) s outs)
       (CFG inputOK secContext secContextNull ins
          (NS s (exec [SOME (SLc (PL plCommand))]))
          (Out s (exec [SOME (SLc (PL plCommand))])::outs)) ⟺
     authenticationTest inputOK
       [Name PlatoonLeader says
        prop (SOME (SLc (PL plCommand)))] ∧
     CFGInterpret (M,Oi,Os)
       (CFG inputOK secContext secContextNull
          ([Name PlatoonLeader says
            prop (SOME (SLc (PL plCommand)))]::ins) s outs) ∧
     (M,Oi,Os) satList [prop (SOME (SLc (PL plCommand)))]
```

[PlatoonLeader_notWARNO_notreport1_exec_plCommand_lemma]

```
 ⊢ s ≠ WARNO ⇒
   plCommand ≠ invalidPlCommand ⇒
   plCommand ≠ report1 ⇒
   ∀M Oi Os.
     CFGInterpret (M,Oi,Os)
       (CFG inputOK secContext secContextNull
          ([Name PlatoonLeader says
            prop (SOME (SLc (PL plCommand)))]::ins) s outs) ⇒
```

```
      (M,Oi,Os) satList
      propCommandList
        [Name PlatoonLeader says
         prop (SOME (SLc (PL plCommand)))]
```

[PlatoonLeader_psgCommand_notDiscard_thm]

```
 ⊢ ∀NS Out M Oi Os.
      ¬TR (M,Oi,Os) (discard [SOME (SLc (PSG psgCommand))])
          (CFG inputOK secContext secContextNull
             ([Name PlatoonLeader says
               prop (SOME (SLc (PSG psgCommand)))]::ins) s outs)
          (CFG inputOK secContext secContextNull ins
             (NS s (discard [SOME (SLc (PSG psgCommand))]))
             (Out s (discard [SOME (SLc (PSG psgCommand))])::
                  outs))
```

[PlatoonLeader_trap_psgCommand_justified_lemma]

```
 ⊢ ∀NS Out M Oi Os.
      TR (M,Oi,Os)
        (trap
           (inputList
              [Name PlatoonLeader says
               prop (SOME (SLc (PSG psgCommand)))]))
        (CFG inputOK secContext secContextNull
           ([Name PlatoonLeader says
             prop (SOME (SLc (PSG psgCommand)))]::ins) s outs)
        (CFG inputOK secContext secContextNull ins
           (NS s
              (trap
                 (inputList
                    [Name PlatoonLeader says
                     prop (SOME (SLc (PSG psgCommand)))])))
           (Out s
              (trap
                 (inputList
                    [Name PlatoonLeader says
                     prop (SOME (SLc (PSG psgCommand)))]))::
                outs)) ⟺
      authenticationTest inputOK
        [Name PlatoonLeader says
         prop (SOME (SLc (PSG psgCommand)))] ∧
      CFGInterpret (M,Oi,Os)
        (CFG inputOK secContext secContextNull
           ([Name PlatoonLeader says
             prop (SOME (SLc (PSG psgCommand)))]::ins) s outs) ∧
      (M,Oi,Os) sat prop NONE
```

[PlatoonLeader_trap_psgCommand_lemma]

```
⊢ ∀ M  Oi  Os .
     CFGInterpret (M ,Oi ,Os)
       (CFG inputOK secContext secContextNull
          ([Name PlatoonLeader says
            prop (SOME (SLc (PSG psgCommand)))]::ins)  s  outs) ⇒
     (M ,Oi ,Os) sat prop NONE
```

[PlatoonLeader_WARNO_exec_report1_justified_lemma]

```
⊢ ∀ NS  Out  M  Oi  Os .
     TR (M ,Oi ,Os)
       (exec
          (inputList
             [Name PlatoonLeader says
              prop (SOME (SLc (PL recon)));
              Name PlatoonLeader says
              prop (SOME (SLc (PL tentativePlan)));
              Name PlatoonSergeant says
              prop (SOME (SLc (PSG initiateMovement)));
              Name PlatoonLeader says
              prop (SOME (SLc (PL report1)))]))
       (CFG inputOK secContext secContextNull
          ([Name PlatoonLeader says
            prop (SOME (SLc (PL recon)));
            Name PlatoonLeader says
            prop (SOME (SLc (PL tentativePlan)));
            Name PlatoonSergeant says
            prop (SOME (SLc (PSG initiateMovement)));
            Name PlatoonLeader says
            prop (SOME (SLc (PL report1)))]::ins) WARNO  outs)
       (CFG inputOK secContext secContextNull ins
          (NS WARNO
             (exec
                (inputList
                   [Name PlatoonLeader says
                    prop (SOME (SLc (PL recon)));
                    Name PlatoonLeader says
                    prop (SOME (SLc (PL tentativePlan)));
                    Name PlatoonSergeant says
                    prop (SOME (SLc (PSG initiateMovement)));
                    Name PlatoonLeader says
                    prop (SOME (SLc (PL report1)))])))
          (Out WARNO
             (exec
                (inputList
                   [Name PlatoonLeader says
                    prop (SOME (SLc (PL recon)));
                    Name PlatoonLeader says
                    prop (SOME (SLc (PL tentativePlan)));
                    Name PlatoonSergeant says
                   prop (SOME (SLc (PSG initiateMovement)));
                   Name PlatoonLeader says
                   prop (SOME (SLc (PL report1)))]])::outs)) ⟺
      authenticationTest inputOK
        [Name PlatoonLeader says prop (SOME (SLc (PL recon)));
         Name PlatoonLeader says
         prop (SOME (SLc (PL tentativePlan)));
         Name PlatoonSergeant says
         prop (SOME (SLc (PSG initiateMovement)));
         Name PlatoonLeader says
         prop (SOME (SLc (PL report1)))] ∧
      CFGInterpret (M,Oi,Os)
        (CFG inputOK secContext secContextNull
           ([Name PlatoonLeader says
             prop (SOME (SLc (PL recon)));
             Name PlatoonLeader says
             prop (SOME (SLc (PL tentativePlan)));
             Name PlatoonSergeant says
             prop (SOME (SLc (PSG initiateMovement)));
             Name PlatoonLeader says
             prop (SOME (SLc (PL report1)))]::ins) WARNO outs) ∧
      (M,Oi,Os) satList
      propCommandList
        [Name PlatoonLeader says prop (SOME (SLc (PL recon)));
         Name PlatoonLeader says
         prop (SOME (SLc (PL tentativePlan)));
         Name PlatoonSergeant says
         prop (SOME (SLc (PSG initiateMovement)));
         Name PlatoonLeader says prop (SOME (SLc (PL report1)))]
```

[PlatoonLeader_WARNO_exec_report1_justified_thm]

```
 ⊢ ∀NS Out M Oi Os.
      TR (M,Oi,Os)
        (exec
           [SOME (SLc (PL recon)); SOME (SLc (PL tentativePlan));
            SOME (SLc (PSG initiateMovement));
            SOME (SLc (PL report1))])
        (CFG inputOK secContext secContextNull
           ([Name PlatoonLeader says
             prop (SOME (SLc (PL recon)));
             Name PlatoonLeader says
             prop (SOME (SLc (PL tentativePlan)));
             Name PlatoonSergeant says
             prop (SOME (SLc (PSG initiateMovement)));
             Name PlatoonLeader says
             prop (SOME (SLc (PL report1)))]::ins) WARNO outs)
        (CFG inputOK secContext secContextNull ins
           (NS WARNO
              (exec
                 [SOME (SLc (PL recon));
                  SOME (SLc (PL tentativePlan));
                  SOME (SLc (PSG initiateMovement));
                  SOME (SLc (PL report1))]))
           (Out WARNO
              (exec
                 [SOME (SLc (PL recon));
                  SOME (SLc (PL tentativePlan));
                  SOME (SLc (PSG initiateMovement));
                  SOME (SLc (PL report1))])::outs)) ⟺
      authenticationTest inputOK
        [Name PlatoonLeader says prop (SOME (SLc (PL recon)));
         Name PlatoonLeader says
         prop (SOME (SLc (PL tentativePlan)));
         Name PlatoonSergeant says
         prop (SOME (SLc (PSG initiateMovement)));
         Name PlatoonLeader says
         prop (SOME (SLc (PL report1)))] ∧
      CFGInterpret (M,Oi,Os)
        (CFG inputOK secContext secContextNull
           ([Name PlatoonLeader says
             prop (SOME (SLc (PL recon)));
             Name PlatoonLeader says
             prop (SOME (SLc (PL tentativePlan)));
             Name PlatoonSergeant says
             prop (SOME (SLc (PSG initiateMovement)));
             Name PlatoonLeader says
             prop (SOME (SLc (PL report1)))]::ins) WARNO outs) ∧
      (M,Oi,Os) satList
      [prop (SOME (SLc (PL recon)));
       prop (SOME (SLc (PL tentativePlan)));
       prop (SOME (SLc (PSG initiateMovement)));
       prop (SOME (SLc (PL report1)))]
```

[PlatoonLeader_WARNO_exec_report1_lemma]

```
 ⊢ ∀M Oi Os.
      CFGInterpret (M,Oi,Os)
        (CFG inputOK secContext secContextNull
           ([Name PlatoonLeader says
             prop (SOME (SLc (PL recon)));
             Name PlatoonLeader says
             prop (SOME (SLc (PL tentativePlan)));
             Name PlatoonSergeant says
             prop (SOME (SLc (PSG initiateMovement)));
             Name PlatoonLeader says
             prop (SOME (SLc (PL report1)))]::ins) WARNO outs) ⇒
      (M,Oi,Os) satList
      propCommandList
        [Name PlatoonLeader says prop (SOME (SLc (PL recon)));
        Name PlatoonLeader says
        prop (SOME (SLc (PL tentativePlan)));
        Name PlatoonSergeant says
        prop (SOME (SLc (PSG initiateMovement)));
        Name PlatoonLeader says prop (SOME (SLc (PL report1)))]
```

[PlatoonSergeant_trap_plCommand_justified_lemma]

```
 ⊢ ∀NS Out M Oi Os.
     TR (M,Oi,Os)
       (trap
          (inputList
             [Name PlatoonSergeant says
              prop (SOME (SLc (PL plCommand)))]))
       (CFG inputOK secContext secContextNull
          ([Name PlatoonSergeant says
            prop (SOME (SLc (PL plCommand)))]::ins) s outs)
       (CFG inputOK secContext secContextNull ins
          (NS s
             (trap
                (inputList
                   [Name PlatoonSergeant says
                    prop (SOME (SLc (PL plCommand)))])))
          (Out s
             (trap
                (inputList
                   [Name PlatoonSergeant says
                    prop (SOME (SLc (PL plCommand)))]))::
               outs)) ⟺
     authenticationTest inputOK
       [Name PlatoonSergeant says
        prop (SOME (SLc (PL plCommand)))] ∧
     CFGInterpret (M,Oi,Os)
       (CFG inputOK secContext secContextNull
          ([Name PlatoonSergeant says
            prop (SOME (SLc (PL plCommand)))]::ins) s outs) ∧
     (M,Oi,Os) sat prop NONE
```

[PlatoonSergeant_trap_plCommand_justified_thm]

```
 ⊢ ∀NS Out M Oi Os.
     TR (M,Oi,Os) (trap [SOME (SLc (PL plCommand))])
       (CFG inputOK secContext secContextNull
          ([Name PlatoonSergeant says
            prop (SOME (SLc (PL plCommand)))]::ins) s outs)
       (CFG inputOK secContext secContextNull ins
          (NS s (trap [SOME (SLc (PL plCommand))]))
          (Out s (trap [SOME (SLc (PL plCommand))])::outs)) ⟺
     authenticationTest inputOK
       [Name PlatoonSergeant says
        prop (SOME (SLc (PL plCommand)))] ∧
     CFGInterpret (M,Oi,Os)
       (CFG inputOK secContext secContextNull
          ([Name PlatoonSergeant says
            prop (SOME (SLc (PL plCommand)))]::ins) s outs) ∧
     (M,Oi,Os) sat prop NONE
```

[PlatoonSergeant_trap_plCommand_lemma]

```
 ⊢ ∀M Oi Os.
     CFGInterpret (M,Oi,Os)
       (CFG inputOK secContext secContextNull
          ([Name PlatoonSergeant says
            prop (SOME (SLc (PL plCommand)))]::ins) s outs) ⇒
     (M,Oi,Os) sat prop NONE
```

3 PlanPBDef Theory

Built: 10 June 2018

Parent Theories: PlanPBType, aclfoundation, OMNIType

3.1 Definitions

[PL_notWARNO_Auth_def]

```
 ⊢ ∀cmd.
     PL_notWARNO_Auth cmd =
     if cmd = report1 then prop NONE
     else
       Name PlatoonLeader says prop (SOME (SLc (PL cmd))) impf
       Name PlatoonLeader controls prop (SOME (SLc (PL cmd)))
```

[PL_WARNO_Auth_def]

```
 ⊢ PL_WARNO_Auth =
   prop (SOME (SLc (PL recon))) impf
   prop (SOME (SLc (PL tentativePlan))) impf
   prop (SOME (SLc (PSG initiateMovement))) impf
   Name PlatoonLeader controls prop (SOME (SLc (PL report1)))
```

[secContext_def]

```
 ⊢ ∀s x.
     secContext s x =
     if s = WARNO then
       if
         (getRecon x = [SOME (SLc (PL recon))]) ∧
         (getTenativePlan x = [SOME (SLc (PL tentativePlan))]) ∧
         (getReport x = [SOME (SLc (PL report1))]) ∧
         (getInitMove x = [SOME (SLc (PSG initiateMovement))])
       then
         [PL_WARNO_Auth;
```

```
          Name PlatoonLeader controls
          prop (SOME (SLc (PL recon)));
          Name PlatoonLeader controls
          prop (SOME (SLc (PL tentativePlan)));
          Name PlatoonSergeant controls
          prop (SOME (SLc (PSG initiateMovement)))]
       else [prop NONE]
     else if getPlCom x = invalidPlCommand then [prop NONE]
     else [PL_notWARNO_Auth (getPlCom x)]
```

[secContextNull_def]

```
 ⊢ ∀x. secContextNull x = [TT]
```

3.2 Theorems

[getInitMove_def]

```
 ⊢ (getInitMove [] = [NONE]) ∧
   (∀xs.
      getInitMove
        (Name PlatoonSergeant says
         prop (SOME (SLc (PSG initiateMovement)))::xs) =
      [SOME (SLc (PSG initiateMovement))]) ∧
   (∀xs. getInitMove (TT::xs) = getInitMove xs) ∧
   (∀xs. getInitMove (FF::xs) = getInitMove xs) ∧
   (∀xs v2. getInitMove (prop v2::xs) = getInitMove xs) ∧
   (∀xs v3. getInitMove (notf v3::xs) = getInitMove xs) ∧
   (∀xs v5 v4. getInitMove (v4 andf v5::xs) = getInitMove xs) ∧
   (∀xs v7 v6. getInitMove (v6 orf v7::xs) = getInitMove xs) ∧
   (∀xs v9 v8. getInitMove (v8 impf v9::xs) = getInitMove xs) ∧
   (∀xs v11 v10.
      getInitMove (v10 eqf v11::xs) = getInitMove xs) ∧
   (∀xs v12. getInitMove (v12 says TT::xs) = getInitMove xs) ∧
   (∀xs v12. getInitMove (v12 says FF::xs) = getInitMove xs) ∧
   (∀xs v134.
      getInitMove (Name v134 says prop NONE::xs) =
      getInitMove xs) ∧
   (∀xs v144.
      getInitMove
        (Name PlatoonLeader says prop (SOME v144)::xs) =
      getInitMove xs) ∧
   (∀xs v146.
      getInitMove
        (Name PlatoonSergeant says prop (SOME (ESCc v146))::
             xs) =
      getInitMove xs) ∧
   (∀xs v150.
      getInitMove
        (Name PlatoonSergeant says prop (SOME (SLc (PL v150)))::
             xs) =
       getInitMove xs) ∧
    (∀ xs.
       getInitMove
         (Name PlatoonSergeant says
          prop (SOME (SLc (PSG psgIncomplete)))::xs) =
       getInitMove xs) ∧
    (∀ xs.
       getInitMove
         (Name PlatoonSergeant says
          prop (SOME (SLc (PSG invalidPsgCommand)))::xs) =
       getInitMove xs) ∧
    (∀ xs v68 v136 v135.
       getInitMove (v135 meet v136 says prop v68::xs) =
       getInitMove xs) ∧
    (∀ xs v68 v138 v137.
       getInitMove (v137 quoting v138 says prop v68::xs) =
       getInitMove xs) ∧
    (∀ xs v69 v12.
       getInitMove (v12 says notf v69::xs) = getInitMove xs) ∧
    (∀ xs v71 v70 v12.
       getInitMove (v12 says (v70 andf v71)::xs) =
       getInitMove xs) ∧
    (∀ xs v73 v72 v12.
       getInitMove (v12 says (v72 orf v73)::xs) =
       getInitMove xs) ∧
    (∀ xs v75 v74 v12.
       getInitMove (v12 says (v74 impf v75)::xs) =
       getInitMove xs) ∧
    (∀ xs v77 v76 v12.
       getInitMove (v12 says (v76 eqf v77)::xs) =
       getInitMove xs) ∧
    (∀ xs v79 v78 v12.
       getInitMove (v12 says v78 says v79::xs) =
       getInitMove xs) ∧
    (∀ xs v81 v80 v12.
       getInitMove (v12 says v80 speaks_for v81::xs) =
       getInitMove xs) ∧
    (∀ xs v83 v82 v12.
       getInitMove (v12 says v82 controls v83::xs) =
       getInitMove xs) ∧
    (∀ xs v86 v85 v84 v12.
       getInitMove (v12 says reps v84 v85 v86::xs) =
       getInitMove xs) ∧
    (∀ xs v88 v87 v12.
       getInitMove (v12 says v87 domi v88::xs) =
       getInitMove xs) ∧
    (∀ xs v90 v89 v12.
       getInitMove (v12 says v89 eqi v90::xs) = getInitMove xs) ∧
```

```
  ($\forall xs$ $v_{92}$ $v_{91}$ $v_{12}$.
     getInitMove ($v_{12}$ says $v_{91}$ doms $v_{92}$::$xs$) =
     getInitMove $xs$) $\wedge$
  ($\forall xs$ $v_{94}$ $v_{93}$ $v_{12}$.
     getInitMove ($v_{12}$ says $v_{93}$ eqs $v_{94}$::$xs$) = getInitMove $xs$) $\wedge$
  ($\forall xs$ $v_{96}$ $v_{95}$ $v_{12}$.
     getInitMove ($v_{12}$ says $v_{95}$ eqn $v_{96}$::$xs$) = getInitMove $xs$) $\wedge$
  ($\forall xs$ $v_{98}$ $v_{97}$ $v_{12}$.
     getInitMove ($v_{12}$ says $v_{97}$ lte $v_{98}$::$xs$) = getInitMove $xs$) $\wedge$
  ($\forall xs$ $v_{99}$ $v_{12}$ $v100$.
     getInitMove ($v_{12}$ says $v_{99}$ lt $v100$::$xs$) = getInitMove $xs$) $\wedge$
  ($\forall xs$ $v_{15}$ $v_{14}$.
     getInitMove ($v_{14}$ speaks_for $v_{15}$::$xs$) = getInitMove $xs$) $\wedge$
  ($\forall xs$ $v_{17}$ $v_{16}$.
     getInitMove ($v_{16}$ controls $v_{17}$::$xs$) = getInitMove $xs$) $\wedge$
  ($\forall xs$ $v_{20}$ $v_{19}$ $v_{18}$.
     getInitMove (reps $v_{18}$ $v_{19}$ $v_{20}$::$xs$) = getInitMove $xs$) $\wedge$
  ($\forall xs$ $v_{22}$ $v_{21}$.
     getInitMove ($v_{21}$ domi $v_{22}$::$xs$) = getInitMove $xs$) $\wedge$
  ($\forall xs$ $v_{24}$ $v_{23}$.
     getInitMove ($v_{23}$ eqi $v_{24}$::$xs$) = getInitMove $xs$) $\wedge$
  ($\forall xs$ $v_{26}$ $v_{25}$.
     getInitMove ($v_{25}$ doms $v_{26}$::$xs$) = getInitMove $xs$) $\wedge$
  ($\forall xs$ $v_{28}$ $v_{27}$.
     getInitMove ($v_{27}$ eqs $v_{28}$::$xs$) = getInitMove $xs$) $\wedge$
  ($\forall xs$ $v_{30}$ $v_{29}$.
     getInitMove ($v_{29}$ eqn $v_{30}$::$xs$) = getInitMove $xs$) $\wedge$
  ($\forall xs$ $v_{32}$ $v_{31}$.
     getInitMove ($v_{31}$ lte $v_{32}$::$xs$) = getInitMove $xs$) $\wedge$
  $\forall xs$ $v_{34}$ $v_{33}$. getInitMove ($v_{33}$ lt $v_{34}$::$xs$) = getInitMove $xs$
```

[getInitMove_ind]

```
 $\vdash$ $\forall P$.
      $P$ [] $\wedge$
      ($\forall xs$.
         $P$
           (Name PlatoonSergeant says
            prop (SOME (SLc (PSG initiateMovement)))::$xs$)) $\wedge$
      ($\forall xs$. $P$ $xs$ $\Rightarrow$ $P$ (TT::$xs$)) $\wedge$ ($\forall xs$. $P$ $xs$ $\Rightarrow$ $P$ (FF::$xs$)) $\wedge$
      ($\forall v_2$ $xs$. $P$ $xs$ $\Rightarrow$ $P$ (prop $v_2$::$xs$)) $\wedge$
      ($\forall v_3$ $xs$. $P$ $xs$ $\Rightarrow$ $P$ (notf $v_3$::$xs$)) $\wedge$
      ($\forall v_4$ $v_5$ $xs$. $P$ $xs$ $\Rightarrow$ $P$ ($v_4$ andf $v_5$::$xs$)) $\wedge$
      ($\forall v_6$ $v_7$ $xs$. $P$ $xs$ $\Rightarrow$ $P$ ($v_6$ orf $v_7$::$xs$)) $\wedge$
      ($\forall v_8$ $v_9$ $xs$. $P$ $xs$ $\Rightarrow$ $P$ ($v_8$ impf $v_9$::$xs$)) $\wedge$
      ($\forall v_{10}$ $v_{11}$ $xs$. $P$ $xs$ $\Rightarrow$ $P$ ($v_{10}$ eqf $v_{11}$::$xs$)) $\wedge$
      ($\forall v_{12}$ $xs$. $P$ $xs$ $\Rightarrow$ $P$ ($v_{12}$ says TT::$xs$)) $\wedge$
      ($\forall v_{12}$ $xs$. $P$ $xs$ $\Rightarrow$ $P$ ($v_{12}$ says FF::$xs$)) $\wedge$
      ($\forall v134$ $xs$. $P$ $xs$ $\Rightarrow$ $P$ (Name $v134$ says prop NONE::$xs$)) $\wedge$
      ($\forall v144$ $xs$.
     P xs ⇒
     P (Name PlatoonLeader says prop (SOME v144)::xs)) ∧
  (∀ v146 xs.
     P xs ⇒
     P
       (Name PlatoonSergeant says prop (SOME (ESCc v146))::
            xs)) ∧
  (∀ v150 xs.
     P xs ⇒
     P
       (Name PlatoonSergeant says
        prop (SOME (SLc (PL v150)))::xs)) ∧
  (∀ xs.
     P xs ⇒
     P
       (Name PlatoonSergeant says
        prop (SOME (SLc (PSG psgIncomplete)))::xs)) ∧
  (∀ xs.
     P xs ⇒
     P
       (Name PlatoonSergeant says
        prop (SOME (SLc (PSG invalidPsgCommand)))::xs)) ∧
  (∀ v135 v136 $v_{68}$ xs.
     P xs ⇒ P (v135 meet v136 says prop $v_{68}$::xs)) ∧
  (∀ v137 v138 $v_{68}$ xs.
     P xs ⇒ P (v137 quoting v138 says prop $v_{68}$::xs)) ∧
  (∀ $v_{12}$ $v_{69}$ xs. P xs ⇒ P ($v_{12}$ says notf $v_{69}$::xs)) ∧
  (∀ $v_{12}$ $v_{70}$ $v_{71}$ xs. P xs ⇒ P ($v_{12}$ says ($v_{70}$ andf $v_{71}$)::xs)) ∧
  (∀ $v_{12}$ $v_{72}$ $v_{73}$ xs. P xs ⇒ P ($v_{12}$ says ($v_{72}$ orf $v_{73}$)::xs)) ∧
  (∀ $v_{12}$ $v_{74}$ $v_{75}$ xs. P xs ⇒ P ($v_{12}$ says ($v_{74}$ impf $v_{75}$)::xs)) ∧
  (∀ $v_{12}$ $v_{76}$ $v_{77}$ xs. P xs ⇒ P ($v_{12}$ says ($v_{76}$ eqf $v_{77}$)::xs)) ∧
  (∀ $v_{12}$ $v_{78}$ $v_{79}$ xs. P xs ⇒ P ($v_{12}$ says $v_{78}$ says $v_{79}$::xs)) ∧
  (∀ $v_{12}$ $v_{80}$ $v_{81}$ xs.
     P xs ⇒ P ($v_{12}$ says $v_{80}$ speaks_for $v_{81}$::xs)) ∧
  (∀ $v_{12}$ $v_{82}$ $v_{83}$ xs.
     P xs ⇒ P ($v_{12}$ says $v_{82}$ controls $v_{83}$::xs)) ∧
  (∀ $v_{12}$ $v_{84}$ $v_{85}$ $v_{86}$ xs.
     P xs ⇒ P ($v_{12}$ says reps $v_{84}$ $v_{85}$ $v_{86}$::xs)) ∧
  (∀ $v_{12}$ $v_{87}$ $v_{88}$ xs. P xs ⇒ P ($v_{12}$ says $v_{87}$ domi $v_{88}$::xs)) ∧
  (∀ $v_{12}$ $v_{89}$ $v_{90}$ xs. P xs ⇒ P ($v_{12}$ says $v_{89}$ eqi $v_{90}$::xs)) ∧
  (∀ $v_{12}$ $v_{91}$ $v_{92}$ xs. P xs ⇒ P ($v_{12}$ says $v_{91}$ doms $v_{92}$::xs)) ∧
  (∀ $v_{12}$ $v_{93}$ $v_{94}$ xs. P xs ⇒ P ($v_{12}$ says $v_{93}$ eqs $v_{94}$::xs)) ∧
  (∀ $v_{12}$ $v_{95}$ $v_{96}$ xs. P xs ⇒ P ($v_{12}$ says $v_{95}$ eqn $v_{96}$::xs)) ∧
  (∀ $v_{12}$ $v_{97}$ $v_{98}$ xs. P xs ⇒ P ($v_{12}$ says $v_{97}$ lte $v_{98}$::xs)) ∧
  (∀ $v_{12}$ $v_{99}$ v100 xs. P xs ⇒ P ($v_{12}$ says $v_{99}$ lt v100::xs)) ∧
  (∀ $v_{14}$ $v_{15}$ xs. P xs ⇒ P ($v_{14}$ speaks_for $v_{15}$::xs)) ∧
  (∀ $v_{16}$ $v_{17}$ xs. P xs ⇒ P ($v_{16}$ controls $v_{17}$::xs)) ∧
  (∀ $v_{18}$ $v_{19}$ $v_{20}$ xs. P xs ⇒ P (reps $v_{18}$ $v_{19}$ $v_{20}$::xs)) ∧
  (∀ $v_{21}$ $v_{22}$ xs. P xs ⇒ P ($v_{21}$ domi $v_{22}$::xs)) ∧
```

```
    (∀ v_23 v_24 xs. P xs ⇒ P (v_23 eqi v_24::xs)) ∧
    (∀ v_25 v_26 xs. P xs ⇒ P (v_25 doms v_26::xs)) ∧
    (∀ v_27 v_28 xs. P xs ⇒ P (v_27 eqs v_28::xs)) ∧
    (∀ v_29 v_30 xs. P xs ⇒ P (v_29 eqn v_30::xs)) ∧
    (∀ v_31 v_32 xs. P xs ⇒ P (v_31 lte v_32::xs)) ∧
    (∀ v_33 v_34 xs. P xs ⇒ P (v_33 lt v_34::xs)) ⇒
    ∀ v. P v
```

[getPlCom_def]

```
 ⊢ (getPlCom [] = invalidPlCommand) ∧
   (∀ xs cmd.
      getPlCom
        (Name PlatoonLeader says prop (SOME (SLc (PL cmd))))::
             xs) =
      cmd) ∧ (∀ xs. getPlCom (TT::xs) = getPlCom xs) ∧
   (∀ xs. getPlCom (FF::xs) = getPlCom xs) ∧
   (∀ xs v_2. getPlCom (prop v_2::xs) = getPlCom xs) ∧
   (∀ xs v_3. getPlCom (notf v_3::xs) = getPlCom xs) ∧
   (∀ xs v_5 v_4. getPlCom (v_4 andf v_5::xs) = getPlCom xs) ∧
   (∀ xs v_7 v_6. getPlCom (v_6 orf v_7::xs) = getPlCom xs) ∧
   (∀ xs v_9 v_8. getPlCom (v_8 impf v_9::xs) = getPlCom xs) ∧
   (∀ xs v_11 v_10. getPlCom (v_10 eqf v_11::xs) = getPlCom xs) ∧
   (∀ xs v_12. getPlCom (v_12 says TT::xs) = getPlCom xs) ∧
   (∀ xs v_12. getPlCom (v_12 says FF::xs) = getPlCom xs) ∧
   (∀ xs v134.
      getPlCom (Name v134 says prop NONE::xs) = getPlCom xs) ∧
   (∀ xs v146.
      getPlCom
        (Name PlatoonLeader says prop (SOME (ESCc v146))::xs) =
      getPlCom xs) ∧
   (∀ xs v151.
      getPlCom
        (Name PlatoonLeader says prop (SOME (SLc (PSG v151))))::
             xs) =
      getPlCom xs) ∧
   (∀ xs v144.
      getPlCom
        (Name PlatoonSergeant says prop (SOME v144)::xs) =
      getPlCom xs) ∧
   (∀ xs v_68 v136 v135.
      getPlCom (v135 meet v136 says prop v_68::xs) =
      getPlCom xs) ∧
   (∀ xs v_68 v138 v137.
      getPlCom (v137 quoting v138 says prop v_68::xs) =
      getPlCom xs) ∧
   (∀ xs v_69 v_12.
      getPlCom (v_12 says notf v_69::xs) = getPlCom xs) ∧
   (∀ xs v_71 v_70 v_12.
      getPlCom (v_12 says (v_70 andf v_71)::xs) = getPlCom xs) ∧
```

($\forall xs\ v_{73}\ v_{72}\ v_{12}$.
getPlCom (v_{12} says (v_{72} orf v_{73})::xs) = getPlCom xs) $\land$
($\forall xs\ v_{75}\ v_{74}\ v_{12}$.
getPlCom (v_{12} says (v_{74} impf v_{75})::xs) = getPlCom xs) $\land$
($\forall xs\ v_{77}\ v_{76}\ v_{12}$.
getPlCom (v_{12} says (v_{76} eqf v_{77})::xs) = getPlCom xs) $\land$
($\forall xs\ v_{79}\ v_{78}\ v_{12}$.
getPlCom (v_{12} says v_{78} says v_{79}::xs) = getPlCom xs) $\land$
($\forall xs\ v_{81}\ v_{80}\ v_{12}$.
getPlCom (v_{12} says v_{80} speaks_for v_{81}::xs) =
getPlCom xs) $\land$
($\forall xs\ v_{83}\ v_{82}\ v_{12}$.
getPlCom (v_{12} says v_{82} controls v_{83}::xs) = getPlCom xs) $\land$
($\forall xs\ v_{86}\ v_{85}\ v_{84}\ v_{12}$.
getPlCom (v_{12} says reps v_{84} v_{85} v_{86}::xs) = getPlCom xs) $\land$
($\forall xs\ v_{88}\ v_{87}\ v_{12}$.
getPlCom (v_{12} says v_{87} domi v_{88}::xs) = getPlCom xs) $\land$
($\forall xs\ v_{90}\ v_{89}\ v_{12}$.
getPlCom (v_{12} says v_{89} eqi v_{90}::xs) = getPlCom xs) $\land$
($\forall xs\ v_{92}\ v_{91}\ v_{12}$.
getPlCom (v_{12} says v_{91} doms v_{92}::xs) = getPlCom xs) $\land$
($\forall xs\ v_{94}\ v_{93}\ v_{12}$.
getPlCom (v_{12} says v_{93} eqs v_{94}::xs) = getPlCom xs) $\land$
($\forall xs\ v_{96}\ v_{95}\ v_{12}$.
getPlCom (v_{12} says v_{95} eqn v_{96}::xs) = getPlCom xs) $\land$
($\forall xs\ v_{98}\ v_{97}\ v_{12}$.
getPlCom (v_{12} says v_{97} lte v_{98}::xs) = getPlCom xs) $\land$
($\forall xs\ v_{99}\ v_{12}\ v100$.
getPlCom (v_{12} says v_{99} lt $v100$::xs) = getPlCom xs) $\land$
($\forall xs\ v_{15}\ v_{14}$.
getPlCom (v_{14} speaks_for v_{15}::xs) = getPlCom xs) $\land$
($\forall xs\ v_{17}\ v_{16}$.
getPlCom (v_{16} controls v_{17}::xs) = getPlCom xs) $\land$
($\forall xs\ v_{20}\ v_{19}\ v_{18}$.
getPlCom (reps v_{18} v_{19} v_{20}::xs) = getPlCom xs) $\land$
($\forall xs\ v_{22}\ v_{21}$. getPlCom ($v_{21}$ domi v_{22}::xs) = getPlCom xs) $\land$
($\forall xs\ v_{24}\ v_{23}$. getPlCom ($v_{23}$ eqi v_{24}::xs) = getPlCom xs) $\land$
($\forall xs\ v_{26}\ v_{25}$. getPlCom ($v_{25}$ doms v_{26}::xs) = getPlCom xs) $\land$
($\forall xs\ v_{28}\ v_{27}$. getPlCom ($v_{27}$ eqs v_{28}::xs) = getPlCom xs) $\land$
($\forall xs\ v_{30}\ v_{29}$. getPlCom ($v_{29}$ eqn v_{30}::xs) = getPlCom xs) $\land$
($\forall xs\ v_{32}\ v_{31}$. getPlCom ($v_{31}$ lte v_{32}::xs) = getPlCom xs) $\land$
$\forall xs\ v_{34}\ v_{33}$. getPlCom ($v_{33}$ lt v_{34}::xs) = getPlCom xs

[getPlCom_ind]

$\vdash \forall P$.
P [] $\land$
($\forall cmd\ xs$.
P
(Name PlatoonLeader says prop (SOME (SLc (PL 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)) ∧
(∀v146 xs.
  P xs ⇒
  P
    (Name PlatoonLeader says prop (SOME (ESCc v146))::
         xs)) ∧
(∀v151 xs.
  P xs ⇒
  P
    (Name PlatoonLeader says
     prop (SOME (SLc (PSG v151)))::xs)) ∧
(∀v144 xs.
  P xs ⇒
  P (Name PlatoonSergeant 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
```

[getPsgCom_def]

```
 ⊢ (getPsgCom [] = invalidPsgCommand) ∧
   (∀ xs cmd.
      getPsgCom
        (Name PlatoonSergeant says prop (SOME (SLc (PSG cmd)))::
             xs) =
      cmd) ∧ (∀ xs. getPsgCom (TT::xs) = getPsgCom xs) ∧
   (∀ xs. getPsgCom (FF::xs) = getPsgCom xs) ∧
   (∀ xs v2. getPsgCom (prop v2::xs) = getPsgCom xs) ∧
   (∀ xs v3. getPsgCom (notf v3::xs) = getPsgCom xs) ∧
   (∀ xs v5 v4. getPsgCom (v4 andf v5::xs) = getPsgCom xs) ∧
   (∀ xs v7 v6. getPsgCom (v6 orf v7::xs) = getPsgCom xs) ∧
   (∀ xs v9 v8. getPsgCom (v8 impf v9::xs) = getPsgCom xs) ∧
   (∀ xs v11 v10. getPsgCom (v10 eqf v11::xs) = getPsgCom xs) ∧
   (∀ xs v12. getPsgCom (v12 says TT::xs) = getPsgCom xs) ∧
   (∀ xs v12. getPsgCom (v12 says FF::xs) = getPsgCom xs) ∧
   (∀ xs v134.
      getPsgCom (Name v134 says prop NONE::xs) = getPsgCom xs) ∧
   (∀ xs v144.
      getPsgCom (Name PlatoonLeader says prop (SOME v144)::xs) =
      getPsgCom xs) ∧
   (∀ xs v146.
      getPsgCom
        (Name PlatoonSergeant says prop (SOME (ESCc v146))::
             xs) =
      getPsgCom xs) ∧
   (∀ xs v150.
      getPsgCom
        (Name PlatoonSergeant says prop (SOME (SLc (PL v150)))::
             xs) =
      getPsgCom xs) ∧
   (∀ xs v68 v136 v135.
      getPsgCom (v135 meet v136 says prop v68::xs) =
      getPsgCom xs) ∧
   (∀ xs v68 v138 v137.
      getPsgCom (v137 quoting v138 says prop v68::xs) =
      getPsgCom xs) ∧
   (∀ xs v69 v12.
      getPsgCom (v12 says notf v69::xs) = getPsgCom xs) ∧
```

$$\begin{array}{l}
(\forall xs\ v_{71}\ v_{70}\ v_{12}.\\
\quad \texttt{getPsgCom}\ (v_{12}\ \texttt{says}\ (v_{70}\ \texttt{andf}\ v_{71})\texttt{::}xs)\ \texttt{=}\ \texttt{getPsgCom}\ xs)\ \land\\
(\forall xs\ v_{73}\ v_{72}\ v_{12}.\\
\quad \texttt{getPsgCom}\ (v_{12}\ \texttt{says}\ (v_{72}\ \texttt{orf}\ v_{73})\texttt{::}xs)\ \texttt{=}\ \texttt{getPsgCom}\ xs)\ \land\\
(\forall xs\ v_{75}\ v_{74}\ v_{12}.\\
\quad \texttt{getPsgCom}\ (v_{12}\ \texttt{says}\ (v_{74}\ \texttt{impf}\ v_{75})\texttt{::}xs)\ \texttt{=}\ \texttt{getPsgCom}\ xs)\ \land\\
(\forall xs\ v_{77}\ v_{76}\ v_{12}.\\
\quad \texttt{getPsgCom}\ (v_{12}\ \texttt{says}\ (v_{76}\ \texttt{eqf}\ v_{77})\texttt{::}xs)\ \texttt{=}\ \texttt{getPsgCom}\ xs)\ \land\\
(\forall xs\ v_{79}\ v_{78}\ v_{12}.\\
\quad \texttt{getPsgCom}\ (v_{12}\ \texttt{says}\ v_{78}\ \texttt{says}\ v_{79}\texttt{::}xs)\ \texttt{=}\ \texttt{getPsgCom}\ xs)\ \land\\
(\forall xs\ v_{81}\ v_{80}\ v_{12}.\\
\quad \texttt{getPsgCom}\ (v_{12}\ \texttt{says}\ v_{80}\ \texttt{speaks_for}\ v_{81}\texttt{::}xs)\ \texttt{=}\\
\quad \texttt{getPsgCom}\ xs)\ \land\\
(\forall xs\ v_{83}\ v_{82}\ v_{12}.\\
\quad \texttt{getPsgCom}\ (v_{12}\ \texttt{says}\ v_{82}\ \texttt{controls}\ v_{83}\texttt{::}xs)\ \texttt{=}\\
\quad \texttt{getPsgCom}\ xs)\ \land\\
(\forall xs\ v_{86}\ v_{85}\ v_{84}\ v_{12}.\\
\quad \texttt{getPsgCom}\ (v_{12}\ \texttt{says}\ \texttt{reps}\ v_{84}\ v_{85}\ v_{86}\texttt{::}xs)\ \texttt{=}\\
\quad \texttt{getPsgCom}\ xs)\ \land\\
(\forall xs\ v_{88}\ v_{87}\ v_{12}.\\
\quad \texttt{getPsgCom}\ (v_{12}\ \texttt{says}\ v_{87}\ \texttt{domi}\ v_{88}\texttt{::}xs)\ \texttt{=}\ \texttt{getPsgCom}\ xs)\ \land\\
(\forall xs\ v_{90}\ v_{89}\ v_{12}.\\
\quad \texttt{getPsgCom}\ (v_{12}\ \texttt{says}\ v_{89}\ \texttt{eqi}\ v_{90}\texttt{::}xs)\ \texttt{=}\ \texttt{getPsgCom}\ xs)\ \land\\
(\forall xs\ v_{92}\ v_{91}\ v_{12}.\\
\quad \texttt{getPsgCom}\ (v_{12}\ \texttt{says}\ v_{91}\ \texttt{doms}\ v_{92}\texttt{::}xs)\ \texttt{=}\ \texttt{getPsgCom}\ xs)\ \land\\
(\forall xs\ v_{94}\ v_{93}\ v_{12}.\\
\quad \texttt{getPsgCom}\ (v_{12}\ \texttt{says}\ v_{93}\ \texttt{eqs}\ v_{94}\texttt{::}xs)\ \texttt{=}\ \texttt{getPsgCom}\ xs)\ \land\\
(\forall xs\ v_{96}\ v_{95}\ v_{12}.\\
\quad \texttt{getPsgCom}\ (v_{12}\ \texttt{says}\ v_{95}\ \texttt{eqn}\ v_{96}\texttt{::}xs)\ \texttt{=}\ \texttt{getPsgCom}\ xs)\ \land\\
(\forall xs\ v_{98}\ v_{97}\ v_{12}.\\
\quad \texttt{getPsgCom}\ (v_{12}\ \texttt{says}\ v_{97}\ \texttt{lte}\ v_{98}\texttt{::}xs)\ \texttt{=}\ \texttt{getPsgCom}\ xs)\ \land\\
(\forall xs\ v_{99}\ v_{12}\ \mathit{v100}.\\
\quad \texttt{getPsgCom}\ (v_{12}\ \texttt{says}\ v_{99}\ \texttt{lt}\ \mathit{v100}\texttt{::}xs)\ \texttt{=}\ \texttt{getPsgCom}\ xs)\ \land\\
(\forall xs\ v_{15}\ v_{14}.\\
\quad \texttt{getPsgCom}\ (v_{14}\ \texttt{speaks_for}\ v_{15}\texttt{::}xs)\ \texttt{=}\ \texttt{getPsgCom}\ xs)\ \land\\
(\forall xs\ v_{17}\ v_{16}.\\
\quad \texttt{getPsgCom}\ (v_{16}\ \texttt{controls}\ v_{17}\texttt{::}xs)\ \texttt{=}\ \texttt{getPsgCom}\ xs)\ \land\\
(\forall xs\ v_{20}\ v_{19}\ v_{18}.\\
\quad \texttt{getPsgCom}\ (\texttt{reps}\ v_{18}\ v_{19}\ v_{20}\texttt{::}xs)\ \texttt{=}\ \texttt{getPsgCom}\ xs)\ \land\\
(\forall xs\ v_{22}\ v_{21}.\ \texttt{getPsgCom}\ (v_{21}\ \texttt{domi}\ v_{22}\texttt{::}xs)\ \texttt{=}\ \texttt{getPsgCom}\ xs)\ \land\\
(\forall xs\ v_{24}\ v_{23}.\ \texttt{getPsgCom}\ (v_{23}\ \texttt{eqi}\ v_{24}\texttt{::}xs)\ \texttt{=}\ \texttt{getPsgCom}\ xs)\ \land\\
(\forall xs\ v_{26}\ v_{25}.\ \texttt{getPsgCom}\ (v_{25}\ \texttt{doms}\ v_{26}\texttt{::}xs)\ \texttt{=}\ \texttt{getPsgCom}\ xs)\ \land\\
(\forall xs\ v_{28}\ v_{27}.\ \texttt{getPsgCom}\ (v_{27}\ \texttt{eqs}\ v_{28}\texttt{::}xs)\ \texttt{=}\ \texttt{getPsgCom}\ xs)\ \land\\
(\forall xs\ v_{30}\ v_{29}.\ \texttt{getPsgCom}\ (v_{29}\ \texttt{eqn}\ v_{30}\texttt{::}xs)\ \texttt{=}\ \texttt{getPsgCom}\ xs)\ \land\\
(\forall xs\ v_{32}\ v_{31}.\ \texttt{getPsgCom}\ (v_{31}\ \texttt{lte}\ v_{32}\texttt{::}xs)\ \texttt{=}\ \texttt{getPsgCom}\ xs)\ \land\\
\forall xs\ v_{34}\ v_{33}.\ \texttt{getPsgCom}\ (v_{33}\ \texttt{lt}\ v_{34}\texttt{::}xs)\ \texttt{=}\ \texttt{getPsgCom}\ xs
\end{array}$$

[getPsgCom_ind]

$\vdash\ \forall P.$

```
P [] ∧
(∀ cmd xs.
   P
     (Name PlatoonSergeant says
      prop (SOME (SLc (PSG 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 PlatoonSergeant says prop (SOME (ESCc v146))::
          xs)) ∧
(∀ v150 xs.
   P xs ⇒
   P
     (Name PlatoonSergeant says
      prop (SOME (SLc (PL v150)))::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
```

[getRecon_def]

```
 ⊢ (getRecon [] = [NONE]) ∧
   (∀ xs.
      getRecon
        (Name PlatoonLeader says prop (SOME (SLc (PL recon)))::
               xs) =
      [SOME (SLc (PL recon))]) ∧
   (∀ xs. getRecon (TT::xs) = getRecon xs) ∧
   (∀ xs. getRecon (FF::xs) = getRecon xs) ∧
   (∀ xs v2. getRecon (prop v2::xs) = getRecon xs) ∧
   (∀ xs v3. getRecon (notf v3::xs) = getRecon xs) ∧
   (∀ xs v5 v4. getRecon (v4 andf v5::xs) = getRecon xs) ∧
   (∀ xs v7 v6. getRecon (v6 orf v7::xs) = getRecon xs) ∧
   (∀ xs v9 v8. getRecon (v8 impf v9::xs) = getRecon xs) ∧
   (∀ xs v11 v10. getRecon (v10 eqf v11::xs) = getRecon xs) ∧
   (∀ xs v12. getRecon (v12 says TT::xs) = getRecon xs) ∧
   (∀ xs v12. getRecon (v12 says FF::xs) = getRecon xs) ∧
   (∀ xs v134.
      getRecon (Name v134 says prop NONE::xs) = getRecon xs) ∧
   (∀ xs v146.
      getRecon
        (Name PlatoonLeader says prop (SOME (ESCc v146))::xs) =
      getRecon xs) ∧
   (∀ xs.
      getRecon
        (Name PlatoonLeader says
         prop (SOME (SLc (PL receiveMission)))::xs) =
      getRecon xs) ∧
   (∀ xs.
      getRecon
        (Name PlatoonLeader says prop (SOME (SLc (PL warno)))::
               xs) =
      getRecon xs) ∧
   (∀ xs.
      getRecon
```

```
       (Name PlatoonLeader says
        prop (SOME (SLc (PL tentativePlan)))::xs) =
     getRecon xs) ∧
  (∀ xs.
     getRecon
       (Name PlatoonLeader says
        prop (SOME (SLc (PL report1)))::xs) =
     getRecon xs) ∧
  (∀ xs.
     getRecon
       (Name PlatoonLeader says
        prop (SOME (SLc (PL completePlan)))::xs) =
     getRecon xs) ∧
  (∀ xs.
     getRecon
       (Name PlatoonLeader says prop (SOME (SLc (PL opoid)))::
            xs) =
     getRecon xs) ∧
  (∀ xs.
     getRecon
       (Name PlatoonLeader says
        prop (SOME (SLc (PL supervise)))::xs) =
     getRecon xs) ∧
  (∀ xs.
     getRecon
       (Name PlatoonLeader says
        prop (SOME (SLc (PL report2)))::xs) =
     getRecon xs) ∧
  (∀ xs.
     getRecon
       (Name PlatoonLeader says
        prop (SOME (SLc (PL complete)))::xs) =
     getRecon xs) ∧
  (∀ xs.
     getRecon
       (Name PlatoonLeader says
        prop (SOME (SLc (PL plIncomplete)))::xs) =
     getRecon xs) ∧
  (∀ xs.
     getRecon
       (Name PlatoonLeader says
        prop (SOME (SLc (PL invalidPlCommand)))::xs) =
     getRecon xs) ∧
  (∀ xs v151.
     getRecon
       (Name PlatoonLeader says prop (SOME (SLc (PSG v151)))::
            xs) =
     getRecon xs) ∧
  (∀ xs v144.
     getRecon
       (Name PlatoonSergeant says prop (SOME v144)::xs) =
     getRecon xs) ∧
  (∀xs v68 v136 v135.
     getRecon (v135 meet v136 says prop v68::xs) =
     getRecon xs) ∧
  (∀xs v68 v138 v137.
     getRecon (v137 quoting v138 says prop v68::xs) =
     getRecon xs) ∧
  (∀xs v69 v12.
     getRecon (v12 says notf v69::xs) = getRecon xs) ∧
  (∀xs v71 v70 v12.
     getRecon (v12 says (v70 andf v71)::xs) = getRecon xs) ∧
  (∀xs v73 v72 v12.
     getRecon (v12 says (v72 orf v73)::xs) = getRecon xs) ∧
  (∀xs v75 v74 v12.
     getRecon (v12 says (v74 impf v75)::xs) = getRecon xs) ∧
  (∀xs v77 v76 v12.
     getRecon (v12 says (v76 eqf v77)::xs) = getRecon xs) ∧
  (∀xs v79 v78 v12.
     getRecon (v12 says v78 says v79::xs) = getRecon xs) ∧
  (∀xs v81 v80 v12.
     getRecon (v12 says v80 speaks_for v81::xs) =
     getRecon xs) ∧
  (∀xs v83 v82 v12.
     getRecon (v12 says v82 controls v83::xs) = getRecon xs) ∧
  (∀xs v86 v85 v84 v12.
     getRecon (v12 says reps v84 v85 v86::xs) = getRecon xs) ∧
  (∀xs v88 v87 v12.
     getRecon (v12 says v87 domi v88::xs) = getRecon xs) ∧
  (∀xs v90 v89 v12.
     getRecon (v12 says v89 eqi v90::xs) = getRecon xs) ∧
  (∀xs v92 v91 v12.
     getRecon (v12 says v91 doms v92::xs) = getRecon xs) ∧
  (∀xs v94 v93 v12.
     getRecon (v12 says v93 eqs v94::xs) = getRecon xs) ∧
  (∀xs v96 v95 v12.
     getRecon (v12 says v95 eqn v96::xs) = getRecon xs) ∧
  (∀xs v98 v97 v12.
     getRecon (v12 says v97 lte v98::xs) = getRecon xs) ∧
  (∀xs v99 v12 v100.
     getRecon (v12 says v99 lt v100::xs) = getRecon xs) ∧
  (∀xs v15 v14.
     getRecon (v14 speaks_for v15::xs) = getRecon xs) ∧
  (∀xs v17 v16.
     getRecon (v16 controls v17::xs) = getRecon xs) ∧
  (∀xs v20 v19 v18.
     getRecon (reps v18 v19 v20::xs) = getRecon xs) ∧
  (∀xs v22 v21. getRecon (v21 domi v22::xs) = getRecon xs) ∧
```

```
   (∀xs v24 v23. getRecon (v23 eqi v24::xs) = getRecon xs) ∧
   (∀xs v26 v25. getRecon (v25 doms v26::xs) = getRecon xs) ∧
   (∀xs v28 v27. getRecon (v27 eqs v28::xs) = getRecon xs) ∧
   (∀xs v30 v29. getRecon (v29 eqn v30::xs) = getRecon xs) ∧
   (∀xs v32 v31. getRecon (v31 lte v32::xs) = getRecon xs) ∧
   ∀xs v34 v33. getRecon (v33 lt v34::xs) = getRecon xs
```

[getRecon_ind]

```
 ⊢ ∀P.
     P [] ∧
     (∀xs.
        P
          (Name PlatoonLeader says
           prop (SOME (SLc (PL recon)))::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)) ∧
     (∀v146 xs.
        P xs ⇒
        P
          (Name PlatoonLeader says prop (SOME (ESCc v146))::
             xs)) ∧
     (∀xs.
        P xs ⇒
        P
          (Name PlatoonLeader says
           prop (SOME (SLc (PL receiveMission)))::xs)) ∧
     (∀xs.
        P xs ⇒
        P
          (Name PlatoonLeader says
           prop (SOME (SLc (PL warno)))::xs)) ∧
     (∀xs.
        P xs ⇒
        P
          (Name PlatoonLeader says
           prop (SOME (SLc (PL tentativePlan)))::xs)) ∧
     (∀xs.
        P xs ⇒
        P
          (Name PlatoonLeader says
           prop (SOME (SLc (PL report1)))::xs)) ∧
```

```
(∀ xs.
   P xs ⇒
   P
     (Name PlatoonLeader says
      prop (SOME (SLc (PL completePlan)))::xs)) ∧
(∀ xs.
   P xs ⇒
   P
     (Name PlatoonLeader says
      prop (SOME (SLc (PL opoid)))::xs)) ∧
(∀ xs.
   P xs ⇒
   P
     (Name PlatoonLeader says
      prop (SOME (SLc (PL supervise)))::xs)) ∧
(∀ xs.
   P xs ⇒
   P
     (Name PlatoonLeader says
      prop (SOME (SLc (PL report2)))::xs)) ∧
(∀ xs.
   P xs ⇒
   P
     (Name PlatoonLeader says
      prop (SOME (SLc (PL complete)))::xs)) ∧
(∀ xs.
   P xs ⇒
   P
     (Name PlatoonLeader says
      prop (SOME (SLc (PL plIncomplete)))::xs)) ∧
(∀ xs.
   P xs ⇒
   P
     (Name PlatoonLeader says
      prop (SOME (SLc (PL invalidPlCommand)))::xs)) ∧
(∀ v151 xs.
   P xs ⇒
   P
     (Name PlatoonLeader says
      prop (SOME (SLc (PSG v151)))::xs)) ∧
(∀ v144 xs.
   P xs ⇒
   P (Name PlatoonSergeant 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
```

[getReport_def]

```
 ⊢ (getReport [] = [NONE]) ∧
   (∀ xs.
      getReport
        (Name PlatoonLeader says
         prop (SOME (SLc (PL report1)))::xs) =
      [SOME (SLc (PL report1))]) ∧
   (∀ xs. getReport (TT::xs) = getReport xs) ∧
   (∀ xs. getReport (FF::xs) = getReport xs) ∧
   (∀ xs v2. getReport (prop v2::xs) = getReport xs) ∧
   (∀ xs v3. getReport (notf v3::xs) = getReport xs) ∧
   (∀ xs v5 v4. getReport (v4 andf v5::xs) = getReport xs) ∧
   (∀ xs v7 v6. getReport (v6 orf v7::xs) = getReport xs) ∧
   (∀ xs v9 v8. getReport (v8 impf v9::xs) = getReport xs) ∧
   (∀ xs v11 v10. getReport (v10 eqf v11::xs) = getReport xs) ∧
   (∀ xs v12. getReport (v12 says TT::xs) = getReport xs) ∧
   (∀ xs v12. getReport (v12 says FF::xs) = getReport xs) ∧
   (∀ xs v134.
      getReport (Name v134 says prop NONE::xs) = getReport xs) ∧
   (∀ xs v146.
   getReport
     (Name PlatoonLeader says prop (SOME (ESCc v146))::xs) =
   getReport xs) ∧
(∀ xs.
   getReport
     (Name PlatoonLeader says
      prop (SOME (SLc (PL receiveMission)))::xs) =
   getReport xs) ∧
(∀ xs.
   getReport
     (Name PlatoonLeader says prop (SOME (SLc (PL warno)))::
          xs) =
   getReport xs) ∧
(∀ xs.
   getReport
     (Name PlatoonLeader says
      prop (SOME (SLc (PL tentativePlan)))::xs) =
   getReport xs) ∧
(∀ xs.
   getReport
     (Name PlatoonLeader says prop (SOME (SLc (PL recon)))::
          xs) =
   getReport xs) ∧
(∀ xs.
   getReport
     (Name PlatoonLeader says
      prop (SOME (SLc (PL completePlan)))::xs) =
   getReport xs) ∧
(∀ xs.
   getReport
     (Name PlatoonLeader says prop (SOME (SLc (PL opoid)))::
          xs) =
   getReport xs) ∧
(∀ xs.
   getReport
     (Name PlatoonLeader says
      prop (SOME (SLc (PL supervise)))::xs) =
   getReport xs) ∧
(∀ xs.
   getReport
     (Name PlatoonLeader says
      prop (SOME (SLc (PL report2)))::xs) =
   getReport xs) ∧
(∀ xs.
   getReport
     (Name PlatoonLeader says
      prop (SOME (SLc (PL complete)))::xs) =
   getReport xs) ∧
(∀ xs.
     getReport
       (Name PlatoonLeader says
        prop (SOME (SLc (PL plIncomplete)))::xs) =
     getReport xs) ∧
  (∀ xs.
     getReport
       (Name PlatoonLeader says
        prop (SOME (SLc (PL invalidPlCommand)))::xs) =
     getReport xs) ∧
  (∀ xs v151.
     getReport
       (Name PlatoonLeader says prop (SOME (SLc (PSG v151)))::
            xs) =
     getReport xs) ∧
  (∀ xs v144.
     getReport
       (Name PlatoonSergeant says prop (SOME v144)::xs) =
     getReport xs) ∧
  (∀ xs v68 v136 v135.
     getReport (v135 meet v136 says prop v68::xs) =
     getReport xs) ∧
  (∀ xs v68 v138 v137.
     getReport (v137 quoting v138 says prop v68::xs) =
     getReport xs) ∧
  (∀ xs v69 v12.
     getReport (v12 says notf v69::xs) = getReport xs) ∧
  (∀ xs v71 v70 v12.
     getReport (v12 says (v70 andf v71)::xs) = getReport xs) ∧
  (∀ xs v73 v72 v12.
     getReport (v12 says (v72 orf v73)::xs) = getReport xs) ∧
  (∀ xs v75 v74 v12.
     getReport (v12 says (v74 impf v75)::xs) = getReport xs) ∧
  (∀ xs v77 v76 v12.
     getReport (v12 says (v76 eqf v77)::xs) = getReport xs) ∧
  (∀ xs v79 v78 v12.
     getReport (v12 says v78 says v79::xs) = getReport xs) ∧
  (∀ xs v81 v80 v12.
     getReport (v12 says v80 speaks_for v81::xs) =
     getReport xs) ∧
  (∀ xs v83 v82 v12.
     getReport (v12 says v82 controls v83::xs) =
     getReport xs) ∧
  (∀ xs v86 v85 v84 v12.
     getReport (v12 says reps v84 v85 v86::xs) =
     getReport xs) ∧
  (∀ xs v88 v87 v12.
     getReport (v12 says v87 domi v88::xs) = getReport xs) ∧
  (∀ xs v90 v89 v12.
     getReport (v12 says v89 eqi v90::xs) = getReport xs) ∧
  (∀ xs v92 v91 v12.
     getReport (v12 says v91 doms v92::xs) = getReport xs) ∧
  (∀ xs v94 v93 v12.
     getReport (v12 says v93 eqs v94::xs) = getReport xs) ∧
  (∀ xs v96 v95 v12.
     getReport (v12 says v95 eqn v96::xs) = getReport xs) ∧
  (∀ xs v98 v97 v12.
     getReport (v12 says v97 lte v98::xs) = getReport xs) ∧
  (∀ xs v99 v12 v100.
     getReport (v12 says v99 lt v100::xs) = getReport xs) ∧
  (∀ xs v15 v14.
     getReport (v14 speaks_for v15::xs) = getReport xs) ∧
  (∀ xs v17 v16.
     getReport (v16 controls v17::xs) = getReport xs) ∧
  (∀ xs v20 v19 v18.
     getReport (reps v18 v19 v20::xs) = getReport xs) ∧
  (∀ xs v22 v21. getReport (v21 domi v22::xs) = getReport xs) ∧
  (∀ xs v24 v23. getReport (v23 eqi v24::xs) = getReport xs) ∧
  (∀ xs v26 v25. getReport (v25 doms v26::xs) = getReport xs) ∧
  (∀ xs v28 v27. getReport (v27 eqs v28::xs) = getReport xs) ∧
  (∀ xs v30 v29. getReport (v29 eqn v30::xs) = getReport xs) ∧
  (∀ xs v32 v31. getReport (v31 lte v32::xs) = getReport xs) ∧
  ∀ xs v34 v33. getReport (v33 lt v34::xs) = getReport xs
```

[getReport_ind]

```
 ⊢ ∀ P.
     P [] ∧
     (∀ xs.
        P
          (Name PlatoonLeader says
           prop (SOME (SLc (PL report1)))::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)) ∧
     (∀ v146 xs.
        P xs ⇒
        P
          (Name PlatoonLeader says prop (SOME (ESCc v146))::
               xs)) ∧
     (∀ xs.
        P xs ⇒
        P
     (Name PlatoonLeader says
      prop (SOME (SLc (PL receiveMission)))::xs)) ∧
(∀ xs.
   P xs ⇒
   P
     (Name PlatoonLeader says
      prop (SOME (SLc (PL warno)))::xs)) ∧
(∀ xs.
   P xs ⇒
   P
     (Name PlatoonLeader says
      prop (SOME (SLc (PL tentativePlan)))::xs)) ∧
(∀ xs.
   P xs ⇒
   P
     (Name PlatoonLeader says
      prop (SOME (SLc (PL recon)))::xs)) ∧
(∀ xs.
   P xs ⇒
   P
     (Name PlatoonLeader says
      prop (SOME (SLc (PL completePlan)))::xs)) ∧
(∀ xs.
   P xs ⇒
   P
     (Name PlatoonLeader says
      prop (SOME (SLc (PL opoid)))::xs)) ∧
(∀ xs.
   P xs ⇒
   P
     (Name PlatoonLeader says
      prop (SOME (SLc (PL supervise)))::xs)) ∧
(∀ xs.
   P xs ⇒
   P
     (Name PlatoonLeader says
      prop (SOME (SLc (PL report2)))::xs)) ∧
(∀ xs.
   P xs ⇒
   P
     (Name PlatoonLeader says
      prop (SOME (SLc (PL complete)))::xs)) ∧
(∀ xs.
   P xs ⇒
   P
     (Name PlatoonLeader says
      prop (SOME (SLc (PL plIncomplete)))::xs)) ∧
(∀ xs.
   P xs ⇒
```

```
       P
         (Name PlatoonLeader says
          prop (SOME (SLc (PL invalidPlCommand)))::xs)) ∧
    (∀ v151 xs.
       P xs ⇒
       P
         (Name PlatoonLeader says
          prop (SOME (SLc (PSG v151)))::xs)) ∧
    (∀ v144 xs.
       P xs ⇒
       P (Name PlatoonSergeant 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
```

[getTenativePlan_def]

```
 ⊢ (getTenativePlan [] = [NONE]) ∧
   (∀ xs.
   getTenativePlan
     (Name PlatoonLeader says
      prop (SOME (SLc (PL tentativePlan)))::xs) =
   [SOME (SLc (PL tentativePlan))]) ∧
(∀xs. getTenativePlan (TT::xs) = getTenativePlan xs) ∧
(∀xs. getTenativePlan (FF::xs) = getTenativePlan xs) ∧
(∀xs v2.
   getTenativePlan (prop v2::xs) = getTenativePlan xs) ∧
(∀xs v3.
   getTenativePlan (notf v3::xs) = getTenativePlan xs) ∧
(∀xs v5 v4.
   getTenativePlan (v4 andf v5::xs) = getTenativePlan xs) ∧
(∀xs v7 v6.
   getTenativePlan (v6 orf v7::xs) = getTenativePlan xs) ∧
(∀xs v9 v8.
   getTenativePlan (v8 impf v9::xs) = getTenativePlan xs) ∧
(∀xs v11 v10.
   getTenativePlan (v10 eqf v11::xs) = getTenativePlan xs) ∧
(∀xs v12.
   getTenativePlan (v12 says TT::xs) = getTenativePlan xs) ∧
(∀xs v12.
   getTenativePlan (v12 says FF::xs) = getTenativePlan xs) ∧
(∀xs v134.
   getTenativePlan (Name v134 says prop NONE::xs) =
   getTenativePlan xs) ∧
(∀xs v146.
   getTenativePlan
     (Name PlatoonLeader says prop (SOME (ESCc v146))::xs) =
   getTenativePlan xs) ∧
(∀xs.
   getTenativePlan
     (Name PlatoonLeader says
      prop (SOME (SLc (PL receiveMission)))::xs) =
   getTenativePlan xs) ∧
(∀xs.
   getTenativePlan
     (Name PlatoonLeader says prop (SOME (SLc (PL warno)))::
          xs) =
   getTenativePlan xs) ∧
(∀xs.
   getTenativePlan
     (Name PlatoonLeader says prop (SOME (SLc (PL recon)))::
          xs) =
   getTenativePlan xs) ∧
(∀xs.
   getTenativePlan
     (Name PlatoonLeader says
      prop (SOME (SLc (PL report1)))::xs) =
   getTenativePlan xs) ∧
```

```
(∀ xs.
   getTenativePlan
     (Name PlatoonLeader says
      prop (SOME (SLc (PL completePlan)))::xs) =
   getTenativePlan xs) ∧
(∀ xs.
   getTenativePlan
     (Name PlatoonLeader says prop (SOME (SLc (PL opoid)))::
          xs) =
   getTenativePlan xs) ∧
(∀ xs.
   getTenativePlan
     (Name PlatoonLeader says
      prop (SOME (SLc (PL supervise)))::xs) =
   getTenativePlan xs) ∧
(∀ xs.
   getTenativePlan
     (Name PlatoonLeader says
      prop (SOME (SLc (PL report2)))::xs) =
   getTenativePlan xs) ∧
(∀ xs.
   getTenativePlan
     (Name PlatoonLeader says
      prop (SOME (SLc (PL complete)))::xs) =
   getTenativePlan xs) ∧
(∀ xs.
   getTenativePlan
     (Name PlatoonLeader says
      prop (SOME (SLc (PL plIncomplete)))::xs) =
   getTenativePlan xs) ∧
(∀ xs.
   getTenativePlan
     (Name PlatoonLeader says
      prop (SOME (SLc (PL invalidPlCommand)))::xs) =
   getTenativePlan xs) ∧
(∀ xs v151.
   getTenativePlan
     (Name PlatoonLeader says prop (SOME (SLc (PSG v151)))::
          xs) =
   getTenativePlan xs) ∧
(∀ xs v144.
   getTenativePlan
     (Name PlatoonSergeant says prop (SOME v144)::xs) =
   getTenativePlan xs) ∧
(∀ xs v68 v136 v135.
   getTenativePlan (v135 meet v136 says prop v68::xs) =
   getTenativePlan xs) ∧
(∀ xs v68 v138 v137.
   getTenativePlan (v137 quoting v138 says prop v68::xs) =
     getTenativePlan xs) ∧
  (∀ xs v69 v12.
     getTenativePlan (v12 says notf v69::xs) =
     getTenativePlan xs) ∧
  (∀ xs v71 v70 v12.
     getTenativePlan (v12 says (v70 andf v71)::xs) =
     getTenativePlan xs) ∧
  (∀ xs v73 v72 v12.
     getTenativePlan (v12 says (v72 orf v73)::xs) =
     getTenativePlan xs) ∧
  (∀ xs v75 v74 v12.
     getTenativePlan (v12 says (v74 impf v75)::xs) =
     getTenativePlan xs) ∧
  (∀ xs v77 v76 v12.
     getTenativePlan (v12 says (v76 eqf v77)::xs) =
     getTenativePlan xs) ∧
  (∀ xs v79 v78 v12.
     getTenativePlan (v12 says v78 says v79::xs) =
     getTenativePlan xs) ∧
  (∀ xs v81 v80 v12.
     getTenativePlan (v12 says v80 speaks_for v81::xs) =
     getTenativePlan xs) ∧
  (∀ xs v83 v82 v12.
     getTenativePlan (v12 says v82 controls v83::xs) =
     getTenativePlan xs) ∧
  (∀ xs v86 v85 v84 v12.
     getTenativePlan (v12 says reps v84 v85 v86::xs) =
     getTenativePlan xs) ∧
  (∀ xs v88 v87 v12.
     getTenativePlan (v12 says v87 domi v88::xs) =
     getTenativePlan xs) ∧
  (∀ xs v90 v89 v12.
     getTenativePlan (v12 says v89 eqi v90::xs) =
     getTenativePlan xs) ∧
  (∀ xs v92 v91 v12.
     getTenativePlan (v12 says v91 doms v92::xs) =
     getTenativePlan xs) ∧
  (∀ xs v94 v93 v12.
     getTenativePlan (v12 says v93 eqs v94::xs) =
     getTenativePlan xs) ∧
  (∀ xs v96 v95 v12.
     getTenativePlan (v12 says v95 eqn v96::xs) =
     getTenativePlan xs) ∧
  (∀ xs v98 v97 v12.
     getTenativePlan (v12 says v97 lte v98::xs) =
     getTenativePlan xs) ∧
  (∀ xs v99 v12 v100.
     getTenativePlan (v12 says v99 lt v100::xs) =
     getTenativePlan xs) ∧
```

```
(∀xs v15 v14.
   getTenativePlan (v14 speaks_for v15::xs) =
   getTenativePlan xs) ∧
(∀xs v17 v16.
   getTenativePlan (v16 controls v17::xs) =
   getTenativePlan xs) ∧
(∀xs v20 v19 v18.
   getTenativePlan (reps v18 v19 v20::xs) =
   getTenativePlan xs) ∧
(∀xs v22 v21.
   getTenativePlan (v21 domi v22::xs) = getTenativePlan xs) ∧
(∀xs v24 v23.
   getTenativePlan (v23 eqi v24::xs) = getTenativePlan xs) ∧
(∀xs v26 v25.
   getTenativePlan (v25 doms v26::xs) = getTenativePlan xs) ∧
(∀xs v28 v27.
   getTenativePlan (v27 eqs v28::xs) = getTenativePlan xs) ∧
(∀xs v30 v29.
   getTenativePlan (v29 eqn v30::xs) = getTenativePlan xs) ∧
(∀xs v32 v31.
   getTenativePlan (v31 lte v32::xs) = getTenativePlan xs) ∧
∀xs v34 v33.
  getTenativePlan (v33 lt v34::xs) = getTenativePlan xs
```

[getTenativePlan_ind]

```
⊢ ∀P.
    P [] ∧
    (∀xs.
       P
         (Name PlatoonLeader says
          prop (SOME (SLc (PL tentativePlan)))::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)) ∧
    (∀v146 xs.
       P xs ⇒
       P
         (Name PlatoonLeader says prop (SOME (ESCc v146))::
              xs)) ∧
    (∀xs.
       P xs ⇒
       P
```

```
          (Name PlatoonLeader says
           prop (SOME (SLc (PL receiveMission)))::xs)) ∧
    (∀ xs.
       P xs ⇒
       P
         (Name PlatoonLeader says
          prop (SOME (SLc (PL warno)))::xs)) ∧
    (∀ xs.
       P xs ⇒
       P
         (Name PlatoonLeader says
          prop (SOME (SLc (PL recon)))::xs)) ∧
    (∀ xs.
       P xs ⇒
       P
         (Name PlatoonLeader says
          prop (SOME (SLc (PL report1)))::xs)) ∧
    (∀ xs.
       P xs ⇒
       P
         (Name PlatoonLeader says
          prop (SOME (SLc (PL completePlan)))::xs)) ∧
    (∀ xs.
       P xs ⇒
       P
         (Name PlatoonLeader says
          prop (SOME (SLc (PL opoid)))::xs)) ∧
    (∀ xs.
       P xs ⇒
       P
         (Name PlatoonLeader says
          prop (SOME (SLc (PL supervise)))::xs)) ∧
    (∀ xs.
       P xs ⇒
       P
         (Name PlatoonLeader says
          prop (SOME (SLc (PL report2)))::xs)) ∧
    (∀ xs.
       P xs ⇒
       P
         (Name PlatoonLeader says
          prop (SOME (SLc (PL complete)))::xs)) ∧
    (∀ xs.
       P xs ⇒
       P
         (Name PlatoonLeader says
          prop (SOME (SLc (PL plIncomplete)))::xs)) ∧
    (∀ xs.
       P xs ⇒
    P
      (Name PlatoonLeader says
       prop (SOME (SLc (PL invalidPlCommand)))::xs)) ∧
 (∀ v151 xs.
    P xs ⇒
    P
      (Name PlatoonLeader says
       prop (SOME (SLc (PSG v151)))::xs)) ∧
 (∀ v144 xs.
    P xs ⇒
    P (Name PlatoonSergeant says prop (SOME v144)::xs)) ∧
 (∀ v135 v136 v_68 xs.
    P xs ⇒ P (v135 meet v136 says prop v_68::xs)) ∧
 (∀ v137 v138 v_68 xs.
    P xs ⇒ P (v137 quoting v138 says prop v_68::xs)) ∧
 (∀ v_12 v_69 xs. P xs ⇒ P (v_12 says notf v_69::xs)) ∧
 (∀ v_12 v_70 v_71 xs. P xs ⇒ P (v_12 says (v_70 andf v_71)::xs)) ∧
 (∀ v_12 v_72 v_73 xs. P xs ⇒ P (v_12 says (v_72 orf v_73)::xs)) ∧
 (∀ v_12 v_74 v_75 xs. P xs ⇒ P (v_12 says (v_74 impf v_75)::xs)) ∧
 (∀ v_12 v_76 v_77 xs. P xs ⇒ P (v_12 says (v_76 eqf v_77)::xs)) ∧
 (∀ v_12 v_78 v_79 xs. P xs ⇒ P (v_12 says v_78 says v_79::xs)) ∧
 (∀ v_12 v_80 v_81 xs.
    P xs ⇒ P (v_12 says v_80 speaks_for v_81::xs)) ∧
 (∀ v_12 v_82 v_83 xs.
    P xs ⇒ P (v_12 says v_82 controls v_83::xs)) ∧
 (∀ v_12 v_84 v_85 v_86 xs.
    P xs ⇒ P (v_12 says reps v_84 v_85 v_86::xs)) ∧
 (∀ v_12 v_87 v_88 xs. P xs ⇒ P (v_12 says v_87 domi v_88::xs)) ∧
 (∀ v_12 v_89 v_90 xs. P xs ⇒ P (v_12 says v_89 eqi v_90::xs)) ∧
 (∀ v_12 v_91 v_92 xs. P xs ⇒ P (v_12 says v_91 doms v_92::xs)) ∧
 (∀ v_12 v_93 v_94 xs. P xs ⇒ P (v_12 says v_93 eqs v_94::xs)) ∧
 (∀ v_12 v_95 v_96 xs. P xs ⇒ P (v_12 says v_95 eqn v_96::xs)) ∧
 (∀ v_12 v_97 v_98 xs. P xs ⇒ P (v_12 says v_97 lte v_98::xs)) ∧
 (∀ v_12 v_99 v100 xs. P xs ⇒ P (v_12 says v_99 lt v100::xs)) ∧
 (∀ v_14 v_15 xs. P xs ⇒ P (v_14 speaks_for v_15::xs)) ∧
 (∀ v_16 v_17 xs. P xs ⇒ P (v_16 controls v_17::xs)) ∧
 (∀ v_18 v_19 v_20 xs. P xs ⇒ P (reps v_18 v_19 v_20::xs)) ∧
 (∀ v_21 v_22 xs. P xs ⇒ P (v_21 domi v_22::xs)) ∧
 (∀ v_23 v_24 xs. P xs ⇒ P (v_23 eqi v_24::xs)) ∧
 (∀ v_25 v_26 xs. P xs ⇒ P (v_25 doms v_26::xs)) ∧
 (∀ v_27 v_28 xs. P xs ⇒ P (v_27 eqs v_28::xs)) ∧
 (∀ v_29 v_30 xs. P xs ⇒ P (v_29 eqn v_30::xs)) ∧
 (∀ v_31 v_32 xs. P xs ⇒ P (v_31 lte v_32::xs)) ∧
 (∀ v_33 v_34 xs. P xs ⇒ P (v_33 lt v_34::xs)) ⇒
 ∀ v. P v
```

Index