

branch: master

paparazzi / conf / flight_plans / OpenUAS / include_obc2014_mission.xml



dewagter Dec 22, 2014 [conf] obc fix nav_airspeed names, and enable AMSL

2 contributors

585 lines (466 sloc) | 27.829 kb

Raw Blame History

```
1 <!DOCTYPE flight_plan SYSTEM "../flight_plan.dtd">
2 <!-- This is the Flightplan of Team OpenUAS from the Netherlands for use in the
3     Australian UAV Outback Challenge of 2014.
4
5     Read about this challenge here:
6     http://www.uavoutbackchallenge.com.au/
7
8     Thanks to all involved, specifically all the help from CdW, DD, BR, GdC, AvdL
9     * TU Delft University of Technology, L&R http://www.tudelft.nl
10    * MAVLab - http://mavlab.info
11    * 1 bit squared http://lbitsquesred.com
12    * everyone else who helped out
13
14    TYPE 1 OPERATION Flightplan, fully autonomous in-air joe detection integration
15
16    Now with integrated TYPE2 operations option (soon...)
17    TYPE 2 OPERATION Flightplan, fully autonomous search, but on ground joe detection integration
18
19    STRATEGY:
20
21    # Have one flightplan which does it all, no need to upload anything!
22
23    USAGE:
24
25    NOTES:
26
27    # This flightplan contains both the SR, as default mission and the Scrutineering flight.
28    # Mission boundary coordinates are for both Scrutineering and SnR the same
29    # Alert boundaries are different for Scrutineering and SnR Missions
30
31    TODO:
32    # Latest coordinates of KMZ of 20131017, a 100% validationcheck.
33    # TODO:
34    # add times to blocks to prevent to early forcedcrash. e.g after 2 second from tkeof border cannot be crossed in reality
35    # or maybe calculate distance and look at airframe REAL max ground speed(with wind in backeve) and deduct time it would
36    # but if scrutineering day, than it can be since we test it closeby mission boundary
37
38    TESTS:
39    #If we set the air crowbrake make sure it is retracted again when contining normal ops (maybe or above a certain airspeed)
40
41    -->
42
43    <!-- Test via command xmlwf yourflightplanname.xml for welformedness of the xml document -->
44    <!-- The base lat lon of the flightplan is the AIRFIELDHOME point -->
45    <!-- The ground_alt will be set later on also with real testflight, mind the ground height profile please -->
46    <procedure>
47    <!-- ***** HEADERS ***** -->
48    <header>
49
50    #ifndef FLIGHTPLAN_HEADER_DEFINES
51    #define FLIGHTPLAN_HEADER_DEFINES
52
53    //Set Generic options
54    #include "autopilot.h"
55    //Enable AHRS Health test functions
56    #include "subsystems/ahrs.h"
57    //Enable advanced electrical power functions
58    #include "subsystems/electrical.h"
59    //Enable datalink tests
60    #include "subsystems/datalink/datalink.h"
61    //For interaction with the Flight termination Device (FTD)
```

```

62     #define RCChannel(_x) ((*fbw_state).channels[_x])
63
64 //DANGER WARNING
65 // define ACTIVATED_FTD 0
66
67
68 // PHOTOGRAMMETRY settings
69     #define PHOTOGRAMMETRY_OVERLAP 30          // 1-99 Procent
70     #define PHOTOGRAMMETRY_SIDE LAP 20        // 1-99 Procent
71     #define PHOTOGRAMMETRY_RESOLUTION 50      // mm pixel projection size
72
73 // Include airframe.h To be able to use specific variables
74     #include "generated/airframe.h"
75
76 // Completly replace with onboard recon copmptter interface
77     #ifdef DC_AUTOSHOOT_QUARTERSEC_PERIOD
78 //TODO make shooting distance not periodic
79     #define LINE_START_FUNCTION dc_autoshoot = DC_AUTOSHOOT_PERIODIC;
80     #define LINE_STOP_FUNCTION dc_autoshoot = DC_AUTOSHOOT_STOP;
81     #endif
82
83 //Enable energy control commands from within flightplan
84     #include "firmwares/fixedwing/guidance/energy_ctrl.h"
85
86 // States
87     #define AircraftIsBooting() LessThan(nav_block,4) // LessThan(nav_block,IndexOfBlock('Mission.ReadyForDeparture'))
88
89
90 #endif
91
92 </header>
93 <!-- ***** WAYPOINTS ***** -->
94 < waypoints>
95
96 <!-- Scrutineering Course -->
97 <!-- South * East
98 TST-1 = -26° 34' 58.9" * 151° 50' 16.2"
99 TST-2 = -26° 35' 04.1" * 151° 50' 44.5"
100 TST-3 = -26° 35' 23.2" * 151° 50' 45.9"
101 -->
102 <!-- extra boundary maybe add it -->
103 <!-- ABS = Alert Boundary scrutineering flight -->
104 <!-- TODO real positions and better naming-->
105 <!--
106     < waypoint name="_ABS1" lat="" lon="" height="400."/ >
107     < waypoint name="_ABS2" lat="" lon="" height="400."/ >
108     < waypoint name="_ABS3" lat="" lon="" height="400."/ >
109 -->
110
111     < waypoint name="FLARE" x="-100." y="400." height="-40."/ >
112
113 <!-- Additional VIA_EL2 Waypoint since we must go *via* EL2, and to have a little
114     more meters optimal for our scan strips entry, otherwise we could miss one
115     photo and not spot Joe. -->
116
117 <!-- Position where Outback Joe would be located. If this waypoint is situated
118     outside search area(The default situation), by this way the flightplane
119     -knows if it should start a search phase. If it is moved inside the area it
120     will start a rescue phase, a.k.a bottleneck. -->
121
122 <!-- The START from here towards target(Joe), wile dynamically calculating the RELEASE point to target the OUTBACKJOE -
123     < waypoint name="START" x="-100" y="300"/ >
124     < waypoint name="OUTBACKJOE" x="222." y="333."/ >
125     <!-- Note that 70m is the release height, we are not allowed to go below
126     60 meters and need at least some GPS height fluctuation into account
127     Also if we steeply move UP from START to RELEASE the release will maybe a
128     little more accurate, hmmm theoretically -->
129     < waypoint name="RELEASE" x="-100." y="400." height="61."/ >
130
131 <!-- For autonomous landing and drop -->
132     < waypoint name="_BASELEG" x="100" y="300"/ >
133     < waypoint name="_CLIMB" x="100" y="300"/ >
134 </ waypoints>
135
136 <!-- ***** SECTORS ***** -->
137 <!-- Sectors are good for displaying an overlay -->
138 < sectors>
139     < sector name="SurveyArea" color="green">
140         < corner name="SA1"/ >

```

```

141     <corner name="SA2"/>
142     <corner name="SA3"/>
143     <corner name="SA4"/>
144     <corner name="SA5"/>
145 </sector>
146 </sectors>
147
148 <!-- ***** EXCEPTIONS ***** -->
149 <exceptions>
150 <!-- WARNING these are Global Exeption, bettersome to stuff in a flightplan block-->
151     <!-- If datalink is lost for more than 10 seconds and we are in launched state -->
152
153     <!-- below for the real SnR --> <!-- TODO: if datalinkloss but still closeby move to commhold? strange better airfi
154     <exception cond="And((datalink_time/2) > 9, launch > 0) && (!AircraftIsBooting())" deroute="CommsHoldLoiter"/> <!--
155
156     <!-- <exception cond="if Joefoundandretunvalformticket deroute="Waitforjudges"/>-->
157
158     <!-- Optional check, if at seconds of launch GPS is lost -->
159     <!-- <exception cond="And(launch, gps_lost)" deroute="killengine" or test for heighht 40M or so/> -->
160     <!-- see if we can get correct AP voltage -->
161
162     <!-- not used yet maybe switch it on <exception cond="10.0 > PowerVoltage()" deroute="Standby"/> -->
163     <!-- not used yet maybe switch it on <exception cond="11.0 > vsupply" deroute="Standby"/> -->
164     <!-- is less safe and has less chance of AC recovery. It states battery LOW, not battery critical -->
165     <!-- <exception cond="electrical.bat_low && !(nav_block == IndexOfBlock('Setting_home_location')) && !(nav_block == Inc
166     <!--<TODO: Add somethin like pre_call="if (!(InsideRed(GetPosX(), GetPosY())) NavKillThrottle());-->
167 </exceptions>
168
169
170
171
172
173
174
175
176
177 <!-- ***** -->
178 <!-- ***** FLIGHTPLAN STARTINGPOINT ***** -->
179 <!-- ***** -->
180
181 <!-- ***** Wait for GPS fix, 3D by default ***** -->
182 <blocks>
183     <block name="WaitForGPS" pre_call="NavKillThrottle()" >
184         <!-- Do Not Kill -->
185         <call fun="Fly()"/>
186         <!-- IDEA Close the hatch so not able to insert payload before there is a 3D fix , however maybe not so handy anyh
187         <!-- <set value="NavDropCloseHatch" var="unit"/> -->
188         <!-- if no valid fix or gps accuracy> 15m or no AHRS , it a no-go wait-->
189         <while cond="!GpsFixValid() || gps.pacc > 1500 || !(ahrs.status == AHRS_RUNNING)"/>
190     </block>
191
192 <!-- ***** Set the ground reference height and the home position ***** -->
193 <!-- a second init will follow since the plane thrower will still walk to launchpoint -->
194
195     <block name="Geo Init">
196         <while cond="LessThan(NavBlockTime(), 3)"/>
197     <!--
198     <while cond="LessThan(nav_block, IndexOfBlock('Mission.ReadyForDeparture'))"/>
199     <call fun="NavSetGroundReferenceHere()"/>
200     <call fun="NavSetWaypointHere(WP_HOME)"/>
201 -->
202     <!-- TODO Set QNH here based on GPS height if allowed from committee -->
203     <call fun="NavSetAltitudeReferenceHere()" />
204     <set var="air_data.calc_qnh_once" value="TRUE"/>
205 </block>
206
207     <!-- ***** Set the ground reference height and the home position ***** -->
208
209 <!-- a second init will follow since the plane thrower will still walk to launchpoint-->
210
211     <block name="Wait for RC">
212         <!-- We must have RC at least once switched on just to test, this check can be deleted if everything works 100% in
213         <while cond="RCLost()"/> <!-- to make sure we do not hop to AUTO2 and engine starts running , at least switch on t
214         <!-- TODO maybe switch on ACL blick as soon as we are ready to go -->
215     </block>
216
217     <block name="Set Landing Position" pre_call="NavKillThrottle()">
218         <set value="AirbrakesOff()" var="unit"/>
219         <attitude pitch="0" roll="0" throttle="0" vmode="throttle" until="FALSE"/><!-- maybe not moving the plane would st

```

```

220 </block>
221
222 <block name="Landing Position Stored" pre_call="NavKillThrottle()" strip_button="Store" strip_icon="recenter.png">
223 <call fun="NavSetWaypointPosAndAltHere(WP_TD)"/>
224 <set value="DownlinkSendWpNr(WP_TD)" var="unit"/>
225 <!-- CDW set value="SetAltitudeForFinalFromTo(WP_TD,WP_AF)" var="unit"/-->
226 <call fun="nav_compute_final_from_glide(WP_AF, WP_TD, V_CTL_GLIDE_RATIO)"/>
227 <!-- CDW call fun="nav_compute_flare_from_aftd(WP_AF, WP_TD, WP_FLARE)"/-->
228 <set value="DownlinkSendWpNr(WP_AF)" var="unit"/>
229 <set value="DownlinkSendWpNr(WP_FLARE)" var="unit"/>
230 <deroute block="HoldingPoint"/>
231 </block>
232
233 <!-- ***** Throttle off holdingpoint ***** -->
234 <block name="HoldingPoint" pre_call="NavKillThrottle()" >
235 <!-- TODO: make some movements on all defelctros as a sign we are in holding and to see if all works Maybe a sepa
236 <!-- <set var="waypoints[WP_START].x" value="WaypointX(WP_EL2) + 50"/> -->
237 <!-- <set var="waypoints[WP_START].y" value="WaypointY(WP_EL2) + 50"/> -->
238 <!-- <set value="0" var="ap_state->commands[COMMAND_CAMBER]"/> -->
239 <!-- <set value="0" var="ap_state->commands[COMMAND_BRAKE]"/> -->
240 <set value="AirbrakesOff()" var="unit"/>
241 <attitude pitch="0" roll="0" throttle="0" vmode="throttle" until="FALSE"/><!-- maybe not moving the plane would st
242 </block>
243
244 <!-- ***** Search Area Entry Procedure Start *****
245 *****
246 ***** -->
247 <block name="Start Scruteneering Flight" strip_button="Scrut" group="TO">
248 <!-- CDW set value="SCRUTENEERING_FLIGHT" var="obc_flight_mode" /-->
249 <deroute block="ReadyForDeparture"/>
250 </block>
251
252 <block name="Start Search Flight" strip_button="Search" group="TO">
253 <!-- CDW set value="SEARCH_FLIGHT" var="obc_flight_mode" /-->
254 <deroute block="ReadyForDeparture"/>
255 </block>
256
257 <block name="Start Drop Flight" strip_button="Drop" group="TO">
258 <!-- CDW set value="DROP_FLIGHT" var="obc_flight_mode" /-->
259 <deroute block="ReadyForDeparture"/>
260 </block>
261
262 <block name="ReadyForDeparture" >
263 <exception cond="ThrottleHigh()" deroute="Takeoff"/>
264 <attitude pitch="nav_takeoff_pitch_setting" roll="0" throttle="0.0" until="FALSE" vmode="throttle"/>
265 </block>
266
267 <block key="t" name="Takeoff" strip_button="Takeoff" strip_icon="takeoff.png">
268 <!-- <set value="3000" var="ap_state->commands[COMMAND_CAMBER]"/> -->
269 <set value="0" var="kill_throttle"/>
270 <set value="1" var="launch"/>
271 <set value="0" var="autopilot_flight_time"/>
272 <set var="unit" value="AirbrakesOff()"/>
273 <set value="nav_airspeed_nominal_setting" var="v_ctl_auto_airspeed_setpoint"/>
274 <attitude pitch="nav_takeoff_pitch_setting" roll="0" throttle="1.0" until="MoreThan(NavBlockTime(), 2)" vmode="thr
275 </block>
276
277 <block name="Climb-Out">
278 <!--<exception cond="rc_status==RC_REALLY_LOST" after 3 seconds and no IMU deroute="softlydown">if using RC stil c
279 <attitude pitch="nav_takeoff_pitch_setting" roll="0" throttle="1.0" until="GetPosAlt() > ground_alt+40" vmode="thr
280 <set var="v_ctl_auto_throttle_nominal_cruise_throttle" value="1.0" />
281 <set var="v_ctl_auto_throttle_nominal_cruise_pitch" value="nav_takeoff_pitch_setting" />
282 </block>
283
284 <block name="determineflighttype">
285 <!-- CDW exception cond="(obc_flight_mode == SCRUTENEERING_FLIGHT)" deroute="ScrutineeringAutonomyTriangle"/-->
286 <!-- otherwise goto search area via entry lane -->
287 <deroute block="EntryLane1"/>
288 </block>
289
290 <!-- *****
291 ***** Scrutineering Flight *****
292 ***** -->
293
294 <!--
295 As part of the scrutineering the Search and Rescue entrants of the UAV Challenge
296 Outback Rescue will be required to take off, track TST-1, TST-2, TST-3, TST-1... Until
297 requested to land.
298

```

```

299 All emergency and failure requirements of the Challenge must be met,
300
301 with the exception that for Loss of Data Link the procedure will be track direct
302 to "Airfield Home" and orbit for either a landing or regain of Data Link.
303 -->
304
305 <block name="ScrutineeringAutonomyTriangle" strip_button="Triangle" strip_icon="path.png" group="Mission">
306 <!-- TODO: while we are up there take some pictures anyhow :) -->
307 <for var="i" from="1" to="3">
308 <go from="TST-3" hmode="route" wp="TST-1"/>
309 <go from="TST-1" hmode="route" wp="TST-2"/>
310 <go from="TST-2" hmode="route" wp="TST-3"/>
311 </for>
312 <deroute block="ComputeLandingApproach" />
313 </block>
314
315 <!-- *****
316 ***** Search Area Entry Procedure Start *****
317 ***** -->
318
319 <!-- maybe a HelperWaypoint to gettart n return from a mission Tracking to EL1
320 <set value="0" var="ap_state->commands[COMMAND_CAMBER]"/>
321
322 <!-- ***** Track to EL-1 ***** -->
323 <block name="EntryLane1" strip_button="Joe" strip_icon="lookdown.png" group="Mission">
324 <set value="nav_airspeed_tracking_setting" var="v_ctl_auto_airspeed_setpoint"/>
325 <go from="AIRFIELDHOME" wp="EL1" hmode="route"/>
326 </block>
327 <!-- maybe add other waypoint to enterarea in a better way -->
328
329 <!-- ***** Track Direct to EL-2, Entry to Search Area only via EL-2 ***** -->
330 <block name="EntryLane2">
331 <set value="nav_airspeed_tracking_setting" var="v_ctl_auto_airspeed_setpoint"/>
332 <go from="EL1" wp="EL2" hmode="route" approaching_time="10" />
333 </block>
334
335 <block name="determinerescuetype">
336 <!-- CDW exception cond="(obc_flight_mode == DROP_FLIGHT)" deroute="RescueJoe"/-->
337 <deroute block="SearchForJoe" />
338 </block>
339
340 <!-- "Type 1" operation:
341 an areascan with taking pictures of ground realtime anyyse and if joe found an acknolege signal given , autonomous
342
343 "Type 2"
344 * There is a search phase and a rescue phase.
345 ** Search means areascan gathering the imagery needed to pinpoint the location of lost Joe ("Type 1" operation)
346 ** Rescue means the payload dropping phase. ("Type 1" operation)
347 -->
348 <!-- TODO: if defined TYPE2 then -->
349 <!--
350 <block name="determine_search_or_rescue_phase">
351 <exception cond="InsideSurveyArea(WaypointX(WP_OUTBACKJOE), WaypointY(WP_OUTBACKJOE))" deroute="RescueJoe"/>
352 </block>-->
353
354 <!-- *****
355 See for calculations of camera values used
356 http://www.openuas.org/intranet/webapps/pmwiki/pmwiki.php?n=UAS.Targetcalc
357 Max Scanwidth at 120m is 180m
358 Mission scanwidth is 144m thus with a 20% overlap. Overlap depends on lots of factors,
359 best to perform many real life tests with the CAM
360 ***** -->
361
362 <block name="SearchForJoeLateral">
363 <call fun="nav_survey_poly_osam_setup_towards(WP_SA1, 5, 158, WP_SA2)"/> <!-- TODO correct angle of how we want to
364 <deroute block="Searching" />
365 </block>
366
367 <block name="SearchForJoeDiagonal1">
368 <call fun="nav_survey_poly_osam_setup(WP_SA1, 5, 158, 30)"/> <!-- TODO correct angle of how we want to search -->
369 <deroute block="Searching" />
370 </block>
371
372 <block name="SearchForJoeDiagonal2">
373 <call fun="nav_survey_poly_osam_setup(WP_SA1, 5, 158, 55)"/> <!-- TODO correct angle of how we want to search -->
374 <deroute block="Searching" />
375 </block>
376
377 <block name="SearchForJoe">

```

```

378 <!-- <exception cond="FoundJoe" deroute="WaitForJugesDropApprovale"/> -->
379 <!-- Parameters
380 The orientation of the sweeps can ranges from north south to east west and
381 any where in between (-90 <-> 90 degrees respectively).
382 The side of the polygon the aircraft starts on (ex. north or south) is
383 determined by the side of the polygon the entry point is located.
384 call fun="InitializePolygonSurvey(WP_S1, NumOfCorners, SweepWidth, Orientation)"/> -->
385
386 <!-- SweepWidth max photo with at height - 20 pct overlap 180- etc-->
387 <call fun="nav_survey_poly_osam_setup_towards(WP_SA1, 5, 158, WP_SA5)"/> <!-- TODO correct angle of how we want to
388 </block>
389
390 <block name="Searching">
391 <exception cond="PolySurveySweepBackNum >= 1" deroute="ReturnToBase"/> <!-- ? TODO decide to search in reverse ag
392 <set value="nav_airspeed_nominal_setting" var="v_ctl_auto_airspeed_setpoint"/>
393 <call fun="nav_survey_poly_osam_run()"/>
394 </block>
395
396 <!--WaitForJugesDropApproval if we do a Type 1 search otherwise it would be approved already Warning then never enter th
397 <block name="WaitForJugesDropApproval">
398 <!-- circle with circle border already alligend with Drop align route, or maybe some better ideas... -->
399 <!-- TODO: if blocktime approval takes to long do what? -->
400 </block>
401
402 <!-- *****
403 ***** Search Area Departure Procedure *****
404 ***** -->
405
406 <!-- ***** Exit Search Area only via EL-3 ***** -->
407 <block name="ReturnToBase">
408 <set value="nav_airspeed_tracking_setting" var="v_ctl_auto_airspeed_setpoint"/>
409 <go wp="EL3"/>
410 </block>
411
412 <!-- ***** Track Direct to EL-4 ***** -->
413 <block name="ExitLane3">
414 <set value="nav_airspeed_tracking_setting" var="v_ctl_auto_airspeed_setpoint"/>
415 <go from="EL3" wp="EL4" hmode="route"/>
416 </block>
417
418 <!-- ***** Track to Kingaroy Airport ***** -->
419 <block name="ExitLane4">
420 <set value="nav_airspeed_tracking_setting" var="v_ctl_auto_airspeed_setpoint"/>
421 <go from="EL4" wp="AIRFIELDHOME" hmode="route"/>
422 <deroute block="ComputeLandingApproach"/>
423 </block>
424
425 <!-- *****
426 ***** Deliver the Payload *****
427 ***** -->
428
429 <block name="RescueJoe" strip_button="Drop" strip_icon="parachute.png" group="Mission">
430 <!-- <go wp="VIAEL2" hmode="route"/>--> <!-- TODO maybe not needed -->
431 <deroute block="drop"/>
432 </block>
433
434 <!-- AS REMINDER
435 extern unit_t nav_drop_compute_approach( uint8_t wp_target, uint8_t wp_start, float radius );
436 extern unit_t nav_drop_update_release( uint8_t wp_target );
437 extern unit_t nav_drop_shoot( void );
438 extern float nav_drop_trigger_delay, nav_drop_start_qdr;
439 extern bool_t compute_alignment(uint8_t w1, uint8_t w2, uint8_t start, uint8_t end, float d_before, float d_after);
440
441 #define NavDropComputeApproach(_target, _start, _radius) nav_drop_compute_approach(_target, _start, _radius)
442 #define NavDropUpdateRelease(_wp) nav_drop_update_release(_wp)
443 #define NavDropShoot() nav_drop_shoot()
444 #define NavDropCloseHatch() ( { ap_state->commands[COMMAND_HATCH] = MIN_PPRZ; } )
445 #define NavDropAligned() Qdr(DegOfRad(nav_drop_qdr_aligned)) -->
446
447
448 <!-- ***** Drop Compute Approach ***** -->
449 <block name="drop">
450 <!-- TODO set Height to minimum height AGL allowed by commitee ?60M -->
451 <!-- TODO determine what still is success and is fastest value for this 100 -->
452 <!--set var="waypoints[WP_START].x" value="WaypointX(WP_OUTBACKJOE) + 100"/>-->
453 <!--set var="waypoints[WP_START].y" value="WaypointY(WP_OUTBACKJOE) + 100"/>-->
454
455 <set value="nav_drop_compute_approach(WP_OUTBACKJOE, WP_START, WP__BASELEG, WP__CLIMB, nav_radius)" var="unit"/>
456 <set value="DownlinkSendWpNr(WP_RELEASE)" var="unit"/>

```

```

457     <circle radius="nav_radius" until="NavQdrCloseTo(DegOfRad(nav_drop_start_qdr)-15)" wp="_BASELEG"/>
458 </block>
459
460 <!-- ***** Align on release path ***** -->
461 <block name="align">
462     <!-- TODO set AIRSPEED_AT_RELEASE-->
463     <!--TODO set HEIGHT to 60m -->
464     <exception cond="nav_drop_update_release(WP_OUTBACKJOE)" deroute="ReturnToBase"/>
465     <go approaching_time="nav_drop_trigger_delay" from="START" hmode="route" wp="RELEASE"/>
466 </block>
467
468 <!-- ***** Release Payload ***** -->
469 <block name="shoot">
470     <!-- TODO maybe extend crowbreak 80%? for slowes possible speed? -->
471     <!-- <set value="0" var="ap_state->commands[COMMAND_BRAKE]"/> -->
472     <!-- TODO is electrical plane and almost there maybe stop proppelor from spinning for a few seconds -->
473     <set value="NavDropShoot()" var="unit"/>
474     <!-- TODO add maxspeed again -->
475     <go from="RELEASE" hmode="route" wp="EL3"/>
476     <!-- Maybe better not to close, more chance of delivery, TODO maybe even add a sensor to detect release if not try
477     <!-- <set value="NavDropCloseHatch" var="unit"/> -->
478
479     <deroute block="ReturnToBase"/><!-- Just To Make Sure -->
480 </block>
481
482 <!-- ***** Autolanding *****
483 ***** Autolanding *****
484 ***** -->
485
486 <!-- ***** Initiate landing ***** -->
487 <!--TODO better routing in combination with calced wind direction-->
488
489 <!-- ***** Land ***** -->
490
491 <block key="r" name="ConstantSlopeLandRightAFTD" strip_button="Land right (wp AF-TD)" strip_icon="land-right.png" g
492     <set value="LANDING_CIRCLE_RADIUS" var="nav_radius"/>
493     <deroute block="ComputeLandingApproach"/>
494 </block>
495
496 <block key="l" name="ConstantSlopeLandLeftAFTD" strip_button="Land left (wp AF-TD)" strip_icon="land-left.png" grou
497     <set value="-LANDING_CIRCLE_RADIUS" var="nav_radius"/>
498     <deroute block="ComputeLandingApproach"/>
499 </block>
500
501 <block name="ComputeLandingApproach" >
502     <call fun="nav_compute_final_from_glide(WP_AF, WP_TD, V_CTL_GLIDE_RATIO)"/>
503     <!-- CDW call fun="nav_compute_flare_from_aftd(WP_AF, WP_TD, WP_FLARE)"/>
504     <set value="DownlinkSendWpNr(WP_AF)" var="unit"/>
505     <set value="DownlinkSendWpNr(WP_FLARE)" var="unit"/>
506 </block>
507
508 <block name="Land">
509     <set value="nav_airspeed_nominal_setting" var="v_ctl_auto_airspeed_setpoint"/>
510     <call fun="nav_compute_baseleg(WP_AF, WP_TD, WP_BASELEG, nav_radius)"/>
511     <circle radius="nav_radius" until="NavCircleCount() > 0.5" wp="_BASELEG"/>
512     <set value="nav_airspeed_landing_setting" var="v_ctl_auto_airspeed_setpoint"/>
513     <circle radius="nav_radius" until="NavQdrCloseTo(DegOfRad(baseleg_out_qdr)-(nav_radius/fabs(nav_radius))*10) && 15
514 </block>
515
516 <block name="Final">
517     <set value="AirbrakesOn()" var="unit"/>
518     <go approaching_time="8" from="AF" hmode="route" wp="TD"/>
519 </block>
520
521 <block name="Flare">
522     <call fun="NavKillThrottle()" />
523     <set value="nav_airspeed_landing_setting*0.8" var="v_ctl_auto_airspeed_setpoint"/>
524     <go approaching_time="0" from="TD" hmode="route" wp="FLARE"/>
525     <!-- TODO: retract brake after landed is true (with +3seconds margin) better a sensor(witch) that detects IsAircra
526 </block>
527
528
529 <!-- *****
530 ***** Loss of Data Link - Rally Point *****
531 ***** -->
532 <!--
533 TODO: get OBC2014Rules in 5.5.2.1 Loss of Data Link - Rally Point
534
535 * This flight mode must be activated if the data link is lost for 10 seconds.

```

```

536
537 * This flight mode, once activated, must return the aircraft directly to the
538 waypoint "Comms Hold" (Figure 1, Table 1) (no time limit for transit, but it
539 must be direct to the waypoint)
540
541 * and enter a loiter centred on that waypoint.
542 * After 2 minutes of loitering,
543
544 * if the data link has not been re-established, the aircraft must be programmed to return directly to
545 waypoint "Airfield Home" (Figure 1, Table 1), where the RC link
546 maybe re-established for a manual recovery. There is no time limit for
547 the transit, but it must be direct to the waypoint.
548
549 * On arriving at "Airfield Home", the aircraft must enter a loiter centred
550 on that waypoint.
551
552 * If after 2 minutes, RC contact has not been re-established, an autopilot initiated and controlled flight termination
553 mode must be activated. A controlled flight termination mode may be
554 (but not limited to) an autonomous landing or entering the flight
555 termination mode (Section 5.6).
556 -->
557
558 <block name="CommsHoldLoiter" strip_button="CommsHold" strip_icon="recenter.png" group="Link" > <!-- TODO other ICC
559 <!-- stop streaming thumbnails -->
560 <!-- CDW set value="0" var="ticket_thumbnails"/ -->
561 <!-- TODO: maybe set a little lower speed ? max endurance speed? -->
562 <set value="nav_airspeed_nominal_setting" var="v_ctl_auto_airspeed_setpoint"/>
563 <!-- Rules 2014 state a loiter of maximum 2 minutes = 120seconds is allowed -->
564 <exception cond="(stage_time > 120)" deroute="AirfieldHomeLoiter"/>
565 <!-- resume if datalink is regained-->
566 <!--TODO If the data link is regained prior to the conclusion of 2 minutes of loiter at
567 Comms Hold the mission maybe continued otherwise any subsequent TODO -->
568 <!--exception cond="LessThan(datalink_time0, 10)" deroute="EntryLane2"/--> <!-- As soon as we have a link aga
569 <!-- TODO: better continue where we had our last telemerty still OK point if possible -->
570 <!-- Since it arrived we must raise a counter by one. it's not allowed by rules to occur more than twice, then we
571 <!-- TODO: set only after aircraft **arrived!** in the loiter not before the ETA < 1sec -->
572 <!-- <while no link then -->
573 <circle wp="COMMSHOLD" radius="nav_radius"/> <!-- TODO maybe bigger than default radius -->
574 </block>
575
576 <block name="AirfieldHomeLoiter" strip_button="AirfieldHome" strip_icon="home.png" group="Link">
577 <exception cond="stage_time > 120" deroute="ConstantSlopeLandRightAFD"/>
578 <circle wp="AIRFIELDHOME" radius="nav_radius*1.5"/><!-- 1.5 x mean it will be a bit bigger circle -->
579 </block>
580
581
582 </blocks>
583 </procedure>
584

```

