

Standard Operating Procedure

Document Number: TCAS-SYS-001

Standard Operating Procedure (SOP): Traffic Collision
Avoidance System (TCAS) Altitude Advisory Control

Revision: 1.0

Last Updated: [DATE]

Contents

1. Introduction
 - 1.1 Purpose
 - 1.2 System Overview
 - 1.3 Regulatory Compliance
2. System Specifications
 - 2.1 Parameters and Advisory Modes
 - 2.2 Altitude Layers and VSL Ranges
3. Operational Protocols
 - 3.1 Initialization and Evaluation Events
 - 3.2 Advisory Generation Workflow
 - 3.3 Inhibit Logic Processing
4. Emergency Operations
 - 4.1 Failover in RA Computation
 - 4.2 Recovery from Sensor Invalidity
5. Maintenance Requirements
 - 5.1 Resetting System Flags
 - 5.2 Reinitialization upon Fault Detection
6. Quality Assurance
 - 6.1 Altitude and VSL Value Validations
 - 6.2 Advisory History Tracking

- 7. Security Protocols
 - 7.1 Advisory Consistency Checks
 - 7.2 Advisory Transition Verification
 - 8. Environmental Considerations
 - 8.1 Radio and Barometric Altimeter Integration
 - 8.2 Conflict-Free Mode Selector Operation
 - 9. Training Requirements
 - 9.1 System Interpretation
 - 9.2 Simulation Scenarios
 - 10. Document Control
 - 10.1 Revision History
 - 10.2 Authorization
 - 11. Process Flows and State Transitions
 - 11.1 Advisory State Transition Logic
 - 11.2 Climb and Descend Inhibit Sequences
 - 11.3 *AutoSL and EffectiveSL* Computation Logic
-

1. Introduction

1.1 Purpose

- To provide consistent, responsive vertical resolution advisories to avoid mid-air collisions.
- Ensure safe vertical maneuvering guidance under various traffic scenarios.

1.2 System Overview

- The system uses multiple altitude sensors (radio, barometric) to determine position.
- Computes advisories like Climb, Descend, Negative, or No Resolution Advisory (RA).
- Adjusts based on configuration modes, sensor validity, and advisory history.

1.3 Regulatory Compliance

- Adheres to aviation safety and traffic separation standards (e.g. RTCA D0-185B).
 - Complies with certification for TCAS II systems.
-

2. System Specifications

2.1 Parameters and Advisory Modes

- **CompositeRA**: *Advisory issued (Climb, Descend, NoRA, Negative)*
- **Corrective_Climb / Descend**: Binary flags indicating corrective action required
- **Vertical_Control**: Indicates Increase, Maintain, Reversal, etc.
- **Combined_Control**: Merged advisory signal for system-wide logic use

2.2 Altitude Layers and VSL Ranges

- Four altitude layers: *Layer1 to Layer4*
 - Vertical Speed Limits (VSL): *No_VSL, VSL0, VSL500, VSL1000, VSL2000*
 - Advisory thresholds vary by altitude and sensor readings
-

3. Operational Protocols

3.1 Initialization and Evaluation Events

- System begins with `Surveillance_Complete_Event = FALSE`
- Initialization sets advisory mode to `No_RA`
- Events (e.g. `CompositeRAEvaluated_Event`) are triggered post-surveillance

3.2 Advisory Generation Workflow

- Calculate altitude, climb, and descend inhibition values
- Determine advisories based on current and historical states
- Set advisory flags (`Corrective_Climb` , `Corrective_Descend`) accordingly

3.3 Inhibit Logic Processing

- `Climb_Inhibit` and `Descend_Inhibit` are dynamically toggled based on:
 - Altitude thresholds
 - Configured settings
 - Discrete inhibit switches
 - These affect the issuance of climb or descend advisories
-

4. Emergency Operations

4.1 Failover in RA Computation

- Revert to last known stable advisory if new computation fails
- Trigger evaluation events to reattempt advisory resolution

4.2 Recovery from Sensor Invalidity

- If `Radio_Altimeter_Status = Not_Valid` , system uses backup barometric altitudes
 - Advisory generation is deferred until data validity restored
-

5. Maintenance Requirements

5.1 Resetting System Flags

- After system boot or fault, reset:
 - `Effective_SL` , `Auto_SL`
 - All advisory flags and event triggers

5.2 Reinitialization upon Fault Detection

- If altitude sensors remain invalid beyond threshold, reset advisories to `No_RA`
- Recompute altitude layer on next surveillance completion

6. Quality Assurance

6.1 Altitude and VSL Value Validations

- Validate values of `Own_Alt_Rate` , `Own_Alt_Barometric` , `Own_Alt_Radio`
- Ensure no contradictory input from redundant sensors

6.2 Advisory History Tracking

- Store and compare previous `Composite_RA` values
- Use historical context to inform reentry and reversal logic

7. Security Protocols

7.1 Advisory Consistency Checks

- Validate that multiple advisories do not conflict (e.g., `Climb` + `Descend`)

- Ensure toggles (`in_RA` , `prev_in_RA`) are cleanly switched

7.2 Advisory Transition Verification

- Compare past and current RAs to avoid unnecessary reversals
 - Ensure proper reset when exiting RA mode
-

8. Environmental Considerations

8.1 Radio and Barometric Altimeter Integration

- Primary altitude source is chosen dynamically
- Altitude integrity checks performed on both sensors

8.2 Conflict-Free Mode Selector Operation

- `Mode_Selector` options (e.g., `TA_RA`, `Standby`) dictate advisory generation logic
 - Ensure configuration does not violate altitude advisory boundaries
-

9. Training Requirements

9.1 System Interpretation

- Operators must understand meaning of each advisory state
- Must interpret `Vertical_Control` indicators in context of flight path

9.2 Simulation Scenarios

- Practice scenarios with traffic encounters at varying altitudes

- Run simulations of RA reversals, inhibit toggles, and advisory drops
-

10. Document Control

10.1 Revision History

- Rev 1.0 – Initial SOP derived from formal TCAS system specification (tcas.txt)

10.2 Authorization

- Approved by: Avionics Engineering Lead
 - Reviewed by: Flight Systems Certification Authority
-

11. Process Flows and State Transitions

11.1 Advisory State Transition Logic

- No_RA → Climb / Descend → Negative → No_RA
- Transition conditions include:
 - Sensor validity
 - Climb/Descend inhibition
 - Threat altitude differentials

11.2 Climb and Descend Inhibit Sequences

- Climb_Inhibit changes when:
 - CompositeRA = NoRA
 - Configuration flags or limits trigger a change

11.3 AutoSL and EffectiveSL Computation Logic

- SL computed based on:
 - Altitude sensor state

- Mode selector
 - Traffic display permissions
- Recalculated each surveillance cycle