

Contents

1	PlanPBType Theory	3
1.1	Datatypes	3
1.2	Theorems	3
2	ssmPlanPB Theory	6
2.1	Theorems	6
3	PlanPBDef Theory	16
3.1	Definitions	16
3.2	Theorems	17

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 ∧

$\text{report1} \neq \text{opoid} \wedge \text{report1} \neq \text{supervise} \wedge \text{report1} \neq \text{report2} \wedge$
 $\text{report1} \neq \text{complete} \wedge \text{report1} \neq \text{plIncomplete} \wedge$
 $\text{report1} \neq \text{invalidPlCommand} \wedge \text{completePlan} \neq \text{opoid} \wedge$
 $\text{completePlan} \neq \text{supervise} \wedge \text{completePlan} \neq \text{report2} \wedge$
 $\text{completePlan} \neq \text{complete} \wedge \text{completePlan} \neq \text{plIncomplete} \wedge$
 $\text{completePlan} \neq \text{invalidPlCommand} \wedge \text{opoid} \neq \text{supervise} \wedge$
 $\text{opoid} \neq \text{report2} \wedge \text{opoid} \neq \text{complete} \wedge \text{opoid} \neq \text{plIncomplete} \wedge$
 $\text{opoid} \neq \text{invalidPlCommand} \wedge \text{supervise} \neq \text{report2} \wedge$
 $\text{supervise} \neq \text{complete} \wedge \text{supervise} \neq \text{plIncomplete} \wedge$
 $\text{supervise} \neq \text{invalidPlCommand} \wedge \text{report2} \neq \text{complete} \wedge$
 $\text{report2} \neq \text{plIncomplete} \wedge \text{report2} \neq \text{invalidPlCommand} \wedge$
 $\text{complete} \neq \text{plIncomplete} \wedge \text{complete} \neq \text{invalidPlCommand} \wedge$
 $\text{plIncomplete} \neq \text{invalidPlCommand}$

[psgCommand_distinct_clauses]

$\vdash \text{initiateMovement} \neq \text{psgIncomplete} \wedge$
 $\text{initiateMovement} \neq \text{invalidPsgCommand} \wedge$
 $\text{psgIncomplete} \neq \text{invalidPsgCommand}$

[slCommand_distinct_clauses]

$\vdash \forall a' a. \text{PL } a \neq \text{PSG } a'$

[slCommand_one_one]

$\vdash (\forall a a'. (\text{PL } a = \text{PL } a') \iff (a = a')) \wedge$
 $\forall a a'. (\text{PSG } a = \text{PSG } a') \iff (a = a')$

[slOutput_distinct_clauses]

$\vdash \text{PlanPB} \neq \text{ReceiveMission} \wedge \text{PlanPB} \neq \text{Warno} \wedge$
 $\text{PlanPB} \neq \text{TentativePlan} \wedge \text{PlanPB} \neq \text{InitiateMovement} \wedge$
 $\text{PlanPB} \neq \text{Recon} \wedge \text{PlanPB} \neq \text{Report1} \wedge \text{PlanPB} \neq \text{CompletePlan} \wedge$
 $\text{PlanPB} \neq \text{Opoid} \wedge \text{PlanPB} \neq \text{Supervise} \wedge \text{PlanPB} \neq \text{Report2} \wedge$
 $\text{PlanPB} \neq \text{Complete} \wedge \text{PlanPB} \neq \text{unAuthenticated} \wedge$
 $\text{PlanPB} \neq \text{unAuthorized} \wedge \text{ReceiveMission} \neq \text{Warno} \wedge$
 $\text{ReceiveMission} \neq \text{TentativePlan} \wedge$
 $\text{ReceiveMission} \neq \text{InitiateMovement} \wedge \text{ReceiveMission} \neq \text{Recon} \wedge$
 $\text{ReceiveMission} \neq \text{Report1} \wedge \text{ReceiveMission} \neq \text{CompletePlan} \wedge$
 $\text{ReceiveMission} \neq \text{Opoid} \wedge \text{ReceiveMission} \neq \text{Supervise} \wedge$
 $\text{ReceiveMission} \neq \text{Report2} \wedge \text{ReceiveMission} \neq \text{Complete} \wedge$
 $\text{ReceiveMission} \neq \text{unAuthenticated} \wedge$
 $\text{ReceiveMission} \neq \text{unAuthorized} \wedge \text{Warno} \neq \text{TentativePlan} \wedge$
 $\text{Warno} \neq \text{InitiateMovement} \wedge \text{Warno} \neq \text{Recon} \wedge \text{Warno} \neq \text{Report1} \wedge$
 $\text{Warno} \neq \text{CompletePlan} \wedge \text{Warno} \neq \text{Opoid} \wedge \text{Warno} \neq \text{Supervise} \wedge$
 $\text{Warno} \neq \text{Report2} \wedge \text{Warno} \neq \text{Complete} \wedge$
 $\text{Warno} \neq \text{unAuthenticated} \wedge \text{Warno} \neq \text{unAuthorized} \wedge$
 $\text{TentativePlan} \neq \text{InitiateMovement} \wedge \text{TentativePlan} \neq \text{Recon} \wedge$
 $\text{TentativePlan} \neq \text{Report1} \wedge \text{TentativePlan} \neq \text{CompletePlan} \wedge$
 $\text{TentativePlan} \neq \text{Opoid} \wedge \text{TentativePlan} \neq \text{Supervise} \wedge$
 $\text{TentativePlan} \neq \text{Report2} \wedge \text{TentativePlan} \neq \text{Complete} \wedge$

$\text{TentativePlan} \neq \text{unAuthenticated} \wedge$
 $\text{TentativePlan} \neq \text{unAuthorized} \wedge \text{InitiateMovement} \neq \text{Recon} \wedge$
 $\text{InitiateMovement} \neq \text{Report1} \wedge$
 $\text{InitiateMovement} \neq \text{CompletePlan} \wedge \text{InitiateMovement} \neq \text{Opoid} \wedge$
 $\text{InitiateMovement} \neq \text{Supervise} \wedge \text{InitiateMovement} \neq \text{Report2} \wedge$
 $\text{InitiateMovement} \neq \text{Complete} \wedge$
 $\text{InitiateMovement} \neq \text{unAuthenticated} \wedge$
 $\text{InitiateMovement} \neq \text{unAuthorized} \wedge \text{Recon} \neq \text{Report1} \wedge$
 $\text{Recon} \neq \text{CompletePlan} \wedge \text{Recon} \neq \text{Opoid} \wedge \text{Recon} \neq \text{Supervise} \wedge$
 $\text{Recon} \neq \text{Report2} \wedge \text{Recon} \neq \text{Complete} \wedge$
 $\text{Recon} \neq \text{unAuthenticated} \wedge \text{Recon} \neq \text{unAuthorized} \wedge$
 $\text{Report1} \neq \text{CompletePlan} \wedge \text{Report1} \neq \text{Opoid} \wedge$
 $\text{Report1} \neq \text{Supervise} \wedge \text{Report1} \neq \text{Report2} \wedge$
 $\text{Report1} \neq \text{Complete} \wedge \text{Report1} \neq \text{unAuthenticated} \wedge$
 $\text{Report1} \neq \text{unAuthorized} \wedge \text{CompletePlan} \neq \text{Opoid} \wedge$
 $\text{CompletePlan} \neq \text{Supervise} \wedge \text{CompletePlan} \neq \text{Report2} \wedge$
 $\text{CompletePlan} \neq \text{Complete} \wedge \text{CompletePlan} \neq \text{unAuthenticated} \wedge$
 $\text{CompletePlan} \neq \text{unAuthorized} \wedge \text{Opoid} \neq \text{Supervise} \wedge$
 $\text{Opoid} \neq \text{Report2} \wedge \text{Opoid} \neq \text{Complete} \wedge$
 $\text{Opoid} \neq \text{unAuthenticated} \wedge \text{Opoid} \neq \text{unAuthorized} \wedge$
 $\text{Supervise} \neq \text{Report2} \wedge \text{Supervise} \neq \text{Complete} \wedge$
 $\text{Supervise} \neq \text{unAuthenticated} \wedge \text{Supervise} \neq \text{unAuthorized} \wedge$
 $\text{Report2} \neq \text{Complete} \wedge \text{Report2} \neq \text{unAuthenticated} \wedge$
 $\text{Report2} \neq \text{unAuthorized} \wedge \text{Complete} \neq \text{unAuthenticated} \wedge$
 $\text{Complete} \neq \text{unAuthorized} \wedge \text{unAuthenticated} \neq \text{unAuthorized}$

[slRole_distinct_clauses]

$\vdash \text{PlatoonLeader} \neq \text{PlatoonSergeant}$

[slState_distinct_clauses]

$\vdash \text{PLAN_PB} \neq \text{RECEIVE_MISSION} \wedge \text{PLAN_PB} \neq \text{WARNO} \wedge$
 $\text{PLAN_PB} \neq \text{TENTATIVE_PLAN} \wedge \text{PLAN_PB} \neq \text{INITIATE_MOVEMENT} \wedge$
 $\text{PLAN_PB} \neq \text{RECON} \wedge \text{PLAN_PB} \neq \text{REPORT1} \wedge$
 $\text{PLAN_PB} \neq \text{COMPLETE_PLAN} \wedge \text{PLAN_PB} \neq \text{OPOID} \wedge$
 $\text{PLAN_PB} \neq \text{SUPERVISE} \wedge \text{PLAN_PB} \neq \text{REPORT2} \wedge$
 $\text{PLAN_PB} \neq \text{COMPLETE} \wedge \text{RECEIVE_MISSION} \neq \text{WARNO} \wedge$
 $\text{RECEIVE_MISSION} \neq \text{TENTATIVE_PLAN} \wedge$
 $\text{RECEIVE_MISSION} \neq \text{INITIATE_MOVEMENT} \wedge$
 $\text{RECEIVE_MISSION} \neq \text{RECON} \wedge \text{RECEIVE_MISSION} \neq \text{REPORT1} \wedge$
 $\text{RECEIVE_MISSION} \neq \text{COMPLETE_PLAN} \wedge \text{RECEIVE_MISSION} \neq \text{OPOID} \wedge$
 $\text{RECEIVE_MISSION} \neq \text{SUPERVISE} \wedge \text{RECEIVE_MISSION} \neq \text{REPORT2} \wedge$
 $\text{RECEIVE_MISSION} \neq \text{COMPLETE} \wedge \text{WARNO} \neq \text{TENTATIVE_PLAN} \wedge$
 $\text{WARNO} \neq \text{INITIATE_MOVEMENT} \wedge \text{WARNO} \neq \text{RECON} \wedge \text{WARNO} \neq \text{REPORT1} \wedge$
 $\text{WARNO} \neq \text{COMPLETE_PLAN} \wedge \text{WARNO} \neq \text{OPOID} \wedge \text{WARNO} \neq \text{SUPERVISE} \wedge$
 $\text{WARNO} \neq \text{REPORT2} \wedge \text{WARNO} \neq \text{COMPLETE} \wedge$
 $\text{TENTATIVE_PLAN} \neq \text{INITIATE_MOVEMENT} \wedge \text{TENTATIVE_PLAN} \neq \text{RECON} \wedge$
 $\text{TENTATIVE_PLAN} \neq \text{REPORT1} \wedge \text{TENTATIVE_PLAN} \neq \text{COMPLETE_PLAN} \wedge$
 $\text{TENTATIVE_PLAN} \neq \text{OPOID} \wedge \text{TENTATIVE_PLAN} \neq \text{SUPERVISE} \wedge$
 $\text{TENTATIVE_PLAN} \neq \text{REPORT2} \wedge \text{TENTATIVE_PLAN} \neq \text{COMPLETE} \wedge$

$$\begin{aligned}
& \text{INITIATE_MOVEMENT} \neq \text{RECON} \wedge \text{INITIATE_MOVEMENT} \neq \text{REPORT1} \wedge \\
& \text{INITIATE_MOVEMENT} \neq \text{COMPLETE_PLAN} \wedge \\
& \text{INITIATE_MOVEMENT} \neq \text{OPOID} \wedge \text{INITIATE_MOVEMENT} \neq \text{SUPERVISE} \wedge \\
& \text{INITIATE_MOVEMENT} \neq \text{REPORT2} \wedge \text{INITIATE_MOVEMENT} \neq \text{COMPLETE} \wedge \\
& \text{RECON} \neq \text{REPORT1} \wedge \text{RECON} \neq \text{COMPLETE_PLAN} \wedge \text{RECON} \neq \text{OPOID} \wedge \\
& \text{RECON} \neq \text{SUPERVISE} \wedge \text{RECON} \neq \text{REPORT2} \wedge \text{RECON} \neq \text{COMPLETE} \wedge \\
& \text{REPORT1} \neq \text{COMPLETE_PLAN} \wedge \text{REPORT1} \neq \text{OPOID} \wedge \\
& \text{REPORT1} \neq \text{SUPERVISE} \wedge \text{REPORT1} \neq \text{REPORT2} \wedge \\
& \text{REPORT1} \neq \text{COMPLETE} \wedge \text{COMPLETE_PLAN} \neq \text{OPOID} \wedge \\
& \text{COMPLETE_PLAN} \neq \text{SUPERVISE} \wedge \text{COMPLETE_PLAN} \neq \text{REPORT2} \wedge \\
& \text{COMPLETE_PLAN} \neq \text{COMPLETE} \wedge \text{OPOID} \neq \text{SUPERVISE} \wedge \\
& \text{OPOID} \neq \text{REPORT2} \wedge \text{OPOID} \neq \text{COMPLETE} \wedge \text{SUPERVISE} \neq \text{REPORT2} \wedge \\
& \text{SUPERVISE} \neq \text{COMPLETE} \wedge \text{REPORT2} \neq \text{COMPLETE}
\end{aligned}$$

2 ssmPlanPB Theory

Built: 10 June 2018

Parent Theories: PlanPBDef, ssm

2.1 Theorems

[inputOK_def]

$$\begin{aligned}
& \vdash (\text{inputOK} (\text{Name PlatoonLeader says prop } cmd) \iff T) \wedge \\
& (\text{inputOK} (\text{Name PlatoonSergeant says prop } cmd) \iff T) \wedge \\
& (\text{inputOK } TT \iff F) \wedge (\text{inputOK } FF \iff F) \wedge \\
& (\text{inputOK} (\text{prop } v) \iff F) \wedge (\text{inputOK} (\text{notf } v_1) \iff F) \wedge \\
& (\text{inputOK} (v_2 \text{ andf } v_3) \iff F) \wedge (\text{inputOK} (v_4 \text{ orf } v_5) \iff F) \wedge \\
& (\text{inputOK} (v_6 \text{ impf } v_7) \iff F) \wedge (\text{inputOK} (v_8 \text{ eqf } v_9) \iff F) \wedge \\
& (\text{inputOK} (v_{10} \text{ says } TT) \iff F) \wedge (\text{inputOK} (v_{10} \text{ says } FF) \iff F) \wedge \\
& (\text{inputOK} (v_{133} \text{ meet } v_{134} \text{ says prop } v_{66}) \iff F) \wedge \\
& (\text{inputOK} (v_{135} \text{ quoting } v_{136} \text{ says prop } v_{66}) \iff F) \wedge \\
& (\text{inputOK} (v_{10} \text{ says notf } v_{67}) \iff F) \wedge \\
& (\text{inputOK} (v_{10} \text{ says } (v_{68} \text{ andf } v_{69})) \iff F) \wedge \\
& (\text{inputOK} (v_{10} \text{ says } (v_{70} \text{ orf } v_{71})) \iff F) \wedge \\
& (\text{inputOK} (v_{10} \text{ says } (v_{72} \text{ impf } v_{73})) \iff F) \wedge \\
& (\text{inputOK} (v_{10} \text{ says } (v_{74} \text{ eqf } v_{75})) \iff F) \wedge \\
& (\text{inputOK} (v_{10} \text{ says } v_{76} \text{ says } v_{77}) \iff F) \wedge \\
& (\text{inputOK} (v_{10} \text{ says } v_{78} \text{ speaks_for } v_{79}) \iff F) \wedge \\
& (\text{inputOK} (v_{10} \text{ says } v_{80} \text{ controls } v_{81}) \iff F) \wedge \\
& (\text{inputOK} (v_{10} \text{ says reps } v_{82} v_{83} v_{84}) \iff F) \wedge \\
& (\text{inputOK} (v_{10} \text{ says } v_{85} \text{ domi } v_{86}) \iff F) \wedge \\
& (\text{inputOK} (v_{10} \text{ says } v_{87} \text{ eqi } v_{88}) \iff F) \wedge \\
& (\text{inputOK} (v_{10} \text{ says } v_{89} \text{ doms } v_{90}) \iff F) \wedge \\
& (\text{inputOK} (v_{10} \text{ says } v_{91} \text{ eqs } v_{92}) \iff F) \wedge \\
& (\text{inputOK} (v_{10} \text{ says } v_{93} \text{ eqn } v_{94}) \iff F) \wedge \\
& (\text{inputOK} (v_{10} \text{ says } v_{95} \text{ lte } v_{96}) \iff F) \wedge \\
& (\text{inputOK} (v_{10} \text{ says } v_{97} \text{ lt } v_{98}) \iff F) \wedge
\end{aligned}$$

```

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

```

[inputOK_ind]

$\vdash \forall P.$

```

( $\forall$  cmd. P (Name PlatoonLeader says prop cmd))  $\wedge$ 
( $\forall$  cmd. P (Name PlatoonSergeant says prop cmd))  $\wedge$  P TT  $\wedge$ 
P FF  $\wedge$  ( $\forall v.$  P (prop v))  $\wedge$  ( $\forall v_1.$  P (notf v1))  $\wedge$ 
( $\forall v_2 v_3.$  P (v2 andf v3))  $\wedge$  ( $\forall v_4 v_5.$  P (v4 orf v5))  $\wedge$ 
( $\forall v_6 v_7.$  P (v6 impf v7))  $\wedge$  ( $\forall v_8 v_9.$  P (v8 eqf v9))  $\wedge$ 
( $\forall v_{10}.$  P (v10 says TT))  $\wedge$  ( $\forall v_{10}.$  P (v10 says FF))  $\wedge$ 
( $\forall v_{133} v_{134} v_{66}.$  P (v133 meet v134 says prop v66))  $\wedge$ 
( $\forall v_{135} v_{136} v_{66}.$  P (v135 quoting v136 says prop v66))  $\wedge$ 
( $\forall v_{10} v_{67}.$  P (v10 says notf v67))  $\wedge$ 
( $\forall v_{10} v_{68} v_{69}.$  P (v10 says (v68 andf v69)))  $\wedge$ 
( $\forall v_{10} v_{70} v_{71}.$  P (v10 says (v70 orf v71)))  $\wedge$ 
( $\forall v_{10} v_{72} v_{73}.$  P (v10 says (v72 impf v73)))  $\wedge$ 
( $\forall v_{10} v_{74} v_{75}.$  P (v10 says (v74 eqf v75)))  $\wedge$ 
( $\forall v_{10} v_{76} v_{77}.$  P (v10 says v76 says v77))  $\wedge$ 
( $\forall v_{10} v_{78} v_{79}.$  P (v10 says v78 speaks_for v79))  $\wedge$ 
( $\forall v_{10} v_{80} v_{81}.$  P (v10 says v80 controls v81))  $\wedge$ 
( $\forall v_{10} v_{82} v_{83} v_{84}.$  P (v10 says reps v82 v83 v84))  $\wedge$ 
( $\forall v_{10} v_{85} v_{86}.$  P (v10 says v85 domi v86))  $\wedge$ 
( $\forall v_{10} v_{87} v_{88}.$  P (v10 says v87 eqi v88))  $\wedge$ 
( $\forall v_{10} v_{89} v_{90}.$  P (v10 says v89 doms v90))  $\wedge$ 
( $\forall v_{10} v_{91} v_{92}.$  P (v10 says v91 eqs v92))  $\wedge$ 
( $\forall v_{10} v_{93} v_{94}.$  P (v10 says v93 eqn v94))  $\wedge$ 
( $\forall v_{10} v_{95} v_{96}.$  P (v10 says v95 lte v96))  $\wedge$ 
( $\forall v_{10} v_{97} v_{98}.$  P (v10 says v97 lt v98))  $\wedge$ 
( $\forall v_{12} v_{13}.$  P (v12 speaks_for v13))  $\wedge$ 
( $\forall v_{14} v_{15}.$  P (v14 controls v15))  $\wedge$ 
( $\forall v_{16} v_{17} v_{18}.$  P (reps v16 v17 v18))  $\wedge$ 
( $\forall v_{19} v_{20}.$  P (v19 domi v20))  $\wedge$ 
( $\forall v_{21} v_{22}.$  P (v21 eqi v22))  $\wedge$ 
( $\forall v_{23} v_{24}.$  P (v23 doms v24))  $\wedge$ 
( $\forall v_{25} v_{26}.$  P (v25 eqs v26))  $\wedge$  ( $\forall v_{27} v_{28}.$  P (v27 eqn v28))  $\wedge$ 
( $\forall v_{29} v_{30}.$  P (v29 lte v30))  $\wedge$  ( $\forall v_{31} v_{32}.$  P (v31 lt v32))  $\Rightarrow$ 
 $\forall v.$  P v

```

[planPBNS_def]

```

 $\vdash$  (planPBNS WARN0 (exec x) =
  if
    (getRecon x = [SOME (SLc (PL recon))])  $\wedge$ 

```

```

    (getTentativePlan x = [SOME (SLc (PL tentativePlan))]) ∧
    (getReport x = [SOME (SLc (PL report1))]) ∧
    (getInitMove x = [SOME (SLc (PSG initiateMovement))])
  then
    REPORT1
  else WARN0) ∧
(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 WARN0 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 WARN0 (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 WARN0 (exec x) =
  if
    (getRecon x = [SOME (SLc (PL recon))]) ∧
    (getTentativePlan 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))  $\iff$ 
  authenticationTest inputOK
    [Name PlatoonLeader says
      prop (SOME (SLc (PL plCommand)))]  $\wedge$ 
  CFGInterpret (M, Oi, Os)
    (CFG inputOK secContext secContextNull
      ([Name PlatoonLeader says
        prop (SOME (SLc (PL plCommand)))]::ins) s outs)  $\wedge$ 
    (M, Oi, Os) satList
    propCommandList
      [Name PlatoonLeader says
        prop (SOME (SLc (PL plCommand)))]

```

[PlatoonLeader_notWARNO_notreport1_exec_plCommand_justified_thm]

```

 $\vdash s \neq \text{WARNO} \Rightarrow$ 
  plCommand  $\neq$  invalidPlCommand  $\Rightarrow$ 
  plCommand  $\neq$  report1  $\Rightarrow$ 
 $\forall NS$  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))  $\iff$ 
  authenticationTest inputOK
    [Name PlatoonLeader says
      prop (SOME (SLc (PL plCommand)))]  $\wedge$ 
  CFGInterpret (M, Oi, Os)
    (CFG inputOK secContext secContextNull
      ([Name PlatoonLeader says
        prop (SOME (SLc (PL plCommand)))]::ins) s outs)  $\wedge$ 
    (M, Oi, Os) satList [prop (SOME (SLc (PL plCommand)))]

```

[PlatoonLeader_notWARNO_notreport1_exec_plCommand_lemma]

```

 $\vdash s \neq \text{WARNO} \Rightarrow$ 
  plCommand  $\neq$  invalidPlCommand  $\Rightarrow$ 
  plCommand  $\neq$  report1  $\Rightarrow$ 
 $\forall M$  Oi Os.
  CFGInterpret (M, Oi, Os)
    (CFG inputOK secContext secContextNull
      ([Name PlatoonLeader says
        prop (SOME (SLc (PL plCommand)))]::ins) s outs)  $\Rightarrow$ 

```

```

(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]

$\vdash \forall M \text{ } Oi \text{ } Os.$
 CFGInterpret (M, Oi, Os)
 (CFG inputOK secContext secContextNull
 ([Name PlatoonLeader says
 prop (SOME (SLc (PSG psgCommand))))]::ins) $s \text{ } outs$) \Rightarrow
 (M, Oi, Os) sat prop NONE

[PlatoonLeader_WARNO_exec_report1_justified_lemma]

$\vdash \forall NS \text{ } Out \text{ } M \text{ } Oi \text{ } Os.$
 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))  $\iff$ 
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)))]  $\wedge$ 
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) WARNNO outs)  $\wedge$ 
  (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_WARNNO_exec_report1_justified_thm]
 $\vdash \forall NS \text{ Out } M \text{ Oi } Os.$ 
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) WARNNO outs)
  (CFG inputOK secContext secContextNull ins
    (NS WARNNO
      (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))  $\iff$ 
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)))]  $\wedge$ 
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)  $\wedge$ 
(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]

```

 $\vdash \forall M \ Oi \ Os.$ 
  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)  $\Rightarrow$ 
(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, acfFoundation, 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 (PL 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))]) ∧
      (getTentativePlan x = [SOME (SLc (PL tentativePlan))]) ∧
      (getReport x = [SOME (SLc (PL report1))]) ∧
      (getInitMove x = [SOME (SLc (PL 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]

$\vdash \forall x. \text{secContextNull } x = [\text{TT}]$

3.2 Theorems

[getInitMove_def]

```

 $\vdash (\text{getInitMove } [] = [\text{NONE}]) \wedge$ 
 $(\forall xs.$ 
  getInitMove
    (Name PlatoonSergeant says
      prop (SOME (SLc (PSG initiateMovement))))::xs) =
    [SOME (SLc (PSG initiateMovement))]  $\wedge$ 
 $(\forall xs. \text{getInitMove } (\text{TT}::xs) = \text{getInitMove } xs) \wedge$ 
 $(\forall xs. \text{getInitMove } (\text{FF}::xs) = \text{getInitMove } xs) \wedge$ 
 $(\forall xs \ v_2. \text{getInitMove } (\text{prop } v_2::xs) = \text{getInitMove } xs) \wedge$ 
 $(\forall xs \ v_3. \text{getInitMove } (\text{notf } v_3::xs) = \text{getInitMove } xs) \wedge$ 
 $(\forall xs \ v_5 \ v_4. \text{getInitMove } (v_4 \text{ andf } v_5::xs) = \text{getInitMove } xs) \wedge$ 
 $(\forall xs \ v_7 \ v_6. \text{getInitMove } (v_6 \text{ orf } v_7::xs) = \text{getInitMove } xs) \wedge$ 
 $(\forall xs \ v_9 \ v_8. \text{getInitMove } (v_8 \text{ impf } v_9::xs) = \text{getInitMove } xs) \wedge$ 
 $(\forall xs \ v_{11} \ v_{10}. \text{getInitMove } (v_{10} \text{ eqf } v_{11}::xs) = \text{getInitMove } xs) \wedge$ 
 $(\forall xs \ v_{12}. \text{getInitMove } (v_{12} \text{ says TT}::xs) = \text{getInitMove } xs) \wedge$ 
 $(\forall xs \ v_{12}. \text{getInitMove } (v_{12} \text{ says FF}::xs) = \text{getInitMove } xs) \wedge$ 
 $(\forall xs \ v_{134}. \text{getInitMove } (\text{Name } v_{134} \text{ says prop NONE}::xs) =$ 
  getInitMove xs)  $\wedge$ 
 $(\forall xs \ v_{144}. \text{getInitMove}$ 
  (Name PlatoonLeader says prop (SOME v144)::xs) =
  getInitMove xs)  $\wedge$ 
 $(\forall xs \ v_{146}. \text{getInitMove}$ 
  (Name PlatoonSergeant says prop (SOME (ESCc v146))::
    xs) =
  getInitMove xs)  $\wedge$ 
 $(\forall xs \ v_{150}. \text{getInitMove}$ 
  (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) ∧

```

```

(∀ xs v92 v91 v12.
  getInitMove (v12 says v91 doms v92::xs) =
  getInitMove xs) ∧
(∀ xs v94 v93 v12.
  getInitMove (v12 says v93 eqs v94::xs) = getInitMove xs) ∧
(∀ xs v96 v95 v12.
  getInitMove (v12 says v95 eqn v96::xs) = getInitMove xs) ∧
(∀ xs v98 v97 v12.
  getInitMove (v12 says v97 lte v98::xs) = getInitMove xs) ∧
(∀ xs v99 v12 v100.
  getInitMove (v12 says v99 lt v100::xs) = getInitMove xs) ∧
(∀ xs v15 v14.
  getInitMove (v14 speaks_for v15::xs) = getInitMove xs) ∧
(∀ xs v17 v16.
  getInitMove (v16 controls v17::xs) = getInitMove xs) ∧
(∀ xs v20 v19 v18.
  getInitMove (reps v18 v19 v20::xs) = getInitMove xs) ∧
(∀ xs v22 v21.
  getInitMove (v21 domi v22::xs) = getInitMove xs) ∧
(∀ xs v24 v23.
  getInitMove (v23 eqi v24::xs) = getInitMove xs) ∧
(∀ xs v26 v25.
  getInitMove (v25 doms v26::xs) = getInitMove xs) ∧
(∀ xs v28 v27.
  getInitMove (v27 eqs v28::xs) = getInitMove xs) ∧
(∀ xs v30 v29.
  getInitMove (v29 eqn v30::xs) = getInitMove xs) ∧
(∀ xs v32 v31.
  getInitMove (v31 lte v32::xs) = getInitMove xs) ∧
∀ xs v34 v33. getInitMove (v33 lt v34::xs) = getInitMove xs

```

[getInitMove_ind]

```

⊢ ∀ P.
  P [] ∧
  (∀ xs.
    P
      (Name PlatoonSergeant says
        prop (SOME (SLc (PSG initiateMovement)))::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)) ∧
(∀ 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 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)) ∧

```

$$\begin{aligned}
& (\forall v_{23} v_{24} xs. P xs \Rightarrow P (v_{23} \text{ eqi } v_{24} :: xs)) \wedge \\
& (\forall v_{25} v_{26} xs. P xs \Rightarrow P (v_{25} \text{ doms } v_{26} :: xs)) \wedge \\
& (\forall v_{27} v_{28} xs. P xs \Rightarrow P (v_{27} \text{ eqs } v_{28} :: xs)) \wedge \\
& (\forall v_{29} v_{30} xs. P xs \Rightarrow P (v_{29} \text{ eqn } v_{30} :: xs)) \wedge \\
& (\forall v_{31} v_{32} xs. P xs \Rightarrow P (v_{31} \text{ lte } v_{32} :: xs)) \wedge \\
& (\forall v_{33} v_{34} xs. P xs \Rightarrow P (v_{33} \text{ lt } v_{34} :: xs)) \Rightarrow \\
& \forall v. P v
\end{aligned}$$

[getPlCom_def]

$$\begin{aligned}
& \vdash (\text{getPlCom } [] = \text{invalidPlCommand}) \wedge \\
& (\forall xs \text{ cmd}. \\
& \quad \text{getPlCom} \\
& \quad \quad (\text{Name PlatoonLeader says prop (SOME (SLc (PL cmd)))}) :: \\
& \quad \quad \quad xs) = \\
& \quad \text{cmd}) \wedge (\forall xs. \text{getPlCom (TT :: xs)} = \text{getPlCom xs}) \wedge \\
& (\forall xs. \text{getPlCom (FF :: xs)} = \text{getPlCom xs}) \wedge \\
& (\forall xs v_2. \text{getPlCom (prop } v_2 :: xs) = \text{getPlCom xs}) \wedge \\
& (\forall xs v_3. \text{getPlCom (notf } v_3 :: xs) = \text{getPlCom xs}) \wedge \\
& (\forall xs v_5 v_4. \text{getPlCom (v}_4 \text{ andf v}_5 :: xs) = \text{getPlCom xs}) \wedge \\
& (\forall xs v_7 v_6. \text{getPlCom (v}_6 \text{ orf v}_7 :: xs) = \text{getPlCom xs}) \wedge \\
& (\forall xs v_9 v_8. \text{getPlCom (v}_8 \text{ impf v}_9 :: xs) = \text{getPlCom xs}) \wedge \\
& (\forall xs v_{11} v_{10}. \text{getPlCom (v}_{10} \text{ eqf v}_{11} :: xs) = \text{getPlCom xs}) \wedge \\
& (\forall xs v_{12}. \text{getPlCom (v}_{12} \text{ says TT :: xs)} = \text{getPlCom xs}) \wedge \\
& (\forall xs v_{12}. \text{getPlCom (v}_{12} \text{ says FF :: xs)} = \text{getPlCom xs}) \wedge \\
& (\forall xs v_{134}. \\
& \quad \text{getPlCom (Name v}_{134} \text{ says prop NONE :: xs)} = \text{getPlCom xs}) \wedge \\
& (\forall xs v_{146}. \\
& \quad \text{getPlCom} \\
& \quad \quad (\text{Name PlatoonLeader says prop (SOME (ESCc v}_{146}) :: xs) = \\
& \quad \quad \text{getPlCom xs}) \wedge \\
& (\forall xs v_{151}. \\
& \quad \text{getPlCom} \\
& \quad \quad (\text{Name PlatoonLeader says prop (SOME (SLc (PSG v}_{151})) :: \\
& \quad \quad \quad xs) = \\
& \quad \text{getPlCom xs}) \wedge \\
& (\forall xs v_{144}. \\
& \quad \text{getPlCom} \\
& \quad \quad (\text{Name PlatoonSergeant says prop (SOME v}_{144}) :: xs) = \\
& \quad \quad \text{getPlCom xs}) \wedge \\
& (\forall xs v_{68} v_{136} v_{135}. \\
& \quad \text{getPlCom (v}_{135} \text{ meet v}_{136} \text{ says prop v}_{68} :: xs) = \\
& \quad \text{getPlCom xs}) \wedge \\
& (\forall xs v_{68} v_{138} v_{137}. \\
& \quad \text{getPlCom (v}_{137} \text{ quoting v}_{138} \text{ says prop v}_{68} :: xs) = \\
& \quad \text{getPlCom xs}) \wedge \\
& (\forall xs v_{69} v_{12}. \\
& \quad \text{getPlCom (v}_{12} \text{ says notf v}_{69} :: xs) = \text{getPlCom xs}) \wedge \\
& (\forall xs v_{71} v_{70} v_{12}. \\
& \quad \text{getPlCom (v}_{12} \text{ says (v}_{70} \text{ andf v}_{71}) :: xs) = \text{getPlCom xs}) \wedge
\end{aligned}$$

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

[getPlCom_ind]

$$\begin{aligned}
& \vdash \forall P. \\
& \quad P \ [] \wedge \\
& \quad (\forall cmd \ xs. \\
& \quad \quad P \\
& \quad \quad (\text{Name PlatoonLeader says prop (SOME (SLc (PL cmd)))) ::
\end{aligned}$$

$$\begin{aligned}
& 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 \ (\text{prop } v_2::xs)) \wedge \\
& (\forall v_3 \ xs. P \ xs \Rightarrow P \ (\text{notf } v_3::xs)) \wedge \\
& (\forall v_4 \ v_5 \ xs. P \ xs \Rightarrow P \ (v_4 \ \text{andf } v_5::xs)) \wedge \\
& (\forall v_6 \ v_7 \ xs. P \ xs \Rightarrow P \ (v_6 \ \text{orf } v_7::xs)) \wedge \\
& (\forall v_8 \ v_9 \ xs. P \ xs \Rightarrow P \ (v_8 \ \text{impf } v_9::xs)) \wedge \\
& (\forall v_{10} \ v_{11} \ xs. P \ xs \Rightarrow P \ (v_{10} \ \text{eqf } v_{11}::xs)) \wedge \\
& (\forall v_{12} \ xs. P \ xs \Rightarrow P \ (v_{12} \ \text{says } TT::xs)) \wedge \\
& (\forall v_{12} \ xs. P \ xs \Rightarrow P \ (v_{12} \ \text{says } FF::xs)) \wedge \\
& (\forall v_{134} \ xs. P \ xs \Rightarrow P \ (\text{Name } v_{134} \ \text{says prop NONE}::xs)) \wedge \\
& (\forall v_{146} \ xs. \\
& \quad P \ xs \Rightarrow \\
& \quad P \\
& \quad (\text{Name PlatoonLeader says prop (SOME (ESCc } v_{146})):: \\
& \quad \quad xs)) \wedge \\
& (\forall v_{151} \ xs. \\
& \quad P \ xs \Rightarrow \\
& \quad P \\
& \quad (\text{Name PlatoonLeader says} \\
& \quad \quad \text{prop (SOME (SLc (PSG } v_{151}))::xs)) \wedge \\
& (\forall v_{144} \ xs. \\
& \quad P \ xs \Rightarrow \\
& \quad P \ (\text{Name PlatoonSergeant says prop (SOME } v_{144})::xs)) \wedge \\
& (\forall v_{135} \ v_{136} \ v_{68} \ xs. \\
& \quad P \ xs \Rightarrow P \ (v_{135} \ \text{meet } v_{136} \ \text{says prop } v_{68}::xs)) \wedge \\
& (\forall v_{137} \ v_{138} \ v_{68} \ xs. \\
& \quad P \ xs \Rightarrow P \ (v_{137} \ \text{quoting } v_{138} \ \text{says prop } v_{68}::xs)) \wedge \\
& (\forall v_{12} \ v_{69} \ xs. P \ xs \Rightarrow P \ (v_{12} \ \text{says notf } v_{69}::xs)) \wedge \\
& (\forall v_{12} \ v_{70} \ v_{71} \ xs. P \ xs \Rightarrow P \ (v_{12} \ \text{says (} v_{70} \ \text{andf } v_{71})::xs)) \wedge \\
& (\forall v_{12} \ v_{72} \ v_{73} \ xs. P \ xs \Rightarrow P \ (v_{12} \ \text{says (} v_{72} \ \text{orf } v_{73})::xs)) \wedge \\
& (\forall v_{12} \ v_{74} \ v_{75} \ xs. P \ xs \Rightarrow P \ (v_{12} \ \text{says (} v_{74} \ \text{impf } v_{75})::xs)) \wedge \\
& (\forall v_{12} \ v_{76} \ v_{77} \ xs. P \ xs \Rightarrow P \ (v_{12} \ \text{says (} v_{76} \ \text{eqf } v_{77})::xs)) \wedge \\
& (\forall v_{12} \ v_{78} \ v_{79} \ xs. P \ xs \Rightarrow P \ (v_{12} \ \text{says } v_{78} \ \text{says } v_{79}::xs)) \wedge \\
& (\forall v_{12} \ v_{80} \ v_{81} \ xs. \\
& \quad P \ xs \Rightarrow P \ (v_{12} \ \text{says } v_{80} \ \text{speaks_for } v_{81}::xs)) \wedge \\
& (\forall v_{12} \ v_{82} \ v_{83} \ xs. \\
& \quad P \ xs \Rightarrow P \ (v_{12} \ \text{says } v_{82} \ \text{controls } v_{83}::xs)) \wedge \\
& (\forall v_{12} \ v_{84} \ v_{85} \ v_{86} \ xs. \\
& \quad P \ xs \Rightarrow P \ (v_{12} \ \text{says reps } v_{84} \ v_{85} \ v_{86}::xs)) \wedge \\
& (\forall v_{12} \ v_{87} \ v_{88} \ xs. P \ xs \Rightarrow P \ (v_{12} \ \text{says } v_{87} \ \text{domi } v_{88}::xs)) \wedge \\
& (\forall v_{12} \ v_{89} \ v_{90} \ xs. P \ xs \Rightarrow P \ (v_{12} \ \text{says } v_{89} \ \text{eqi } v_{90}::xs)) \wedge \\
& (\forall v_{12} \ v_{91} \ v_{92} \ xs. P \ xs \Rightarrow P \ (v_{12} \ \text{says } v_{91} \ \text{doms } v_{92}::xs)) \wedge \\
& (\forall v_{12} \ v_{93} \ v_{94} \ xs. P \ xs \Rightarrow P \ (v_{12} \ \text{says } v_{93} \ \text{eqs } v_{94}::xs)) \wedge \\
& (\forall v_{12} \ v_{95} \ v_{96} \ xs. P \ xs \Rightarrow P \ (v_{12} \ \text{says } v_{95} \ \text{eqn } v_{96}::xs)) \wedge \\
& (\forall v_{12} \ v_{97} \ v_{98} \ xs. P \ xs \Rightarrow P \ (v_{12} \ \text{says } v_{97} \ \text{lte } v_{98}::xs)) \wedge \\
& (\forall v_{12} \ v_{99} \ v_{100} \ xs. P \ xs \Rightarrow P \ (v_{12} \ \text{says } v_{99} \ \text{lt } v_{100}::xs)) \wedge \\
& (\forall v_{14} \ v_{15} \ xs. P \ xs \Rightarrow P \ (v_{14} \ \text{speaks_for } v_{15}::xs)) \wedge \\
& (\forall v_{16} \ v_{17} \ xs. P \ xs \Rightarrow P \ (v_{16} \ \text{controls } v_{17}::xs)) \wedge
\end{aligned}$$

$$\begin{aligned}
& (\forall v_{18} v_{19} v_{20} xs. P xs \Rightarrow P (\text{reps } v_{18} v_{19} v_{20} :: xs)) \wedge \\
& (\forall v_{21} v_{22} xs. P xs \Rightarrow P (v_{21} \text{ domi } v_{22} :: xs)) \wedge \\
& (\forall v_{23} v_{24} xs. P xs \Rightarrow P (v_{23} \text{ eqi } v_{24} :: xs)) \wedge \\
& (\forall v_{25} v_{26} xs. P xs \Rightarrow P (v_{25} \text{ doms } v_{26} :: xs)) \wedge \\
& (\forall v_{27} v_{28} xs. P xs \Rightarrow P (v_{27} \text{ eqs } v_{28} :: xs)) \wedge \\
& (\forall v_{29} v_{30} xs. P xs \Rightarrow P (v_{29} \text{ eqn } v_{30} :: xs)) \wedge \\
& (\forall v_{31} v_{32} xs. P xs \Rightarrow P (v_{31} \text{ lte } v_{32} :: xs)) \wedge \\
& (\forall v_{33} v_{34} xs. P xs \Rightarrow P (v_{33} \text{ lt } v_{34} :: xs)) \Rightarrow \\
& \forall v. P v
\end{aligned}$$

[getPsgCom_def]

$$\begin{aligned}
& \vdash (\text{getPsgCom } [] = \text{invalidPsgCommand}) \wedge \\
& (\forall xs \text{ cmd}. \\
& \quad \text{getPsgCom} \\
& \quad \quad (\text{Name PlatoonSergeant says prop (SOME (SLc (PSG cmd)))) ::} \\
& \quad \quad \quad xs) = \\
& \quad \quad \text{cmd}) \wedge (\forall xs. \text{getPsgCom (TT :: xs)} = \text{getPsgCom } xs) \wedge \\
& (\forall xs. \text{getPsgCom (FF :: xs)} = \text{getPsgCom } xs) \wedge \\
& (\forall xs \ v_2. \text{getPsgCom (prop } v_2 :: xs) = \text{getPsgCom } xs) \wedge \\
& (\forall xs \ v_3. \text{getPsgCom (notf } v_3 :: xs) = \text{getPsgCom } xs) \wedge \\
& (\forall xs \ v_5 \ v_4. \text{getPsgCom (v}_4 \text{ andf } v_5 :: xs) = \text{getPsgCom } xs) \wedge \\
& (\forall xs \ v_7 \ v_6. \text{getPsgCom (v}_6 \text{ orf } v_7 :: xs) = \text{getPsgCom } xs) \wedge \\
& (\forall xs \ v_9 \ v_8. \text{getPsgCom (v}_8 \text{ impf } v_9 :: xs) = \text{getPsgCom } xs) \wedge \\
& (\forall xs \ v_{11} \ v_{10}. \text{getPsgCom (v}_{10} \text{ eqf } v_{11} :: xs) = \text{getPsgCom } xs) \wedge \\
& (\forall xs \ v_{12}. \text{getPsgCom (v}_{12} \text{ says TT :: xs)} = \text{getPsgCom } xs) \wedge \\
& (\forall xs \ v_{12}. \text{getPsgCom (v}_{12} \text{ says FF :: xs)} = \text{getPsgCom } xs) \wedge \\
& (\forall xs \ v_{134}. \\
& \quad \text{getPsgCom (Name v}_{134} \text{ says prop NONE :: xs)} = \text{getPsgCom } xs) \wedge \\
& (\forall xs \ v_{144}. \\
& \quad \text{getPsgCom (Name PlatoonLeader says prop (SOME v}_{144} :: xs)} = \\
& \quad \text{getPsgCom } xs) \wedge \\
& (\forall xs \ v_{146}. \\
& \quad \text{getPsgCom} \\
& \quad \quad (\text{Name PlatoonSergeant says prop (SOME (ESCc v}_{146} :: xs)) ::} \\
& \quad \quad \quad xs) = \\
& \quad \quad \text{getPsgCom } xs) \wedge \\
& (\forall xs \ v_{150}. \\
& \quad \text{getPsgCom} \\
& \quad \quad (\text{Name PlatoonSergeant says prop (SOME (SLc (PL v}_{150} :: xs)) ::} \\
& \quad \quad \quad xs) = \\
& \quad \quad \text{getPsgCom } xs) \wedge \\
& (\forall xs \ v_{68} \ v_{136} \ v_{135}. \\
& \quad \text{getPsgCom (v}_{135} \text{ meet v}_{136} \text{ says prop v}_{68} :: xs) = \\
& \quad \text{getPsgCom } xs) \wedge \\
& (\forall xs \ v_{68} \ v_{138} \ v_{137}. \\
& \quad \text{getPsgCom (v}_{137} \text{ quoting v}_{138} \text{ says prop v}_{68} :: xs) = \\
& \quad \text{getPsgCom } xs) \wedge \\
& (\forall xs \ v_{69} \ v_{12}. \\
& \quad \text{getPsgCom (v}_{12} \text{ says notf v}_{69} :: xs) = \text{getPsgCom } xs) \wedge
\end{aligned}$$

$$\begin{aligned}
& (\forall xs \ v_{71} \ v_{70} \ v_{12}. \\
& \quad \text{getPsgCom } (v_{12} \text{ says } (v_{70} \text{ andf } v_{71}) :: xs) = \text{getPsgCom } xs) \wedge \\
& (\forall xs \ v_{73} \ v_{72} \ v_{12}. \\
& \quad \text{getPsgCom } (v_{12} \text{ says } (v_{72} \text{ orf } v_{73}) :: xs) = \text{getPsgCom } xs) \wedge \\
& (\forall xs \ v_{75} \ v_{74} \ v_{12}. \\
& \quad \text{getPsgCom } (v_{12} \text{ says } (v_{74} \text{ impf } v_{75}) :: xs) = \text{getPsgCom } xs) \wedge \\
& (\forall xs \ v_{77} \ v_{76} \ v_{12}. \\
& \quad \text{getPsgCom } (v_{12} \text{ says } (v_{76} \text{ eqf } v_{77}) :: xs) = \text{getPsgCom } xs) \wedge \\
& (\forall xs \ v_{79} \ v_{78} \ v_{12}. \\
& \quad \text{getPsgCom } (v_{12} \text{ says } v_{78} \text{ says } v_{79} :: xs) = \text{getPsgCom } xs) \wedge \\
& (\forall xs \ v_{81} \ v_{80} \ v_{12}. \\
& \quad \text{getPsgCom } (v_{12} \text{ says } v_{80} \text{ speaks_for } v_{81} :: xs) = \\
& \quad \text{getPsgCom } xs) \wedge \\
& (\forall xs \ v_{83} \ v_{82} \ v_{12}. \\
& \quad \text{getPsgCom } (v_{12} \text{ says } v_{82} \text{ controls } v_{83} :: xs) = \\
& \quad \text{getPsgCom } xs) \wedge \\
& (\forall xs \ v_{86} \ v_{85} \ v_{84} \ v_{12}. \\
& \quad \text{getPsgCom } (v_{12} \text{ says reps } v_{84} \ v_{85} \ v_{86} :: xs) = \\
& \quad \text{getPsgCom } xs) \wedge \\
& (\forall xs \ v_{88} \ v_{87} \ v_{12}. \\
& \quad \text{getPsgCom } (v_{12} \text{ says } v_{87} \text{ domi } v_{88} :: xs) = \text{getPsgCom } xs) \wedge \\
& (\forall xs \ v_{90} \ v_{89} \ v_{12}. \\
& \quad \text{getPsgCom } (v_{12} \text{ says } v_{89} \text{ eqi } v_{90} :: xs) = \text{getPsgCom } xs) \wedge \\
& (\forall xs \ v_{92} \ v_{91} \ v_{12}. \\
& \quad \text{getPsgCom } (v_{12} \text{ says } v_{91} \text{ doms } v_{92} :: xs) = \text{getPsgCom } xs) \wedge \\
& (\forall xs \ v_{94} \ v_{93} \ v_{12}. \\
& \quad \text{getPsgCom } (v_{12} \text{ says } v_{93} \text{ eqs } v_{94} :: xs) = \text{getPsgCom } xs) \wedge \\
& (\forall xs \ v_{96} \ v_{95} \ v_{12}. \\
& \quad \text{getPsgCom } (v_{12} \text{ says } v_{95} \text{ eqn } v_{96} :: xs) = \text{getPsgCom } xs) \wedge \\
& (\forall xs \ v_{98} \ v_{97} \ v_{12}. \\
& \quad \text{getPsgCom } (v_{12} \text{ says } v_{97} \text{ lte } v_{98} :: xs) = \text{getPsgCom } xs) \wedge \\
& (\forall xs \ v_{99} \ v_{12} \ v_{100}. \\
& \quad \text{getPsgCom } (v_{12} \text{ says } v_{99} \text{ lt } v_{100} :: xs) = \text{getPsgCom } xs) \wedge \\
& (\forall xs \ v_{15} \ v_{14}. \\
& \quad \text{getPsgCom } (v_{14} \text{ speaks_for } v_{15} :: xs) = \text{getPsgCom } xs) \wedge \\
& (\forall xs \ v_{17} \ v_{16}. \\
& \quad \text{getPsgCom } (v_{16} \text{ controls } v_{17} :: xs) = \text{getPsgCom } xs) \wedge \\
& (\forall xs \ v_{20} \ v_{19} \ v_{18}. \\
& \quad \text{getPsgCom } (\text{reps } v_{18} \ v_{19} \ v_{20} :: xs) = \text{getPsgCom } xs) \wedge \\
& (\forall xs \ v_{22} \ v_{21}. \text{getPsgCom } (v_{21} \text{ domi } v_{22} :: xs) = \text{getPsgCom } xs) \wedge \\
& (\forall xs \ v_{24} \ v_{23}. \text{getPsgCom } (v_{23} \text{ eqi } v_{24} :: xs) = \text{getPsgCom } xs) \wedge \\
& (\forall xs \ v_{26} \ v_{25}. \text{getPsgCom } (v_{25} \text{ doms } v_{26} :: xs) = \text{getPsgCom } xs) \wedge \\
& (\forall xs \ v_{28} \ v_{27}. \text{getPsgCom } (v_{27} \text{ eqs } v_{28} :: xs) = \text{getPsgCom } xs) \wedge \\
& (\forall xs \ v_{30} \ v_{29}. \text{getPsgCom } (v_{29} \text{ eqn } v_{30} :: xs) = \text{getPsgCom } xs) \wedge \\
& (\forall xs \ v_{32} \ v_{31}. \text{getPsgCom } (v_{31} \text{ lte } v_{32} :: xs) = \text{getPsgCom } xs) \wedge \\
& \forall xs \ v_{34} \ v_{33}. \text{getPsgCom } (v_{33} \text{ lt } v_{34} :: xs) = \text{getPsgCom } xs
\end{aligned}$$

[getPsgCom_ind]

$\vdash \forall P.$

$P \sqcap \wedge$
 $(\forall cmd \ xs.$
 $\quad P$
 $\quad (\text{Name PlatoonSergeant says}$
 $\quad \quad \text{prop (SOME (SLc (PSG cmd)))::xs})) \wedge$
 $(\forall xs. P \ xs \Rightarrow P \ (TT::xs)) \wedge (\forall xs. P \ xs \Rightarrow P \ (FF::xs)) \wedge$
 $(\forall v_2 \ xs. P \ xs \Rightarrow P \ (\text{prop } v_2::xs)) \wedge$
 $(\forall v_3 \ xs. P \ xs \Rightarrow P \ (\text{notf } v_3::xs)) \wedge$
 $(\forall v_4 \ v_5 \ xs. P \ xs \Rightarrow P \ (v_4 \ \text{andf } v_5::xs)) \wedge$
 $(\forall v_6 \ v_7 \ xs. P \ xs \Rightarrow P \ (v_6 \ \text{orf } v_7::xs)) \wedge$
 $(\forall v_8 \ v_9 \ xs. P \ xs \Rightarrow P \ (v_8 \ \text{impf } v_9::xs)) \wedge$
 $(\forall v_{10} \ v_{11} \ xs. P \ xs \Rightarrow P \ (v_{10} \ \text{eqf } v_{11}::xs)) \wedge$
 $(\forall v_{12} \ xs. P \ xs \Rightarrow P \ (v_{12} \ \text{says } TT::xs)) \wedge$
 $(\forall v_{12} \ xs. P \ xs \Rightarrow P \ (v_{12} \ \text{says } FF::xs)) \wedge$
 $(\forall v_{134} \ xs. P \ xs \Rightarrow P \ (\text{Name } v_{134} \ \text{says prop NONE::xs})) \wedge$
 $(\forall v_{144} \ xs.$
 $\quad P \ xs \Rightarrow$
 $\quad P \ (\text{Name PlatoonLeader says prop (SOME } v_{144})::xs)) \wedge$
 $(\forall v_{146} \ xs.$
 $\quad P \ xs \Rightarrow$
 $\quad P$
 $\quad (\text{Name PlatoonSergeant says prop (SOME (ESCC } v_{146}))::$
 $\quad \quad xs)) \wedge$
 $(\forall v_{150} \ xs.$
 $\quad P \ xs \Rightarrow$
 $\quad P$
 $\quad (\text{Name PlatoonSergeant says}$
 $\quad \quad \text{prop (SOME (SLc (PL } v_{150}))::xs)) \wedge$
 $(\forall v_{135} \ v_{136} \ v_{68} \ xs.$
 $\quad P \ xs \Rightarrow P \ (v_{135} \ \text{meet } v_{136} \ \text{says prop } v_{68}::xs)) \wedge$
 $(\forall v_{137} \ v_{138} \ v_{68} \ xs.$
 $\quad P \ xs \Rightarrow P \ (v_{137} \ \text{quoting } v_{138} \ \text{says prop } v_{68}::xs)) \wedge$
 $(\forall v_{12} \ v_{69} \ xs. P \ xs \Rightarrow P \ (v_{12} \ \text{says notf } v_{69}::xs)) \wedge$
 $(\forall v_{12} \ v_{70} \ v_{71} \ xs. P \ xs \Rightarrow P \ (v_{12} \ \text{says } (v_{70} \ \text{andf } v_{71})::xs)) \wedge$
 $(\forall v_{12} \ v_{72} \ v_{73} \ xs. P \ xs \Rightarrow P \ (v_{12} \ \text{says } (v_{72} \ \text{orf } v_{73})::xs)) \wedge$
 $(\forall v_{12} \ v_{74} \ v_{75} \ xs. P \ xs \Rightarrow P \ (v_{12} \ \text{says } (v_{74} \ \text{impf } v_{75})::xs)) \wedge$
 $(\forall v_{12} \ v_{76} \ v_{77} \ xs. P \ xs \Rightarrow P \ (v_{12} \ \text{says } (v_{76} \ \text{eqf } v_{77})::xs)) \wedge$
 $(\forall v_{12} \ v_{78} \ v_{79} \ xs. P \ xs \Rightarrow P \ (v_{12} \ \text{says } v_{78} \ \text{says } v_{79}::xs)) \wedge$
 $(\forall v_{12} \ v_{80} \ v_{81} \ xs.$
 $\quad P \ xs \Rightarrow P \ (v_{12} \ \text{says } v_{80} \ \text{speaks_for } v_{81}::xs)) \wedge$
 $(\forall v_{12} \ v_{82} \ v_{83} \ xs.$
 $\quad P \ xs \Rightarrow P \ (v_{12} \ \text{says } v_{82} \ \text{controls } v_{83}::xs)) \wedge$
 $(\forall v_{12} \ v_{84} \ v_{85} \ v_{86} \ xs.$
 $\quad P \ xs \Rightarrow P \ (v_{12} \ \text{says reps } v_{84} \ v_{85} \ v_{86}::xs)) \wedge$
 $(\forall v_{12} \ v_{87} \ v_{88} \ xs. P \ xs \Rightarrow P \ (v_{12} \ \text{says } v_{87} \ \text{domi } v_{88}::xs)) \wedge$
 $(\forall v_{12} \ v_{89} \ v_{90} \ xs. P \ xs \Rightarrow P \ (v_{12} \ \text{says } v_{89} \ \text{eqi } v_{90}::xs)) \wedge$
 $(\forall v_{12} \ v_{91} \ v_{92} \ xs. P \ xs \Rightarrow P \ (v_{12} \ \text{says } v_{91} \ \text{doms } v_{92}::xs)) \wedge$
 $(\forall v_{12} \ v_{93} \ v_{94} \ xs. P \ xs \Rightarrow P \ (v_{12} \ \text{says } v_{93} \ \text{eqs } v_{94}::xs)) \wedge$
 $(\forall v_{12} \ v_{95} \ v_{96} \ xs. P \ xs \Rightarrow P \ (v_{12} \ \text{says } v_{95} \ \text{eqn } v_{96}::xs)) \wedge$

$$\begin{aligned}
& (\forall v_{12} v_{97} v_{98} xs. P xs \Rightarrow P (v_{12} \text{ says } v_{97} \text{ lte } v_{98} :: xs)) \wedge \\
& (\forall v_{12} v_{99} v_{100} xs. P xs \Rightarrow P (v_{12} \text{ says } v_{99} \text{ lt } v_{100} :: xs)) \wedge \\
& (\forall v_{14} v_{15} xs. P xs \Rightarrow P (v_{14} \text{ speaks_for } v_{15} :: xs)) \wedge \\
& (\forall v_{16} v_{17} xs. P xs \Rightarrow P (v_{16} \text{ controls } v_{17} :: xs)) \wedge \\
& (\forall v_{18} v_{19} v_{20} xs. P xs \Rightarrow P (\text{reps } v_{18} v_{19} v_{20} :: xs)) \wedge \\
& (\forall v_{21} v_{22} xs. P xs \Rightarrow P (v_{21} \text{ domi } v_{22} :: xs)) \wedge \\
& (\forall v_{23} v_{24} xs. P xs \Rightarrow P (v_{23} \text{ eqi } v_{24} :: xs)) \wedge \\
& (\forall v_{25} v_{26} xs. P xs \Rightarrow P (v_{25} \text{ doms } v_{26} :: xs)) \wedge \\
& (\forall v_{27} v_{28} xs. P xs \Rightarrow P (v_{27} \text{ eqs } v_{28} :: xs)) \wedge \\
& (\forall v_{29} v_{30} xs. P xs \Rightarrow P (v_{29} \text{ eqn } v_{30} :: xs)) \wedge \\
& (\forall v_{31} v_{32} xs. P xs \Rightarrow P (v_{31} \text{ lte } v_{32} :: xs)) \wedge \\
& (\forall v_{33} v_{34} xs. P xs \Rightarrow P (v_{33} \text{ lt } v_{34} :: xs)) \Rightarrow \\
& \forall v. P v
\end{aligned}$$

[getRecon_def]

$$\begin{aligned}
& \vdash (\text{getRecon } [] = [\text{NONE}]) \wedge \\
& (\forall xs. \\
& \quad \text{getRecon} \\
& \quad \quad (\text{Name PlatoonLeader says prop (SOME (SLc (PL recon)))) ::} \\
& \quad \quad \quad xs) = \\
& \quad \quad [\text{SOME (SLc (PL recon))}] \wedge \\
& (\forall xs. \text{getRecon } (\text{TT} :: xs) = \text{getRecon } xs) \wedge \\
& (\forall xs. \text{getRecon } (\text{FF} :: xs) = \text{getRecon } xs) \wedge \\
& (\forall xs v_2. \text{getRecon } (\text{prop } v_2 :: xs) = \text{getRecon } xs) \wedge \\
& (\forall xs v_3. \text{getRecon } (\text{notf } v_3 :: xs) = \text{getRecon } xs) \wedge \\
& (\forall xs v_5 v_4. \text{getRecon } (v_4 \text{ andf } v_5 :: xs) = \text{getRecon } xs) \wedge \\
& (\forall xs v_7 v_6. \text{getRecon } (v_6 \text{ orf } v_7 :: xs) = \text{getRecon } xs) \wedge \\
& (\forall xs v_9 v_8. \text{getRecon } (v_8 \text{ impf } v_9 :: xs) = \text{getRecon } xs) \wedge \\
& (\forall xs v_{11} v_{10}. \text{getRecon } (v_{10} \text{ eqf } v_{11} :: xs) = \text{getRecon } xs) \wedge \\
& (\forall xs v_{12}. \text{getRecon } (v_{12} \text{ says TT} :: xs) = \text{getRecon } xs) \wedge \\
& (\forall xs v_{12}. \text{getRecon } (v_{12} \text{ says FF} :: xs) = \text{getRecon } xs) \wedge \\
& (\forall xs v_{134}. \\
& \quad \text{getRecon } (\text{Name } v_{134} \text{ says prop NONE} :: xs) = \text{getRecon } xs) \wedge \\
& (\forall xs v_{146}. \\
& \quad \text{getRecon} \\
& \quad \quad (\text{Name PlatoonLeader says prop (SOME (ESCc } v_{146})) :: xs) = \\
& \quad \quad \text{getRecon } xs) \wedge \\
& (\forall xs. \\
& \quad \text{getRecon} \\
& \quad \quad (\text{Name PlatoonLeader says} \\
& \quad \quad \quad \text{prop (SOME (SLc (PL receiveMission)))) :: xs} = \\
& \quad \quad \text{getRecon } xs) \wedge \\
& (\forall xs. \\
& \quad \text{getRecon} \\
& \quad \quad (\text{Name PlatoonLeader says prop (SOME (SLc (PL warno)))) ::} \\
& \quad \quad \quad xs) = \\
& \quad \quad \text{getRecon } xs) \wedge \\
& (\forall xs. \\
& \quad \text{getRecon}
\end{aligned}$$

```

      (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 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) ∧

```

$$\begin{aligned}
& (\forall xs \ v_{24} \ v_{23}. \text{getRecon } (v_{23} \text{ eqi } v_{24}::xs) = \text{getRecon } xs) \wedge \\
& (\forall xs \ v_{26} \ v_{25}. \text{getRecon } (v_{25} \text{ doms } v_{26}::xs) = \text{getRecon } xs) \wedge \\
& (\forall xs \ v_{28} \ v_{27}. \text{getRecon } (v_{27} \text{ eqs } v_{28}::xs) = \text{getRecon } xs) \wedge \\
& (\forall xs \ v_{30} \ v_{29}. \text{getRecon } (v_{29} \text{ eqn } v_{30}::xs) = \text{getRecon } xs) \wedge \\
& (\forall xs \ v_{32} \ v_{31}. \text{getRecon } (v_{31} \text{ lte } v_{32}::xs) = \text{getRecon } xs) \wedge \\
& \forall xs \ v_{34} \ v_{33}. \text{getRecon } (v_{33} \text{ lt } v_{34}::xs) = \text{getRecon } xs
\end{aligned}$$

[getRecon_ind]

$$\begin{aligned}
& \vdash \forall P. \\
& \quad P \ [] \ \wedge \\
& \quad (\forall xs. \\
& \quad \quad P \\
& \quad \quad \quad (\text{Name PlatoonLeader says} \\
& \quad \quad \quad \quad \text{prop (SOME (SLc (PL recon)))::xs}) \wedge \\
& \quad \quad (\forall xs. P \ xs \Rightarrow P \ (\text{TT}::xs)) \wedge (\forall xs. P \ xs \Rightarrow P \ (\text{FF}::xs)) \wedge \\
& \quad \quad (\forall v_2 \ xs. P \ xs \Rightarrow P \ (\text{prop } v_2::xs)) \wedge \\
& \quad \quad (\forall v_3 \ xs. P \ xs \Rightarrow P \ (\text{notf } v_3::xs)) \wedge \\
& \quad \quad (\forall v_4 \ v_5 \ xs. P \ xs \Rightarrow P \ (v_4 \ \text{andf } v_5::xs)) \wedge \\
& \quad \quad (\forall v_6 \ v_7 \ xs. P \ xs \Rightarrow P \ (v_6 \ \text{orf } v_7::xs)) \wedge \\
& \quad \quad (\forall v_8 \ v_9 \ xs. P \ xs \Rightarrow P \ (v_8 \ \text{impf } v_9::xs)) \wedge \\
& \quad \quad (\forall v_{10} \ v_{11} \ xs. P \ xs \Rightarrow P \ (v_{10} \ \text{eqf } v_{11}::xs)) \wedge \\
& \quad \quad (\forall v_{12} \ xs. P \ xs \Rightarrow P \ (v_{12} \ \text{says TT}::xs)) \wedge \\
& \quad \quad (\forall v_{12} \ xs. P \ xs \Rightarrow P \ (v_{12} \ \text{says FF}::xs)) \wedge \\
& \quad \quad (\forall v_{134} \ xs. P \ xs \Rightarrow P \ (\text{Name } v_{134} \ \text{says prop NONE}::xs)) \wedge \\
& \quad \quad (\forall v_{146} \ xs. \\
& \quad \quad \quad P \ xs \Rightarrow \\
& \quad \quad \quad P \\
& \quad \quad \quad \quad (\text{Name PlatoonLeader says prop (SOME (ESCc } v_{146})):: \\
& \quad \quad \quad \quad \quad xs)) \wedge \\
& \quad \quad (\forall xs. \\
& \quad \quad \quad P \ xs \Rightarrow \\
& \quad \quad \quad P \\
& \quad \quad \quad \quad (\text{Name PlatoonLeader says} \\
& \quad \quad \quad \quad \quad \text{prop (SOME (SLc (PL receiveMission)))::xs}) \wedge \\
& \quad \quad (\forall xs. \\
& \quad \quad \quad P \ xs \Rightarrow \\
& \quad \quad \quad P \\
& \quad \quad \quad \quad (\text{Name PlatoonLeader says} \\
& \quad \quad \quad \quad \quad \text{prop (SOME (SLc (PL warno)))::xs}) \wedge \\
& \quad \quad (\forall xs. \\
& \quad \quad \quad P \ xs \Rightarrow \\
& \quad \quad \quad P \\
& \quad \quad \quad \quad (\text{Name PlatoonLeader says} \\
& \quad \quad \quad \quad \quad \text{prop (SOME (SLc (PL tentativePlan)))::xs}) \wedge \\
& \quad \quad (\forall xs. \\
& \quad \quad \quad P \ xs \Rightarrow \\
& \quad \quad \quad P \\
& \quad \quad \quad \quad (\text{Name PlatoonLeader says} \\
& \quad \quad \quad \quad \quad \text{prop (SOME (SLc (PL report1)))::xs}) \wedge
\end{aligned}$$

$(\forall xs.$
 $\quad P \quad xs \Rightarrow$
 $\quad P$
 $\quad (\text{Name PlatoonLeader says}$
 $\quad \text{prop (SOME (SLc (PL completePlan))))::xs})) \wedge$
 $(\forall xs.$
 $\quad P \quad xs \Rightarrow$
 $\quad P$
 $\quad (\text{Name PlatoonLeader says}$
 $\quad \text{prop (SOME (SLc (PL opoid))))::xs})) \wedge$
 $(\forall xs.$
 $\quad P \quad xs \Rightarrow$
 $\quad P$
 $\quad (\text{Name PlatoonLeader says}$
 $\quad \text{prop (SOME (SLc (PL supervise))))::xs})) \wedge$
 $(\forall xs.$
 $\quad P \quad xs \Rightarrow$
 $\quad P$
 $\quad (\text{Name PlatoonLeader says}$
 $\quad \text{prop (SOME (SLc (PL report2))))::xs})) \wedge$
 $(\forall xs.$
 $\quad P \quad xs \Rightarrow$
 $\quad P$
 $\quad (\text{Name PlatoonLeader says}$
 $\quad \text{prop (SOME (SLc (PL complete))))::xs})) \wedge$
 $(\forall xs.$
 $\quad P \quad xs \Rightarrow$
 $\quad P$
 $\quad (\text{Name PlatoonLeader says}$
 $\quad \text{prop (SOME (SLc (PL plIncomplete))))::xs})) \wedge$
 $(\forall v151 \quad xs.$
 $\quad P \quad xs \Rightarrow$
 $\quad P$
 $\quad (\text{Name PlatoonLeader says}$
 $\quad \text{prop (SOME (SLc (PSG v151))))::xs})) \wedge$
 $(\forall v144 \quad xs.$
 $\quad P \quad xs \Rightarrow$
 $\quad P \quad (\text{Name PlatoonSergeant says prop (SOME v144)::xs})) \wedge$
 $(\forall v135 \quad v136 \quad v68 \quad xs.$
 $\quad P \quad xs \Rightarrow P \quad (v135 \text{ meet } v136 \text{ says prop } v68::xs)) \wedge$
 $(\forall v137 \quad v138 \quad v68 \quad xs.$
 $\quad P \quad xs \Rightarrow P \quad (v137 \text{ quoting } v138 \text{ says prop } v68::xs)) \wedge$
 $(\forall v_{12} \quad v_{69} \quad xs. \quad P \quad xs \Rightarrow P \quad (v_{12} \text{ says notif } v_{69}::xs)) \wedge$
 $(\forall v_{12} \quad v_{70} \quad v_{71} \quad xs. \quad P \quad xs \Rightarrow P \quad (v_{12} \text{ says } (v_{70} \text{ andf } v_{71})::xs)) \wedge$

$$\begin{aligned}
& (\forall v_{12} v_{72} v_{73} xs. P xs \Rightarrow P (v_{12} \text{ says } (v_{72} \text{ orf } v_{73})::xs)) \wedge \\
& (\forall v_{12} v_{74} v_{75} xs. P xs \Rightarrow P (v_{12} \text{ says } (v_{74} \text{ impf } v_{75})::xs)) \wedge \\
& (\forall v_{12} v_{76} v_{77} xs. P xs \Rightarrow P (v_{12} \text{ says } (v_{76} \text{ eqf } v_{77})::xs)) \wedge \\
& (\forall v_{12} v_{78} v_{79} xs. P xs \Rightarrow P (v_{12} \text{ says } v_{78} \text{ says } v_{79}::xs)) \wedge \\
& (\forall v_{12} v_{80} v_{81} xs. \\
& \quad P xs \Rightarrow P (v_{12} \text{ says } v_{80} \text{ speaks_for } v_{81}::xs)) \wedge \\
& (\forall v_{12} v_{82} v_{83} xs. \\
& \quad P xs \Rightarrow P (v_{12} \text{ says } v_{82} \text{ controls } v_{83}::xs)) \wedge \\
& (\forall v_{12} v_{84} v_{85} v_{86} xs. \\
& \quad P xs \Rightarrow P (v_{12} \text{ says reps } v_{84} v_{85} v_{86}::xs)) \wedge \\
& (\forall v_{12} v_{87} v_{88} xs. P xs \Rightarrow P (v_{12} \text{ says } v_{87} \text{ domi } v_{88}::xs)) \wedge \\
& (\forall v_{12} v_{89} v_{90} xs. P xs \Rightarrow P (v_{12} \text{ says } v_{89} \text{ eqi } v_{90}::xs)) \wedge \\
& (\forall v_{12} v_{91} v_{92} xs. P xs \Rightarrow P (v_{12} \text{ says } v_{91} \text{ doms } v_{92}::xs)) \wedge \\
& (\forall v_{12} v_{93} v_{94} xs. P xs \Rightarrow P (v_{12} \text{ says } v_{93} \text{ eqs } v_{94}::xs)) \wedge \\
& (\forall v_{12} v_{95} v_{96} xs. P xs \Rightarrow P (v_{12} \text{ says } v_{95} \text{ eqn } v_{96}::xs)) \wedge \\
& (\forall v_{12} v_{97} v_{98} xs. P xs \Rightarrow P (v_{12} \text{ says } v_{97} \text{ lte } v_{98}::xs)) \wedge \\
& (\forall v_{12} v_{99} v_{100} xs. P xs \Rightarrow P (v_{12} \text{ says } v_{99} \text{ lt } v_{100}::xs)) \wedge \\
& (\forall v_{14} v_{15} xs. P xs \Rightarrow P (v_{14} \text{ speaks_for } v_{15}::xs)) \wedge \\
& (\forall v_{16} v_{17} xs. P xs \Rightarrow P (v_{16} \text{ controls } v_{17}::xs)) \wedge \\
& (\forall v_{18} v_{19} v_{20} xs. P xs \Rightarrow P (\text{reps } v_{18} v_{19} v_{20}::xs)) \wedge \\
& (\forall v_{21} v_{22} xs. P xs \Rightarrow P (v_{21} \text{ domi } v_{22}::xs)) \wedge \\
& (\forall v_{23} v_{24} xs. P xs \Rightarrow P (v_{23} \text{ eqi } v_{24}::xs)) \wedge \\
& (\forall v_{25} v_{26} xs. P xs \Rightarrow P (v_{25} \text{ doms } v_{26}::xs)) \wedge \\
& (\forall v_{27} v_{28} xs. P xs \Rightarrow P (v_{27} \text{ eqs } v_{28}::xs)) \wedge \\
& (\forall v_{29} v_{30} xs. P xs \Rightarrow P (v_{29} \text{ eqn } v_{30}::xs)) \wedge \\
& (\forall v_{31} v_{32} xs. P xs \Rightarrow P (v_{31} \text{ lte } v_{32}::xs)) \wedge \\
& (\forall v_{33} v_{34} xs. P xs \Rightarrow P (v_{33} \text{ lt } v_{34}::xs)) \Rightarrow \\
& \forall v. P v
\end{aligned}$$

[getReport_def]

$$\begin{aligned}
& \vdash (\text{getReport } [] = [\text{NONE}]) \wedge \\
& (\forall xs. \\
& \quad \text{getReport} \\
& \quad \quad (\text{Name PlatoonLeader says} \\
& \quad \quad \quad \text{prop (SOME (SLc (PL report1)))::xs} = \\
& \quad \quad \quad [\text{SOME (SLc (PL report1))}]) \wedge \\
& (\forall xs. \text{getReport } (\text{TT}::xs) = \text{getReport } xs) \wedge \\
& (\forall xs. \text{getReport } (\text{FF}::xs) = \text{getReport } xs) \wedge \\
& (\forall xs v_2. \text{getReport } (\text{prop } v_2::xs) = \text{getReport } xs) \wedge \\
& (\forall xs v_3. \text{getReport } (\text{notf } v_3::xs) = \text{getReport } xs) \wedge \\
& (\forall xs v_5 v_4. \text{getReport } (v_4 \text{ andf } v_5::xs) = \text{getReport } xs) \wedge \\
& (\forall xs v_7 v_6. \text{getReport } (v_6 \text{ orf } v_7::xs) = \text{getReport } xs) \wedge \\
& (\forall xs v_9 v_8. \text{getReport } (v_8 \text{ impf } v_9::xs) = \text{getReport } xs) \wedge \\
& (\forall xs v_{11} v_{10}. \text{getReport } (v_{10} \text{ eqf } v_{11}::xs) = \text{getReport } xs) \wedge \\
& (\forall xs v_{12}. \text{getReport } (v_{12} \text{ says TT}::xs) = \text{getReport } xs) \wedge \\
& (\forall xs v_{12}. \text{getReport } (v_{12} \text{ says FF}::xs) = \text{getReport } xs) \wedge \\
& (\forall xs v_{134}. \\
& \quad \text{getReport (Name } v_{134} \text{ says prop NONE}::xs) = \text{getReport } xs) \wedge \\
& (\forall xs v_{146}.
\end{aligned}$$


```

    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

```

$$\begin{aligned}
& (\text{Name PlatoonLeader says} \\
& \quad \text{prop (SOME (SLc (PL receiveMission)))::xs})) \wedge \\
(\forall xs. & \\
& \quad P \ xs \Rightarrow \\
& \quad P \\
& \quad (\text{Name PlatoonLeader says} \\
& \quad \quad \text{prop (SOME (SLc (PL warno)))::xs})) \wedge \\
(\forall xs. & \\
& \quad P \ xs \Rightarrow \\
& \quad P \\
& \quad (\text{Name PlatoonLeader says} \\
& \quad \quad \text{prop (SOME (SLc (PL tentativePlan)))::xs})) \wedge \\
(\forall xs. & \\
& \quad P \ xs \Rightarrow \\
& \quad P \\
& \quad (\text{Name PlatoonLeader says} \\
& \quad \quad \text{prop (SOME (SLc (PL recon)))::xs})) \wedge \\
(\forall xs. & \\
& \quad P \ xs \Rightarrow \\
& \quad P \\
& \quad (\text{Name PlatoonLeader says} \\
& \quad \quad \text{prop (SOME (SLc (PL completePlan)))::xs})) \wedge \\
(\forall xs. & \\
& \quad P \ xs \Rightarrow \\
& \quad P \\
& \quad (\text{Name PlatoonLeader says} \\
& \quad \quad \text{prop (SOME (SLc (PL opoid)))::xs})) \wedge \\
(\forall xs. & \\
& \quad P \ xs \Rightarrow \\
& \quad P \\
& \quad (\text{Name PlatoonLeader says} \\
& \quad \quad \text{prop (SOME (SLc (PL supervise)))::xs})) \wedge \\
(\forall xs. & \\
& \quad P \ xs \Rightarrow \\
& \quad P \\
& \quad (\text{Name PlatoonLeader says} \\
& \quad \quad \text{prop (SOME (SLc (PL report2)))::xs})) \wedge \\
(\forall xs. & \\
& \quad P \ xs \Rightarrow \\
& \quad P \\
& \quad (\text{Name PlatoonLeader says} \\
& \quad \quad \text{prop (SOME (SLc (PL complete)))::xs})) \wedge \\
(\forall xs. & \\
& \quad P \ xs \Rightarrow \\
& \quad P \\
& \quad (\text{Name PlatoonLeader says} \\
& \quad \quad \text{prop (SOME (SLc (PL plIncomplete)))::xs})) \wedge \\
(\forall xs. & \\
& \quad P \ xs \Rightarrow
\end{aligned}$$

$$\begin{aligned}
& P \\
& \quad (\text{Name PlatoonLeader says} \\
& \quad \quad \text{prop (SOME (SLc (PL invalidPlCommand)))::xs})) \wedge \\
& (\forall v151 \text{ xs.} \\
& \quad P \text{ xs} \Rightarrow \\
& \quad P \\
& \quad \quad (\text{Name PlatoonLeader says} \\
& \quad \quad \quad \text{prop (SOME (SLc (PSG v151)))::xs})) \wedge \\
& (\forall v144 \text{ xs.} \\
& \quad P \text{ xs} \Rightarrow \\
& \quad P (\text{Name PlatoonSergeant says prop (SOME v144)::xs})) \wedge \\
& (\forall v135 \text{ v136 v68 xs.} \\
& \quad P \text{ xs} \Rightarrow P (v135 \text{ meet v136 says prop v68::xs})) \wedge \\
& (\forall v137 \text{ v138 v68 xs.} \\
& \quad P \text{ xs} \Rightarrow P (v137 \text{ quoting v138 says prop v68::xs})) \wedge \\
& (\forall v12 \text{ v69 xs. } P \text{ xs} \Rightarrow P (v12 \text{ says notf v69::xs})) \wedge \\
& (\forall v12 \text{ v70 v71 xs. } P \text{ xs} \Rightarrow P (v12 \text{ says (v70 andf v71)::xs})) \wedge \\
& (\forall v12 \text{ v72 v73 xs. } P \text{ xs} \Rightarrow P (v12 \text{ says (v72 orf v73)::xs})) \wedge \\
& (\forall v12 \text{ v74 v75 xs. } P \text{ xs} \Rightarrow P (v12 \text{ says (v74 impf v75)::xs})) \wedge \\
& (\forall v12 \text{ v76 v77 xs. } P \text{ xs} \Rightarrow P (v12 \text{ says (v76 eqf v77)::xs})) \wedge \\
& (\forall v12 \text{ v78 v79 xs. } P \text{ xs} \Rightarrow P (v12 \text{ says v78 says v79::xs})) \wedge \\
& (\forall v12 \text{ v80 v81 xs.} \\
& \quad P \text{ xs} \Rightarrow P (v12 \text{ says v80 speaks_for v81::xs})) \wedge \\
& (\forall v12 \text{ v82 v83 xs.} \\
& \quad P \text{ xs} \Rightarrow P (v12 \text{ says v82 controls v83::xs})) \wedge \\
& (\forall v12 \text{ v84 v85 v86 xs.} \\
& \quad P \text{ xs} \Rightarrow P (v12 \text{ says reps v84 v85 v86::xs})) \wedge \\
& (\forall v12 \text{ v87 v88 xs. } P \text{ xs} \Rightarrow P (v12 \text{ says v87 domi v88::xs})) \wedge \\
& (\forall v12 \text{ v89 v90 xs. } P \text{ xs} \Rightarrow P (v12 \text{ says v89 eqi v90::xs})) \wedge \\
& (\forall v12 \text{ v91 v92 xs. } P \text{ xs} \Rightarrow P (v12 \text{ says v91 doms v92::xs})) \wedge \\
& (\forall v12 \text{ v93 v94 xs. } P \text{ xs} \Rightarrow P (v12 \text{ says v93 eqs v94::xs})) \wedge \\
& (\forall v12 \text{ v95 v96 xs. } P \text{ xs} \Rightarrow P (v12 \text{ says v95 eqn v96::xs})) \wedge \\
& (\forall v12 \text{ v97 v98 xs. } P \text{ xs} \Rightarrow P (v12 \text{ says v97 lte v98::xs})) \wedge \\
& (\forall v12 \text{ v99 v100 xs. } P \text{ xs} \Rightarrow P (v12 \text{ says v99 lt v100::xs})) \wedge \\
& (\forall v14 \text{ v15 xs. } P \text{ xs} \Rightarrow P (v14 \text{ speaks_for v15::xs})) \wedge \\
& (\forall v16 \text{ v17 xs. } P \text{ xs} \Rightarrow P (v16 \text{ controls v17::xs})) \wedge \\
& (\forall v18 \text{ v19 v20 xs. } P \text{ xs} \Rightarrow P (\text{reps v18 v19 v20::xs})) \wedge \\
& (\forall v21 \text{ v22 xs. } P \text{ xs} \Rightarrow P (v21 \text{ domi v22::xs})) \wedge \\
& (\forall v23 \text{ v24 xs. } P \text{ xs} \Rightarrow P (v23 \text{ eqi v24::xs})) \wedge \\
& (\forall v25 \text{ v26 xs. } P \text{ xs} \Rightarrow P (v25 \text{ doms v26::xs})) \wedge \\
& (\forall v27 \text{ v28 xs. } P \text{ xs} \Rightarrow P (v27 \text{ eqs v28::xs})) \wedge \\
& (\forall v29 \text{ v30 xs. } P \text{ xs} \Rightarrow P (v29 \text{ eqn v30::xs})) \wedge \\
& (\forall v31 \text{ v32 xs. } P \text{ xs} \Rightarrow P (v31 \text{ lte v32::xs})) \wedge \\
& (\forall v33 \text{ v34 xs. } P \text{ xs} \Rightarrow P (v33 \text{ lt v34::xs})) \Rightarrow \\
& \forall v. P v
\end{aligned}$$

[getTenativePlan_def]

$$\begin{aligned}
& \vdash (\text{getTenativePlan []} = [\text{NONE}]) \wedge \\
& (\forall \text{xs.}
\end{aligned}$$

```

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) =

```

```

    getTentativePlan xs) ∧
(∀ xs v69 v12.
  getTentativePlan (v12 says notf v69::xs) =
  getTentativePlan xs) ∧
(∀ xs v71 v70 v12.
  getTentativePlan (v12 says (v70 andf v71)::xs) =
  getTentativePlan xs) ∧
(∀ xs v73 v72 v12.
  getTentativePlan (v12 says (v72 orf v73)::xs) =
  getTentativePlan xs) ∧
(∀ xs v75 v74 v12.
  getTentativePlan (v12 says (v74 impf v75)::xs) =
  getTentativePlan xs) ∧
(∀ xs v77 v76 v12.
  getTentativePlan (v12 says (v76 eqf v77)::xs) =
  getTentativePlan xs) ∧
(∀ xs v79 v78 v12.
  getTentativePlan (v12 says v78 says v79::xs) =
  getTentativePlan xs) ∧
(∀ xs v81 v80 v12.
  getTentativePlan (v12 says v80 speaks_for v81::xs) =
  getTentativePlan xs) ∧
(∀ xs v83 v82 v12.
  getTentativePlan (v12 says v82 controls v83::xs) =
  getTentativePlan xs) ∧
(∀ xs v86 v85 v84 v12.
  getTentativePlan (v12 says reps v84 v85 v86::xs) =
  getTentativePlan xs) ∧
(∀ xs v88 v87 v12.
  getTentativePlan (v12 says v87 domi v88::xs) =
  getTentativePlan xs) ∧
(∀ xs v90 v89 v12.
  getTentativePlan (v12 says v89 eqi v90::xs) =
  getTentativePlan xs) ∧
(∀ xs v92 v91 v12.
  getTentativePlan (v12 says v91 doms v92::xs) =
  getTentativePlan xs) ∧
(∀ xs v94 v93 v12.
  getTentativePlan (v12 says v93 eqs v94::xs) =
  getTentativePlan xs) ∧
(∀ xs v96 v95 v12.
  getTentativePlan (v12 says v95 eqn v96::xs) =
  getTentativePlan xs) ∧
(∀ xs v98 v97 v12.
  getTentativePlan (v12 says v97 lte v98::xs) =
  getTentativePlan xs) ∧
(∀ xs v99 v12 v100.
  getTentativePlan (v12 says v99 lt v100::xs) =
  getTentativePlan xs) ∧

```



```

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

```

[getTentativePlan_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

```

$$\begin{aligned}
& (\text{Name PlatoonLeader says} \\
& \quad \text{prop (SOME (SLc (PL receiveMission)))::xs})) \wedge \\
(\forall xs. & \\
& \quad P \ xs \Rightarrow \\
& \quad P \\
& \quad (\text{Name PlatoonLeader says} \\
& \quad \quad \text{prop (SOME (SLc (PL warno)))::xs})) \wedge \\
(\forall xs. & \\
& \quad P \ xs \Rightarrow \\
& \quad P \\
& \quad (\text{Name PlatoonLeader says} \\
& \quad \quad \text{prop (SOME (SLc (PL recon)))::xs})) \wedge \\
(\forall xs. & \\
& \quad P \ xs \Rightarrow \\
& \quad P \\
& \quad (\text{Name PlatoonLeader says} \\
& \quad \quad \text{prop (SOME (SLc (PL report1)))::xs})) \wedge \\
(\forall xs. & \\
& \quad P \ xs \Rightarrow \\
& \quad P \\
& \quad (\text{Name PlatoonLeader says} \\
& \quad \quad \text{prop (SOME (SLc (PL completePlan)))::xs})) \wedge \\
(\forall xs. & \\
& \quad P \ xs \Rightarrow \\
& \quad P \\
& \quad (\text{Name PlatoonLeader says} \\
& \quad \quad \text{prop (SOME (SLc (PL opoid)))::xs})) \wedge \\
(\forall xs. & \\
& \quad P \ xs \Rightarrow \\
& \quad P \\
& \quad (\text{Name PlatoonLeader says} \\
& \quad \quad \text{prop (SOME (SLc (PL supervise)))::xs})) \wedge \\
(\forall xs. & \\
& \quad P \ xs \Rightarrow \\
& \quad P \\
& \quad (\text{Name PlatoonLeader says} \\
& \quad \quad \text{prop (SOME (SLc (PL report2)))::xs})) \wedge \\
(\forall xs. & \\
& \quad P \ xs \Rightarrow \\
& \quad P \\
& \quad (\text{Name PlatoonLeader says} \\
& \quad \quad \text{prop (SOME (SLc (PL complete)))::xs})) \wedge \\
(\forall xs. & \\
& \quad P \ xs \Rightarrow \\
& \quad P \\
& \quad (\text{Name PlatoonLeader says} \\
& \quad \quad \text{prop (SOME (SLc (PL plIncomplete)))::xs})) \wedge \\
(\forall xs. & \\
& \quad P \ xs \Rightarrow
\end{aligned}$$

$$\begin{aligned}
& P \\
& \quad (\text{Name PlatoonLeader says} \\
& \quad \quad \text{prop (SOME (SLc (PL invalidPlCommand)))::xs}) \wedge \\
& (\forall v151 \text{ xs.} \\
& \quad P \text{ xs} \Rightarrow \\
& \quad P \\
& \quad \quad (\text{Name PlatoonLeader says} \\
& \quad \quad \quad \text{prop (SOME (SLc (PSG v151)))::xs}) \wedge \\
& (\forall v144 \text{ xs.} \\
& \quad P \text{ xs} \Rightarrow \\
& \quad P (\text{Name PlatoonSergeant says prop (SOME v144)::xs}) \wedge \\
& (\forall v135 \text{ v136 v68 xs.} \\
& \quad P \text{ xs} \Rightarrow P (v135 \text{ meet v136 says prop v68::xs}) \wedge \\
& (\forall v137 \text{ v138 v68 xs.} \\
& \quad P \text{ xs} \Rightarrow P (v137 \text{ quoting v138 says prop v68::xs}) \wedge \\
& (\forall v12 \text{ v69 xs. } P \text{ xs} \Rightarrow P (v12 \text{ says notf v69::xs}) \wedge \\
& (\forall v12 \text{ v70 v71 xs. } P \text{ xs} \Rightarrow P (v12 \text{ says (v70 andf v71)::xs}) \wedge \\
& (\forall v12 \text{ v72 v73 xs. } P \text{ xs} \Rightarrow P (v12 \text{ says (v72 orf v73)::xs}) \wedge \\
& (\forall v12 \text{ v74 v75 xs. } P \text{ xs} \Rightarrow P (v12 \text{ says (v74 impf v75)::xs}) \wedge \\
& (\forall v12 \text{ v76 v77 xs. } P \text{ xs} \Rightarrow P (v12 \text{ says (v76 eqf v77)::xs}) \wedge \\
& (\forall v12 \text{ v78 v79 xs. } P \text{ xs} \Rightarrow P (v12 \text{ says v78 says v79::xs}) \wedge \\
& (\forall v12 \text{ v80 v81 xs.} \\
& \quad P \text{ xs} \Rightarrow P (v12 \text{ says v80 speaks_for v81::xs}) \wedge \\
& (\forall v12 \text{ v82 v83 xs.} \\
& \quad P \text{ xs} \Rightarrow P (v12 \text{ says v82 controls v83::xs}) \wedge \\
& (\forall v12 \text{ v84 v85 v86 xs.} \\
& \quad P \text{ xs} \Rightarrow P (v12 \text{ says reps v84 v85 v86::xs}) \wedge \\
& (\forall v12 \text{ v87 v88 xs. } P \text{ xs} \Rightarrow P (v12 \text{ says v87 domi v88::xs}) \wedge \\
& (\forall v12 \text{ v89 v90 xs. } P \text{ xs} \Rightarrow P (v12 \text{ says v89 eqi v90::xs}) \wedge \\
& (\forall v12 \text{ v91 v92 xs. } P \text{ xs} \Rightarrow P (v12 \text{ says v91 doms v92::xs}) \wedge \\
& (\forall v12 \text{ v93 v94 xs. } P \text{ xs} \Rightarrow P (v12 \text{ says v93 eqs v94::xs}) \wedge \\
& (\forall v12 \text{ v95 v96 xs. } P \text{ xs} \Rightarrow P (v12 \text{ says v95 eqn v96::xs}) \wedge \\
& (\forall v12 \text{ v97 v98 xs. } P \text{ xs} \Rightarrow P (v12 \text{ says v97 lte v98::xs}) \wedge \\
& (\forall v12 \text{ v99 v100 xs. } P \text{ xs} \Rightarrow P (v12 \text{ says v99 lt v100::xs}) \wedge \\
& (\forall v14 \text{ v15 xs. } P \text{ xs} \Rightarrow P (v14 \text{ speaks_for v15::xs}) \wedge \\
& (\forall v16 \text{ v17 xs. } P \text{ xs} \Rightarrow P (v16 \text{ controls v17::xs}) \wedge \\
& (\forall v18 \text{ v19 v20 xs. } P \text{ xs} \Rightarrow P (\text{reps v18 v19 v20::xs}) \wedge \\
& (\forall v21 \text{ v22 xs. } P \text{ xs} \Rightarrow P (v21 \text{ domi v22::xs}) \wedge \\
& (\forall v23 \text{ v24 xs. } P \text{ xs} \Rightarrow P (v23 \text{ eqi v24::xs}) \wedge \\
& (\forall v25 \text{ v26 xs. } P \text{ xs} \Rightarrow P (v25 \text{ doms v26::xs}) \wedge \\
& (\forall v27 \text{ v28 xs. } P \text{ xs} \Rightarrow P (v27 \text{ eqs v28::xs}) \wedge \\
& (\forall v29 \text{ v30 xs. } P \text{ xs} \Rightarrow P (v29 \text{ eqn v30::xs}) \wedge \\
& (\forall v31 \text{ v32 xs. } P \text{ xs} \Rightarrow P (v31 \text{ lte v32::xs}) \wedge \\
& (\forall v33 \text{ v34 xs. } P \text{ xs} \Rightarrow P (v33 \text{ lt v34::xs}) \Rightarrow \\
& \forall v. P v
\end{aligned}$$

Index

PlanPBDef Theory, 16

Definitions, 16

PL_notWARNO_Auth_def, 16

PL_WARNO_Auth_def, 16

secContext_def, 16

secContextNull_def, 17

Theorems, 17

getInitMove_def, 17

getInitMove_ind, 19

getPlCom_def, 21

getPlCom_ind, 22

getPsgCom_def, 24

getPsgCom_ind, 25

getRecon_def, 27

getRecon_ind, 30

getReport_def, 32

getReport_ind, 35

getTenativePlan_def, 37

getTenativePlan_ind, 41

PlanPBType Theory, 3

Datatypes, 3

Theorems, 3

plCommand_distinct_clauses, 3

psgCommand_distinct_clauses, 4

slCommand_distinct_clauses, 4

slCommand_one_one, 4

slOutput_distinct_clauses, 4

slRole_distinct_clauses, 5

slState_distinct_clauses, 5

ssmPlanPB Theory, 6

Theorems, 6

inputOK_def, 6

inputOK_ind, 7

planPBNS_def, 7

planPBNS_ind, 8

planPBOut_def, 8

planPBOut_ind, 9

PlatoonLeader_notWARNO_notreport1_-
exec_plCommand_justified_lemma, 9

PlatoonLeader_notWARNO_notreport1_-
exec_plCommand_justified_thm, 10

PlatoonLeader_notWARNO_notreport1_-
exec_plCommand_lemma, 10

PlatoonLeader_psgCommand_notDis-
card_thm, 11

PlatoonLeader_trap_psgCommand_jus-
tified_lemma, 11

PlatoonLeader_trap_psgCommand_lemma,
11

PlatoonLeader_WARNO_exec_report1_-
justified_lemma, 12

PlatoonLeader_WARNO_exec_report1_-
justified_thm, 13

PlatoonLeader_WARNO_exec_report1_-
lemma, 14

PlatoonSergeant_trap_plCommand_jus-
tified_lemma, 15

PlatoonSergeant_trap_plCommand_jus-
tified_thm, 15

PlatoonSergeant_trap_plCommand_lemma,
16