



Implementacja systemów sterowania systemami bezzałogowymi na układy STM32 z wykorzystaniem technologii Model-Based-Design (testy typu PIL,HIL)

mjr dr inż. Marcin CHODNICKI



SYSTEMY BEZZAŁOGOWE

WIELOWIRNIKOWCE ROZPOZNAWCZE I BOJOWE

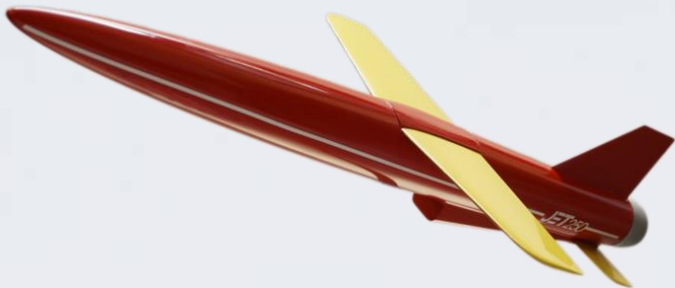


OBECNIE
REALIZOWANY
PROJEKT:
PLATFORMA WODNO-
POWIETRZNA



SAMOLOTY

POJAZDY NAZIEMNE



MODEL MATEMATYCZNY OBIEKTÓW LATAJĄCYCH

$$\dot{U} = \frac{F_x}{m} + rV - qW$$

$$\dot{V} = \frac{F_y}{m} + pW - rU$$

$$\dot{W} = \frac{F_z}{m} + qU - pV$$

$$\dot{p} = \frac{M_x + (I_y - I_z)qr}{I_x}$$

$$\dot{q} = \frac{M_y + (I_z - I_x)rp}{I_y}$$

$$\dot{r} = \frac{M_z + (I_x - I_y)pq}{I_z}$$

$$\dot{\Phi} = p + (q \sin \Phi + r \cos \Phi) \tan \theta$$

$$\dot{\theta} = q \cos \Phi - r \sin \Phi$$

$$\dot{\Psi} = \frac{r \cos \Phi + q \sin \Phi}{\cos \theta}$$

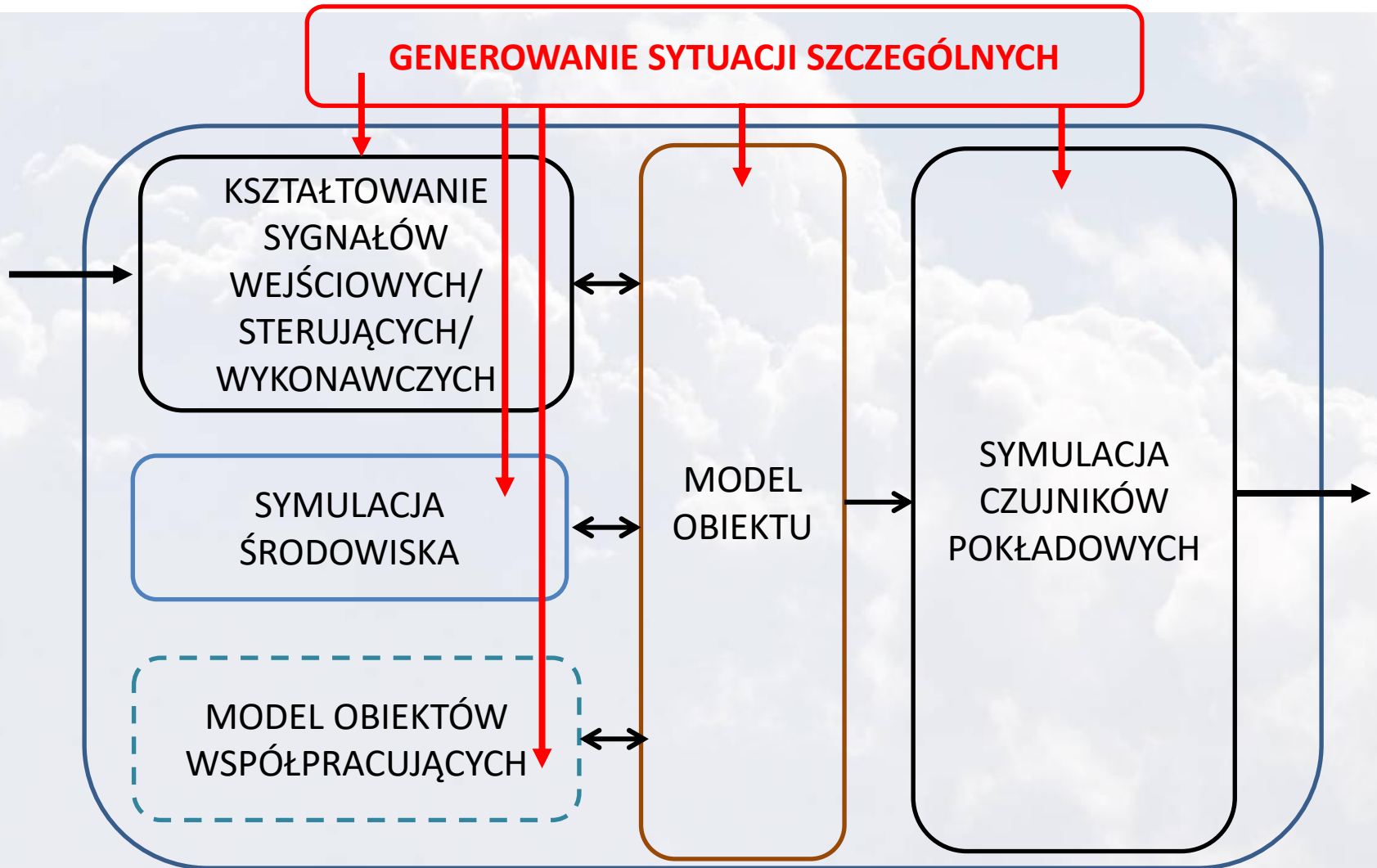
wektor parametrów lotu

$$X = [U, V, W, p, q, r, \phi, \theta, \Psi, x_g, y_g, z_g]^T$$





MODEL SYMULACYJNY





PROJEKTOWANIE W MATLAB SIMULINK

Source Control Information

Integration: Built-In SVN Integration (1.8)

Repository location: file:///Z:/Repozytorium/Senior/AP_Rover

To change the source control integration being used, select one from the list of available tools and select "Reload" to reload the project.

Built-In SVN Integration (1.8) [Reload]

The Built-In SVN Integration supports SVN sandboxes and repositories at version 1.8 and below. You do not need to install SVN to use this integration, because it includes an implementation of SVN. This integration ignores any existing SVN installation. If you need support for a later version of SVN, then select "Command-Line SVN Integration" in the list above, which will use your installed copy of the command-line SVN client.

| Name | Status | SVN | Revision | Review | Classification |
|----------------------------|--------|-----|----------|--------|----------------|
| Analiza_logow | ✓ | ● | 4 | | None |
| branches | ✓ | ● | 4 | | None |
| code_gen | ✓ | ● | 4 | | None |
| Cube_MX | ✓ | ● | 5 | | None |
| PdistToLineAB2D.m | ✓ | ● | 4 | | Design |
| RTOS_Tasks.m | ✓ | ● | 5 | | Design |
| RTOS_Tasks_AP_PIL_STM_H7.m | ✓ | ● | 5 | | Design |
| RTOS_Tasks_AP_STM_v2_3.m | ✓ | ● | 5 | | Design |
| strefa_wielokat.m | ✓ | ● | 4 | | Design |

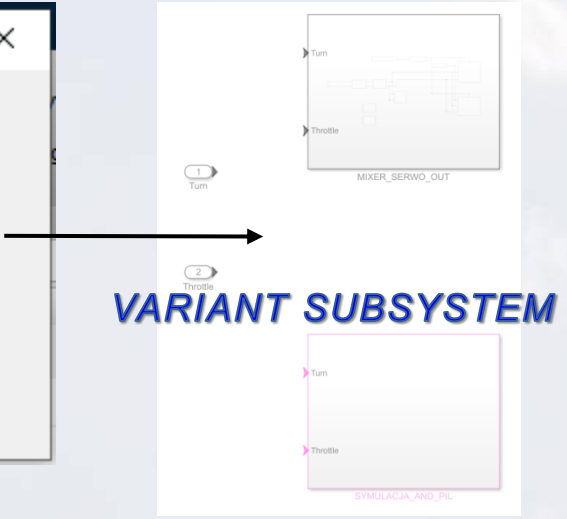
SIMULINK PROJECT + SVN

GUI

PCB

- AP_Rover_symulacja
- AP_Rover_PIL_H7
- AP_Rover_STM32F4_v2_3

Uruchom



Code Generation Report for 'rtwdemo_rtweintro'

Contents

- Summary
- Subsystem Report
- Code Interface Report
- Traceability Report
- Static Code Metrics Report
- Code Replacements Report

Generated Code

- Main file: [ert_main.c](#)
- Model files: [rtwdemo_rtweintro.c](#), [rtwdemo_rtweintro.h](#)
- Shared files (3)

Model Information

| | |
|------------------|---------------------|
| Author | The MathWorks, Inc. |
| Last Modified By | The MathWorks, Inc. |
| Model Version | 1.259 |
| Tasking Mode | MultiTasking |

Configuration settings at time of code generation

Code Information

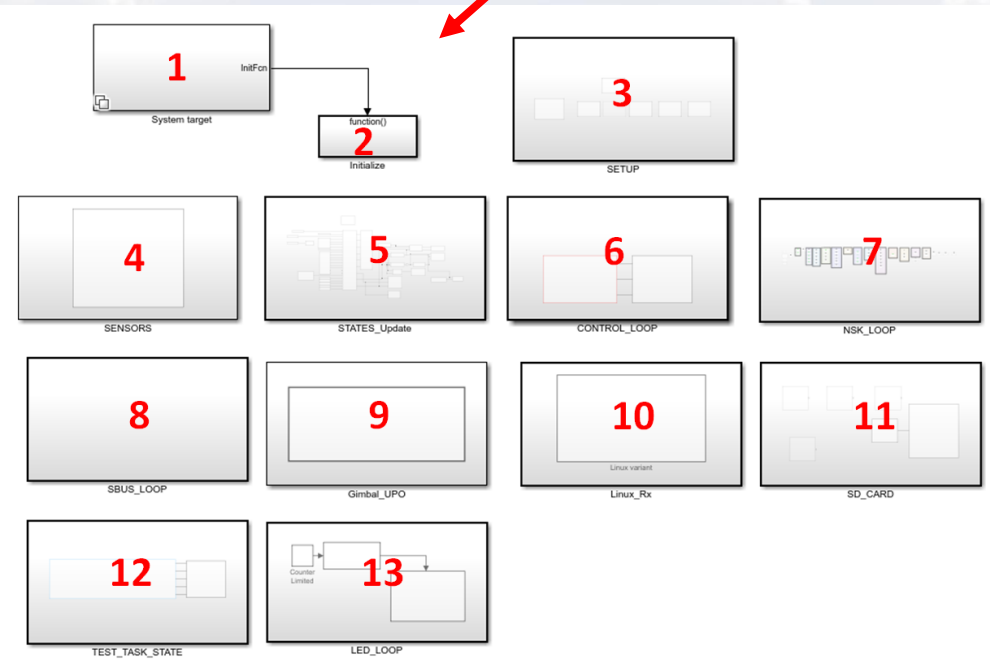
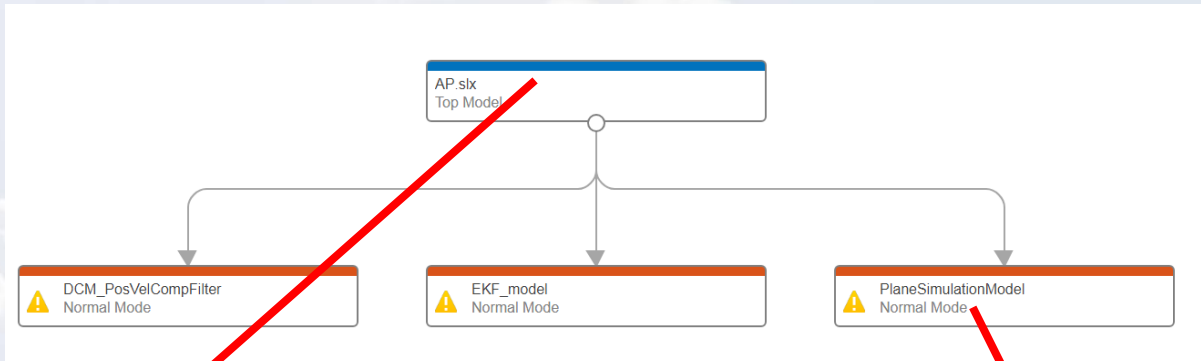
| | |
|------------------------------------|---|
| System Target File | ert.tlc |
| Hardware Device Type | Intel->x86-64 (Windows64) |
| Simulink Coder Version | 8.12 (R2017a) 22-Dec-2016 |
| Timestamp of Generated Source Code | Thu Jan 05 14:59:56 2017 |
| Location of Generated Source Code | H:\Documents\MATLAB\examples\ecoder-ex93319999\rtwdemo_rtweintro_ert_rtw\ |
| Type of Build | Model |
| Objectives Specified | Execution efficiency, Traceability |

Additional Information

| | |
|-------------------------|---------|
| Code Generation Advisor | Not run |
|-------------------------|---------|



UPROSZCZONA STRUKTURA MODELU SKL

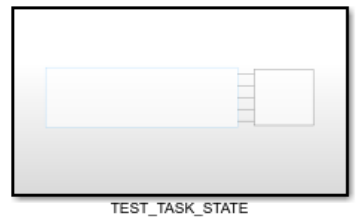
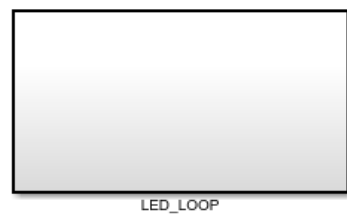
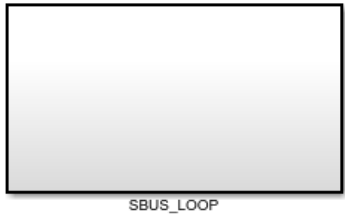
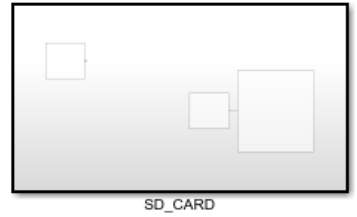
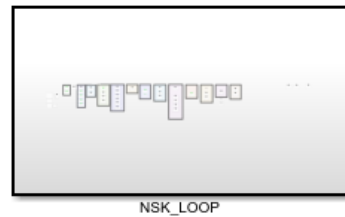
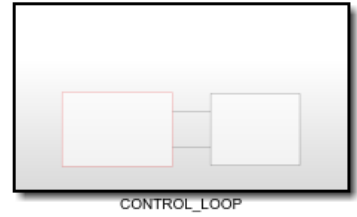
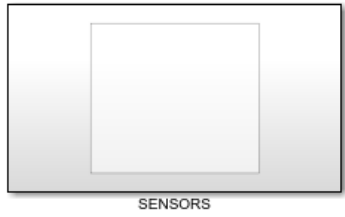
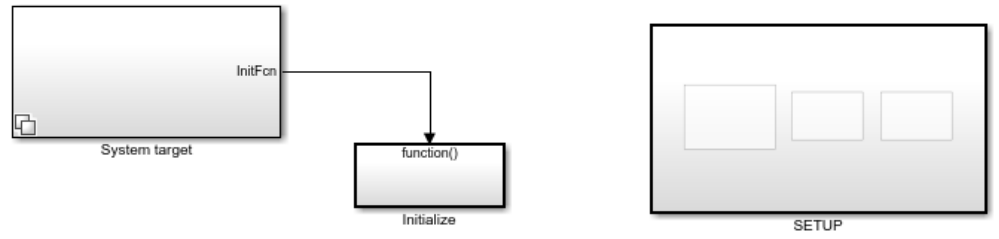




MODEL PROJEKTU SYSTEMU KONTROLI LOTU (SKL)

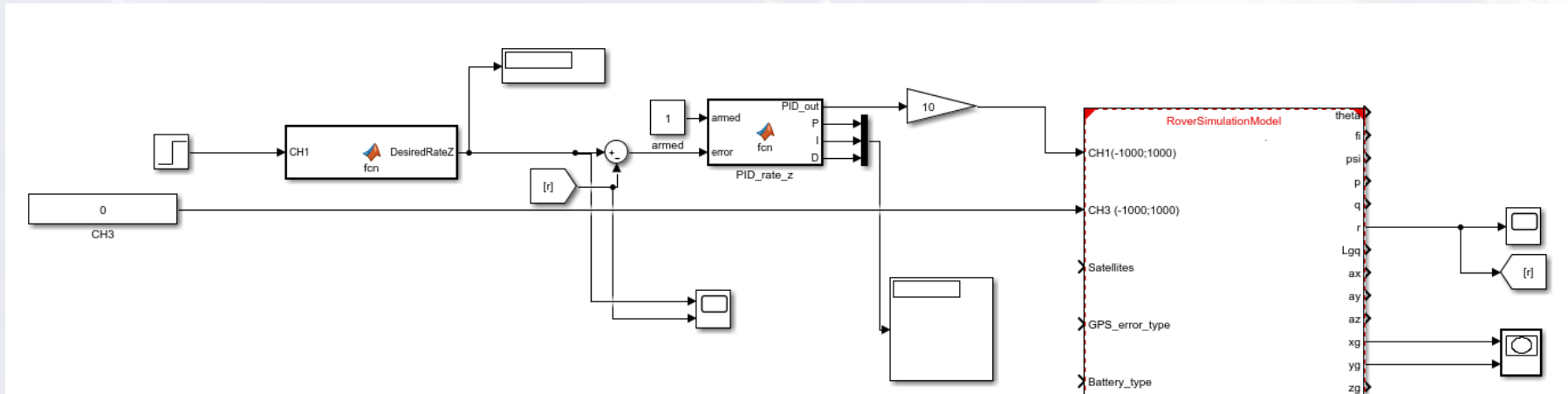
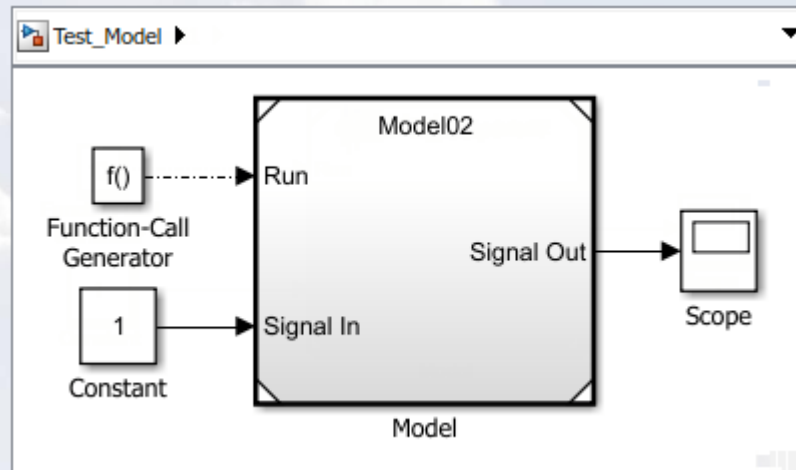
```
%% Task 1 Control_loop 400Hz
i = i+1;
tasks(i,1)={'ControlLoop'}; % name of task
tasks(i,2)={'3'};           % priority '-3'-idle, '3'-real time
tasks(i,3)={'3468'};       % stack size (min. 128)
tasks(i,4)={'400'};        % frequency task
tasks(i,5)={'1000'};       % time out us
tasks(i,6)={'0'};          % max count of error
tasks(i,7)={'IMU_Update'}; % function_1
tasks(i,8)={'STATES_Update'};
tasks(i,9)={'CONTROL_LOOP'};
tasks(1,10)={'TEST_TASK_STATE'};
tasks(i,11)={'SD_CARD'};
```

```
164 //Task
165 void StartTask_ControlLoop(void const * argument)
166 {
167     TickType_t xLastWakeTime;
168     const TickType_t xFrequency = configTICK_RATE_HZ/400;
169     xLastWakeTime = xTaskGetTickCount();
170
171     #ifdef task_diagnostic_Handle_present
172
173     uxHighWaterMark_task1 = uxTaskGetStackHighWaterMark( NULL );
174
175     #endif
176
177     /* Infinite loop */
178     for (;;) {
179
180     #ifdef task_diagnostic_Handle_present
181
182     get_tim_counter(&tim7, &task1_time_start);
183
184     #endif
185
186     AP_IMU_Update();
187     AP_STATES_Update();
188     AP_CONTROL_LOOP();
189     AP_TEST_TASK_STATE();
190     AP_SD_CARD();
191
192     #ifdef task_diagnostic_Handle_present
193
194     uxHighWaterMark_task1 = uxTaskGetStackHighWaterMark(NULL);
195     if (uxHighWaterMark_task1 <= 30 &&1>0) {
196         //osThreadSuspend(NULL);
197         osThreadTerminate(NULL);
198     }
199
200     get_loop_time(&tim7, &task1_time_start, &task1_time, &task1_time_update);
201
202     #endif
203
204     vTaskDelayUntil( &xLastWakeTime, xFrequency );
205 }
206 }
207
```



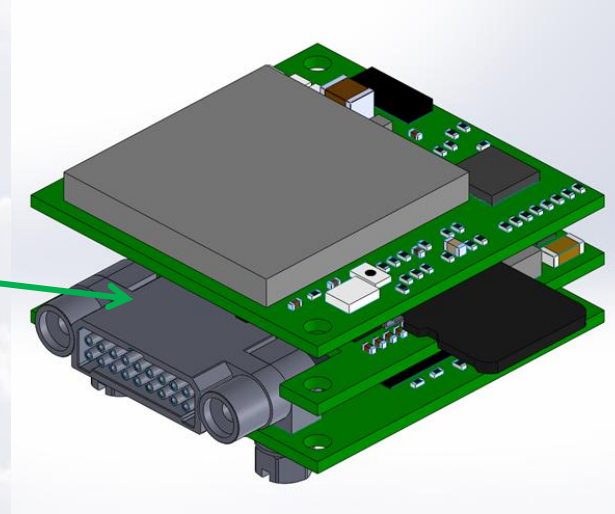


TESTY METODĄ MODEL-IN-THE-LOOP / JAŻMA TESTOWE



TESTY METODĄ HARDWARE/PROCESSOR/SOFTWARE-IN-THE-LOOP

Matlab/Simulink Embedded Coder



Autopilot

Komputer PC

Naziemna Stacja Kontroli (NSK)

Wizualizacja 3D



TESTY NA MODELU SKALOWALNYM I PLATFORMIE DOCELOWEJ

PLATFORMY TESTOWE



PLATFORMY DOCELOWE

