

## **System zobrazowania badawczego urządzenia symulującego lot śmigłowca**

**Andrzej ĆWIK, Marek SALAMON, Wojciech SATAŁA,  
Andrzej SZYMAŃSKI**

Instytut Teleinformatyki i Automatyki WAT  
ul. Kaliskiego 2, 00-908 Warszawa

**STRESZCZENIE:** W artykule przedstawiono opis systemu zobrazowania Badawczego Urządzenia Symulacyjnego (BUS) przeznaczonego do analizy zachowania się w trakcie lotu śmigłowca jednowirnikowego. Prace nad systemem prowadzone były w ramach grantu KBN realizowanego w Wydziale Mechatroniki WAT. Zespół autorski odpowiedzialny był za projekt i implementację podsystemów związanych ze zobrazowaniem, rejestracją i odtwarzaniem przebiegu lotu. W wyniku prowadzonych prac system został uruchomiony, przetestowany i zainstalowany na stanowisku laboratoryjnym wyposażonym w makietę kabiny śmigłowca.

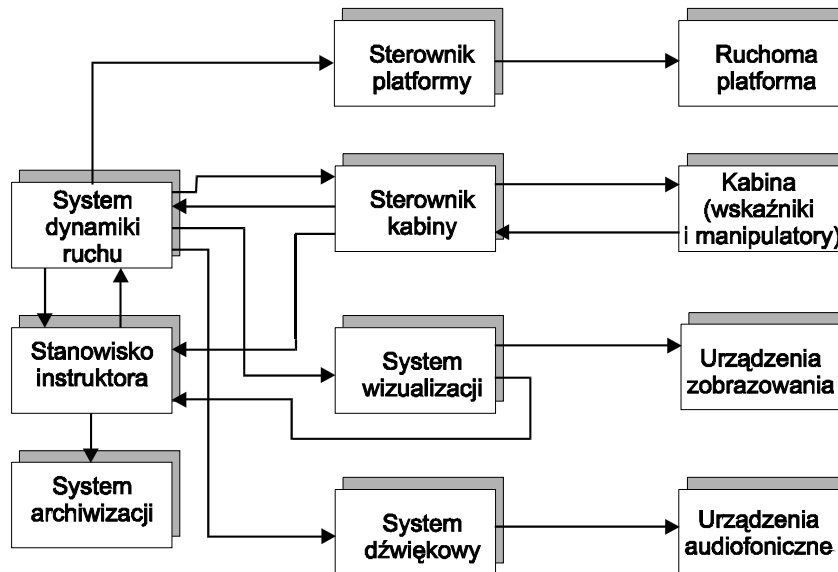
### **1. Architektura sprzętowo-programowa urządzenia symulacyjnego**

Konstruowane w ostatnich latach symulatory statków powietrznych można podzielić na trzy kategorie ze względu na zakres spełnianych przez nie funkcji:

- Symulatory kompleksowe – zapewniają symulowanie wszystkich faz lotu i funkcji statku powietrznego z uwzględnieniem oddziaływań zewnętrznych. Umożliwiają realizację prawie pełnego procesu szkolenia personelu latającego.
- Symulatory specjalizowane – ich możliwości są ograniczone do kilku wybranych funkcji i częściowo symulują rzeczywiste warunki lotu. Są przeznaczone do treningu tylko niektórych faz lotu lub manewrów.
- Symulatory proceduralne – są najmniej złożone i umożliwiają jedynie zapoznanie się z rozmieszczeniem i wykorzystaniem przyrządów

i elementów sterowania, a także trenowanie pewnych procedur związanych z realizacją określonych funkcji.

Struktura funkcjonalna symulatora kompleksowego pokazana jest na rys. 1.



Rys. 1. Schemat blokowy typowego symulatora kompleksowego

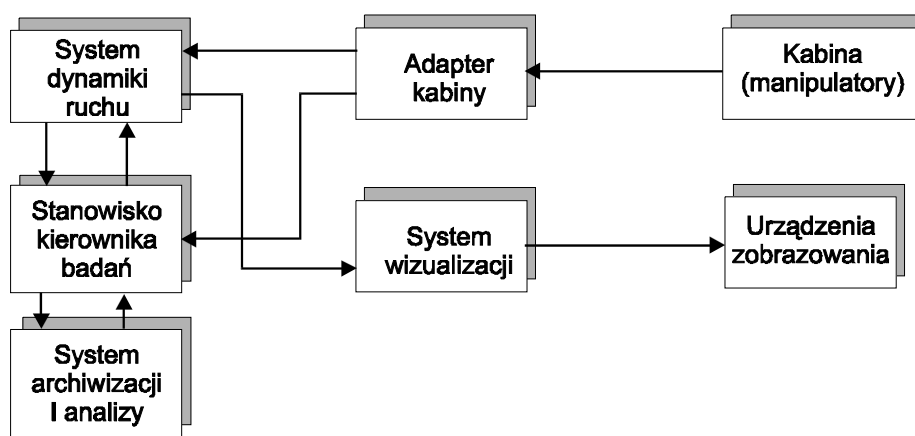
Przeprowadzona analiza wymagań i założeń wstępnych na Badawcze Urządzenie Symulacyjne (BUS) wykazała, że jego struktura funkcjonalna nie musi być tak rozbudowana, jak w klasycznym symulatorze kompleksowym. Wynika to z faktu, że podstawowym przeznaczeniem BUS nie jest szkolenie pilotów, tylko symulowanie lotów próbnych umożliwiających badanie wpływu różnych parametrów aerodynamicznych śmigłowca na jego zachowanie się podczas lotu, zwłaszcza podczas manewrów ekstremalnych. Takie założenie pozwala przede wszystkim zrezygnować z budowania ruchomej platformy dla kabiny oraz układu sterowania platformą. Sama kabina ma postać szczątkową, gdyż zawiera tylko podstawowe elementy manipulacyjne (sterownice). Zestaw podstawowych wskaźników i przyrządów pokładowych jest natomiast zobrazowany na monitorze systemu wizualizacji. BUS w podstawowej wersji nie jest również wyposażony w system dźwiękowy, chociaż dzięki otwartej architekturze będzie możliwe w przyszłości dodanie tego elementu do urządzenia.

## 1.1. Schemat blokowy

W celu zapewnienia możliwości prowadzenia wszechstronnych i wiarygodnych badań aerodynamicznych oraz wizualnej weryfikacji i analizy ich wyników BUS składa się z następujących podsystemów:

- system dynamiki ruchu,
- system wizualizacji,
- adapter kabiny,
- stanowisko kierownika badań (w miejsce stanowiska instruktora występującego w typowych symulatorach lotu),
- system archiwizacji i analizy.

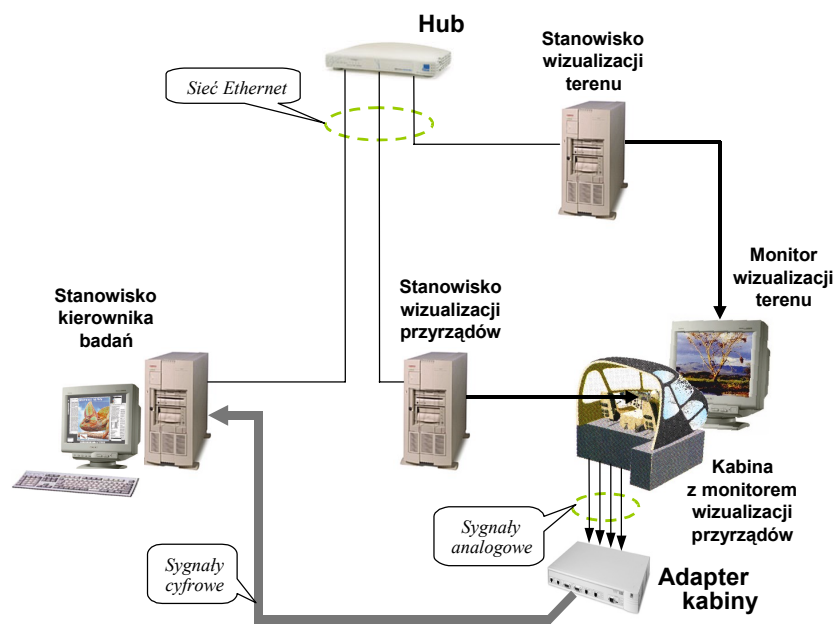
W związku z tym struktura funkcjonalna BUS przyjmuje postać, jak na rys. 2.



Rys. 2. Schemat blokowy BUS

W celu zminimalizowania kosztów realizacji projektu, a także spełnienia wymagania na otwartą architekturę całego systemu zdecydowano się na rozproszenie mocy obliczeniowej pomiędzy typowe komputery klasy PC, zamiast stosować kosztowne systemy wbudowane lub specjalizowane stacje robocze. Dzięki takiemu rozproszeniu poszczególne moduły sprzętowo-programowe są bardziej podatne na modyfikacje i rozbudowę. Dla zapewnienia odpowiednio wydajnej komunikacji pomiędzy poszczególnymi modułami została zastosowana szybka sieć komputerowa Fast Ethernet.

Zastosowana struktura sprzętowa i topologia połączeń poszczególnych komponentów BUS są przedstawione na rys. 3.



Rys. 3. Struktura sprzętowa BUS

## 1.2. Opis funkcjonalny elementów systemu

Zaprojektowany system składa się z trzech stanowisk komputerowych:

- Stanowisko Kierownika Badań (SKB),
- Stanowisko Wizualizacji Terenu (SWT),
- Stanowisko Wizualizacji Przyrządów (SWP),

oraz elementów dodatkowym takich, jak:

- kabina wraz z monitorami,
- adapter kabiny,
- infrastruktura sieci LAN.

Nadrzędną rolę w systemie pełni SKB. Realizuje ono następujące funkcje:

- odbieranie sygnałów sterujących z adaptera kabiny; sygnały te są generowane w efekcie działania manipulatorów poruszanych ręką pilota,
- odbieranie danych sterujących i poleceń od kierownika badań; np.: położenie początkowe, siła i kierunek wiatru, warunki widoczności, stany awaryjne, przerwanie symulacji, odtworzenie symulacji itp.,
- wyznaczanie kolejnych położzeń i ułożeń symulowanego statku powietrznego na drodze rozwiązywania równań różniczkowych dynamiki lotu,
- wykrywanie kolizji statku powietrznego z obiektami istniejącymi w symulowanym terenie,
- wyznaczanie stanów lub wskazań wszystkich przyrządów pokładowych, jakie są zobrazowane w kabinie,
- wysyłanie wszystkich wyliczonych wartości do stanowisk wizualizacji za pośrednictwem lokalnej sieci komputerowej,
- archiwizowanie wszystkich istotnych parametrów lotu w pamięci masowej przez cały czas trwania symulacji,
- odtwarzanie przebiegu zakończonej symulacji z możliwością zadawania prędkości odtwarzania oraz położenia obserwatora w symulowanej scenie, zobrazowania przebiegu zmienności wybranych parametrów w postaci wykresów lub tabel itp.

SWT ma za zadanie:

- przechowywać w pamięci masowej graficzną bazę danych o terenie, w którym odbywa się symulacja; baza danych zawiera informacje o ukształtowaniu terenu, jego pokryciu oraz o charakterystycznych obiektach znajdujących się na nim,
- odbierać z SKB za pośrednictwem sieci dane o aktualnym położeniu i ułożeniu statku powietrznego względem terenu, dane o warunkach widoczności oraz informację o wystąpieniu kolizji,
- generować na podstawie odebranych danych obraz terenu widziany z kabiny pilota; kolejne kadry obrazu są generowane 30 razy na sekundę metodą renderingu przy zachowaniu możliwie dużej realistyczności obrazu.

Do zadań SWP należy:

- przechowywać w pamięci oraz wyświetlać na ekranie podstawowy obraz tablicy przyrządów z kabiny symulowanego statku powietrznego,
- odbierać z SKB za pośrednictwem sieci dane o stanie i wskazaniach poszczególnych przyrządów,
- generować na podstawie odebranych danych obraz wskazań wybranych przyrządów pokładowych i wskaźników.

Pod pojęciem Kabiny należy rozumieć fotel pilota wyposażony w zestaw podstawowych manipulatorów niezbędnych do sterowania symulowanym statkiem powietrznym.

Adapter kabiny jest specjalizowanym interfejsem przetwarzającym sygnały analogowe, odbierane z manipulatorów Kabiny, na sygnały cyfrowe, przekazywane w określonym standardzie do SKB.

### 1.3. Specyfikacja sprzętowa

Stanowiska SWT i SWP zbudowane zostały na bazie identycznych zestawów komputerowych PC. Konfiguracja pojedynczego zestawu przedstawiono w tab. 1. Konfiguracja SKB przedstawiona została w tab. 2.

Tab. 1. Konfiguracja SWT i SWP

Procesor:	<i>AMD Athlon 1000MHz</i>
Płyta główna:	<i>Abit KT7A-RAID, FSB 200MHz, UDMA/100, AGP 4x</i>
RAM:	<i>256MB SDRAM/133MHz</i>
Karta graficzna:	<i>Elsa Gladiac-Ultra, GeForce2-Ultra, 64MB DDR-RAM</i>
Dysk twardy:	<i>Seagate Barracuda II, 30GB, ATA100 (7200 RPM)</i>
Karta sieciowa:	<i>3Com EtherLink 10/100Base-TX, (PCI)</i>
Monitor:	<i>Samsung SyncMaster 20Glsi, 20"</i>

Tab. 2. Konfiguracja SKB

Procesor:	<i>Intel Pentium 4 2GHz</i>
Płyta główna:	<i>FSB 333MHz, UDMA/133MHz, AGPx8</i>
RAM:	<i>512MB DDR/333MHz</i>
Karta graficzna:	<i>Leadtek WinFast GeForce2MX-400, 64MB</i>
Dysk twardy:	<i>Seagate Barracuda II, 60GB, ATA100 (7200 RPM)</i>
Karta sieciowa:	<i>3Com EtherLink 10/100Base-TX, (PCI)</i>
Monitor:	<i>Samsung SyncMaster 20GIsi, 20"</i>

#### 1.4. Specyfikacja oprogramowania

Poszczególne stanowiska komputerowe wyposażone są w składniki oprogramowania, zgodnie z zestawieniem przedstawionym w tab. 3.

Moduły oprogramowania związane z wizualizacją, komunikacją sieciową, interfejsem kierownika badań oraz archiwizacji i analizy wchodzi w skład oprogramowania systemu zobrazowania BUS i zostały opracowane przez zespół autorski.

## 2. Właściwości funkcjonalne oprogramowania systemu zobrazowania

Zasadniczym zadaniem systemu zobrazowania jest wizualizacja w reżimie czasu rzeczywistego dynamicznie zmieniającego się środowiska, które otacza pilota śmigłowca. Wyróżnić można dwa rodzaje elementów środowiska, które z punktu widzenia pilota mają fundamentalne znaczenie podczas wykonywania dowolnego zadania:

- elementy związane z terenem, nad którym odbywa się lot,
- elementy związane z różnego typu urządzeniami wskaźnikowymi umieszczonymi w kokpicie śmigłowca.

Z tego względu system zobrazowania stanowi bardzo istotną część urządzenia symulacyjnego BUS, która wpływa na jakość całego symulatora.

**Tab. 3. Konfiguracja oprogramowania na stanowiskach BUS**

<b>Stanowisko</b>	<b>SKB</b>	<b>SWT</b>	<b>SWP</b>
<b>System operacyjny</b>	<ul style="list-style-type: none"> <li>• Windows 2000 Professional</li> </ul>	<ul style="list-style-type: none"> <li>• Linux RedHat 6.2</li> </ul>	<ul style="list-style-type: none"> <li>• Linux RedHat 6.2</li> </ul>
<b>Biblioteki</b>	<ul style="list-style-type: none"> <li>• MFC</li> <li>• MAK RTI</li> </ul>	<ul style="list-style-type: none"> <li>• Open GL Performer 2.3</li> <li>• MAK RTI</li> </ul>	<ul style="list-style-type: none"> <li>• Open GL Performer 2.3</li> <li>• MAK RTI</li> </ul>
<b>Oprogramowanie narzędziowe</b>	<ul style="list-style-type: none"> <li>• Fortran Compiler</li> <li>• Visual C++</li> </ul>	<ul style="list-style-type: none"> <li>• GNU egcs</li> </ul>	<ul style="list-style-type: none"> <li>• GNU egcs</li> </ul>
<b>Moduły oprogramowania</b>	<ul style="list-style-type: none"> <li>• Moduł interfejsu kabiny</li> <li>• Moduł komunikacji sieciowej</li> <li>• Moduł symulacji ruchu</li> <li>• Moduł graficznego interfejsu kierownika badań</li> <li>• Moduł archiwizacji i analizy</li> </ul>	<ul style="list-style-type: none"> <li>• Moduł komunikacji sieciowej</li> <li>• Moduł wizualizacji terenu</li> </ul>	<ul style="list-style-type: none"> <li>• Moduł komunikacji sieciowej</li> <li>• Moduł wizualizacji przyrządów</li> </ul>

## 2.1. Parametry użytkowe systemu zobrazowania

System zobrazowania prezentuje pobrane i przetworzone dane na dwóch monitorach komputerowych. Pierwszy służy do wizualizacji terenu obserwowanego przez pilota z kabiny śmigłowca. Zapewnienie wrażenia rzeczywistego lotu system realizuje przez spełnienie następujących warunków:



- zachowanie minimalnej częstotliwości odświeżania obrazu 30 klatek na sekundę,
- dobór odpowiednich parametrów ostrośłupa widzenia, uwzględniających ograniczone możliwości obserwacji terenu znajdującego się bezpośrednio pod i nad śmigłowcem,
- przestrzeganie ograniczonego czasu odpowiedzi systemu na akcje użytkownika w celu wyeliminowania wrażenia nadmiernej bezwładności pilotowanego obiektu; czas odpowiedzi nie przekracza 50 ms,
- zastosowanie odpowiednich urządzeń projekcyjnych, w tym wypadku monitora o następujących parametrach: przekątna ekranu 20", rozdzielczość 1024x768, częstotliwość odświeżania pionowego 80Hz, plamka 0.24 mm.

Drugi z monitorów przeznaczony jest do zobrazowania wskaźników, umożliwiających pilotowi ciągle orientowanie się w bieżącej sytuacji i dających mu informacje o podstawowych parametrach lotu. Zbiór wizualizowanych wskaźników jest następujący:

- prędkościomierz barometryczny,
- wysokościomierz barometryczny
- radio-wysokościomierz,
- sztuczny horyzont,
- busola magnetyczna,
- wariometr,
- wskaźnik obrotów wirnika.

## 2.2. Dane wejściowe

Dane wejściowe dla systemu zobrazowania można podzielić na dwie grupy:

- dane o charakterze ciągłym,
- dane o charakterze dyskretnym (zdarzeniowym).

Do pierwszej grupy zalicza się wszystkie dane wyznaczone przez moduł dynamiki, niezbędne do realizacji zadań wizualizacji terenu oraz przyrządów pokładowych. Należą do nich:

- prędkość liniowa obiektu,
- prędkość kątowna obiektu,
- położenie obiektu,
- zorientowanie obiektu,
- wysokość obiektu na terenie,
- prędkość kątowna wirnika.

Drugą grupę stanowią dane, których wartości ustawiane są przez operatora SKB. Dotyczą one parametrów początkowych ćwiczenia oraz zdarzeń inicjowanych przez operatora w trakcie symulowanego lotu (zmiana pory dnia, zachmurzenia itp.).

### **2.3. Bazy danych**

System zobrazowania wymaga dostępu do dwóch rodzajów baz danych:

- baz danych opisujących tereny dla lotów ćwiczebnych,
- baz danych opisujących wskaźniki zegarowe i inne elementy z tablicy przyrządów śmigłowca.

Baza danych o terenie została przygotowana zgodnie z postawionymi wymaganiami i charakteryzuje się następującymi własnościami:

- umożliwia wybór terenu,
- zapewnia wymagany poziom szczegółowości terenu poprzez uwzględnienie obiektów naturalnych oraz będących wytworem działalności człowieka,
- zapewnia odpowiedni obszar terenu.

Baza danych o terenie zawiera modele 3D różnorodnie ukształtowanych fragmentów ziemi:

- model terenu równinnego,
- model terenu pagórkowatego,

- model terenu górzystego.

Modele zawierają obiekty naziemne w postaci budynków, słupów trakcji elektrycznej, mostów, dróg oraz różnego rodzaju roślinność, z uwzględnieniem pojedynczych drzew, krzewów, a także lasów. Każdy obiekt terenowy posiada własną reprezentację w postaci modelu trójwymiarowego z właściwym kolorowaniem i teksturowaniem.

Wielkość przygotowywanych modeli terenu jest tak dobrana, aby wykluczyć lub znacznie ograniczyć możliwość przekraczania przez śmigłowiec płaszczyzn ograniczających model terenu. Rozmiary przygotowywanego terenu zostały ustalone na podstawie:

- przewidywanych scenariuszy lotów i zadań badawczych,
- danych taktyczno-technicznych śmigłowca, m.in. maksymalnego zasięgu prowadzenia działań,
- możliwości wydajnościowych generatora obrazu,
- kosztów wytworzenia baz danych o terenie.

### **2.3.1. Baza danych przyrządów pokładowych**

Baza danych przyrządów pokładowych to zbiór modeli dwu- i trójwymiarowych przedstawiających urządzenia wskaźnikowe kabiny śmigłowca. Zobrazowane zostały najważniejsze zegary i wskaźniki wymienione w p.2.1. Modele wskaźników zostały przygotowane w sposób umożliwiający łatwe sterowanie ich ruchomymi elementami, (np. wskazówką w prędkościomierzu). Częstotliwość aktualizacji położenia elementów dynamicznych odpowiada częstotliwości odświeżania obrazu widzianego z kabiny pilota. Modele przyrządów zostały zbudowane na podstawie zdjęć rzeczywistych wskaźników śmigłowca.

### **2.3.2. Dane uzupełniające**

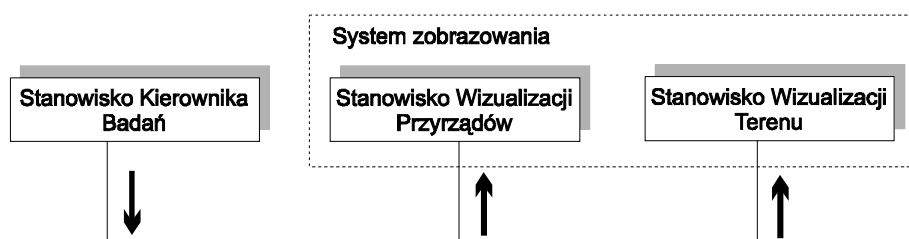
Procedury modułu dynamiki wymagają dostępu do informacji o rzeźbie terenu. W tym celu przygotowano zbiory danych charakteryzujących rzeźbę terenu w postaci umożliwiającej szybkie pozyskanie informacji na temat aktualnej wysokości obiektu nad terenem. Dane w postaci siatki punktów wysokościowych o rozdzielczości 10 metrów wygenerowano na podstawie baz

danych o terenie przygotowanych w formacie OpenFlight za pomocą oprogramowania MultiGen II Pro.

Realizacja funkcji archiwizacji i analizy lotu (patrz p.2.5.2) wymagała przygotowania trójwymiarowego modelu śmigłowca. Niektóre z opcji systemu umożliwiają bowiem usytuowanie obserwatora poza śmigłowcem tak, aby możliwe było oglądanie maszyny podczas wykonywania zadania. Jest to dodatkowa baza danych graficznych zawierająca modele 3D różnych typów śmigłowców.

## 2.4. Komunikacja

Rys. 4 przedstawia ogólny schemat przepływu informacji w systemie BUS. Dane wyznaczone w module dynamiki na SKB są przesyłane do stanowisk systemu zobrazowania, które po ich przetworzeniu, generują kolejne ramki obrazu. Cały system pracuje w reżimie czasu rzeczywistego. Częstotliwość przesyłania danych odpowiada częstotliwości ich wyznaczania w module dynamiki.



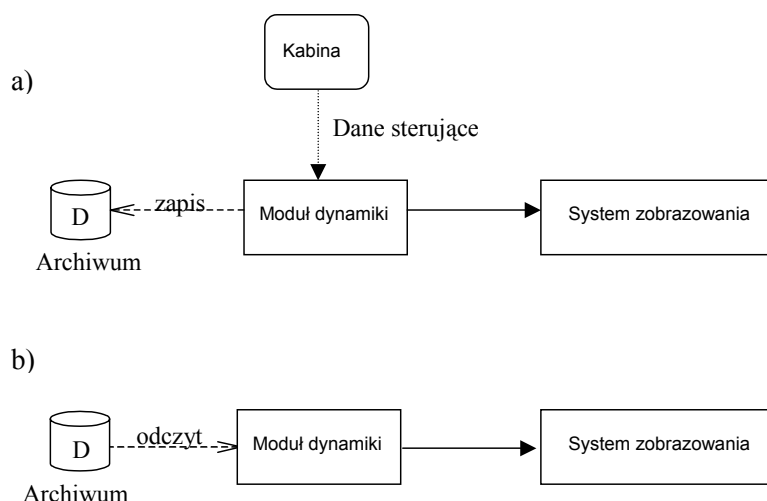
Rys. 4. Schemat przepływu danych w BUS

Stanowiska z rys.4 oparte są na oddzielnych komputerach typu PC dołączonych do wydzielonej dla nich sieci Fast Ethernet. Wymiana danych zorganizowana została zgodnie z zasadami architektury HLA (ang. *High Level Architecture*). Rozwiązanie to zapewnia:

- dostarczania niezmienionej informacji do wszystkich odbiorców,
- zachowanie prawidłowej kolejności dostarczania pakietów danych,
- kontrolę przepływu informacji,
- scentralizowane zarządzanie wszystkimi elementami systemu,
- możliwość adaptacji systemu BUS do współpracy z innymi systemami rozproszonymi opartymi na architekturze HLA.

## 2.5. Monitorowanie, archiwizacja i odtwarzanie

Funkcje monitorowania, archiwizacji oraz odtwarzania przebiegu lotu, ze względu na badawczy aspekt wykorzystania urządzenia, należą do istotnych zadań systemu. Zaimplementowane są na SKB, więc przejście systemu w tryb odtwarzania nie wymaga wykonywania dodatkowych procedur po stronie systemu zobrazowania. Generator obrazu pracuje jako element bierny, a jego jedynym zadaniem jest pobieranie i wizualizacja danych bez względu na to, w jakim trybie pracuje cały system – symulacji czy odtwarzania (rys. 5).



Rys. 5. Tryby pracy BUS: a) symulacja, b) odtwarzanie

### 2.5.1. Funkcje monitorowania i archiwizacji

Funkcje odtwarzania lotu wymagają, aby archiwum zawierało wszystkie dane wyznaczone w czasie symulacji i wysyłane do systemu zobrazowania. Proces monitorowania zapewnia dokładne śledzenie pojawiających się na wyjściu modułu dynamiki informacji, a proces archiwizacji – odpowiednie zapamiętywanie kolejnych danych w plikach binarnych. Z każdym wykonanym lotem może być związany zbiór historii, dostępny na życzenie, o teoretycznie nieskończonym „czasie życia”. Plik archiwum zawiera następujące elementy:

- nagłówek zbioru przechowujący liczbę zapamiętanych rekordów oraz parametry inicjujące określające rodzaj terenu, ustawienia środowiskowe oraz parametry początkowe lotu,
- rekordy opisujące kolejne stany symulacji, zawierające dane o locie, ewentualnie informacje o zmianie warunków środowiskowych.

### 2.5.2. Funkcje odtwarzania

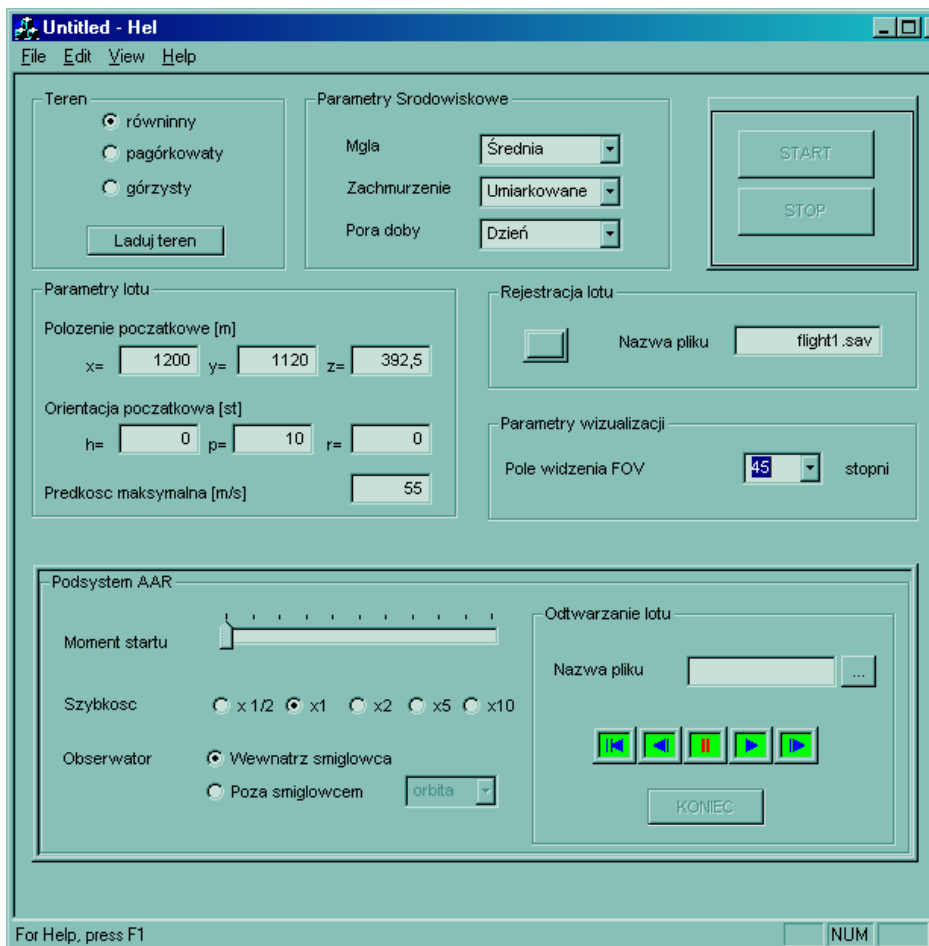
Podsystem odtwarzania ćwiczenia został zaimplementowany w celu umożliwienia przeprowadzenia dogłębnej analizy wykonania zadania. Oferuje następujące funkcje:

- Wybór pliku z nagrany przebiegiem lotu.
- Różne prędkości odtwarzania:
  - odtwarzanie w czasie rzeczywistym,
  - odtwarzanie przyspieszone z mnożnikiem 2, 5 i 10,
  - odtwarzanie spowolnione z mnożnikiem 0.5.
- Swobodne poruszanie się po osi czasu:
  - ustalanie znacznika startu w czasie szybkiego przeglądania,
  - szybkie przewijanie do przodu,
  - szybkie przewijanie do tyłu,
  - odtwarzanie wprzód lub wstecz po jednej klatce,
  - funkcje START/STOP/PAUZA.
- Wybór widoku odtwarzania z możliwością jego zmiany:
  - widok standardowy z kabiny pilota,
  - widok z pozycji obserwatora znajdującego się poza śmigłowcem:
    - obserwator podążający za obiektem (pogoń),
    - obserwator okrążający obiekt, zwrócony w stronę obiektu (spin).

### 3. Oprogramowanie systemu

Oprogramowanie BUS składa się z trzech podstawowych elementów:

- Symulatora,
- Generатора Obrazu Przyrządów (GOP),
- Generатора Obrazu Terenu (GOT).



Rys. 6. Interfejs graficzny kierownika badań.

### 3.1. Symulator

Symulator stanowi najważniejszy element systemu BUS. Za jego pomocą kierownik badań ustawia wartości parametrów symulacji, steruje jej przebiegiem oraz przeprowadza analizę badania na podstawie zarejestrowanych danych. Symulator jest programem z graficznym interfejsem użytkownika (rys. 6), za pomocą którego kierownik wprowadza niezbędne ustawienia zarówno przed rozpoczęciem badania, jak i w jego trakcie. Jądrzem symulatora jest moduł dynamiki. Jest on odpowiedzialny za wyznaczanie parametrów ruchu śmigłowca w oparciu o wprowadzane przez kierownika dane oraz aktualne położenia przyrządów manipulacyjnych zainstalowanych w kabinie symulatora. Komunikacja Symulatora z generatorami obrazów odbywa się zgodnie ze specyfikacją HLA (ang. *High Level Architecture*) określającą sposób współdziałania aplikacji (federatów) w rozproszonym zadaniu symulacyjnym (federacji). Specyfikacja ta składa się z trzech części:

- definicji interfejsu pomiędzy infrastrukturą pośredniczącą w wymianie danych RTI (ang. *Run-Time Infrastructure*) a federatami,
- szablonu OMT (ang. *Object Model Template*) określającego formę dokumentowania obiektowych modeli federacyjnych (FOM) i obiektowych modeli symulacyjnych (SOM),
- zbioru reguł, do których muszą stosować się federaci współdziałających w obrębie federacji.

W systemie BUS wykorzystany został referencyjny model federacji RPR-FOM 1.0 (ang. *Real-Time Platform Reference FOM*). Jest to model FOM opracowany przez organizację SISO na potrzeby rozproszonych środowisk symulacyjnych czasu rzeczywistego. Jego zastosowanie pozwoli w przyszłości na łączenie systemu BUS z innymi symulatorami opartymi na RPR-FOM 1.0.

Podstawową klasą obiektów w RPR-FOM jest jednostka symulacji. Symulator powołuje w federacji pojedynczą instancję tej klasy i powiadamia zainteresowane aplikacje o jej stanie. Stan jednostki opisany jest zbiorem kilkudziesięciu atrybutów, spośród których najważniejszymi są:

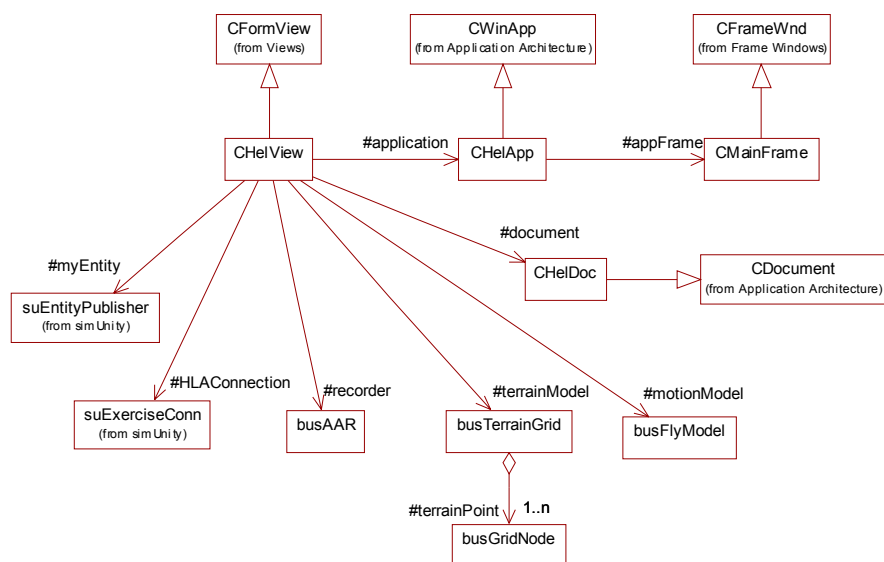
- wektory położenia, orientacji, prędkość liniowej, przyspieszenia i prędkości kątowej jednostki w układzie geocentrycznym ziemi,
- typ jednostki reprezentowany siedmioelementowym wektorem liczb całkowitych, którego składowe oznaczają kolejno: rodzaj jednostki, domenę działania, przynależność państwową, kategorię, podkategorię, specyfikację oraz identyfikator uzupełniający,



- stan jednostki (brak uszkodzeń, jednostka uszkodzona, jednostka zniszczona),
- lista ruchomych części i opisujących je parametrów (np. wirnik główny opisany parametrem prędkość obrotowa wirnika).

W modelu RPR-FOM zdefiniowane są również klasy interakcji służące do powiadamiania federatów o wystąpieniu określonych zdarzeń lub przesyłania danych nie ujętych w opisach jednostek. W systemie BUS Symulator wysyła interakcje w celu:

- zasygnalizowania kolizji śmigłowca z terenem lub inną przeszkodą,
- skonfigurowania lub dynamicznego przekonfigurowania generatorów obrazu,
- przesyłania danych o warunkach meteorologicznych,
- przesyłania aktualnej wysokości śmigłowca nad terenem (informacji takiej nie ma w opisie jednostki).



Rys. 7. Diagram klas Symulatora

Kierownik w dowolnym momencie badania ma możliwość włączenia lub wyłączenia rejestrowania jego przebiegu. Rejestracja odbywa się w pliku.

Zarejestrowane badanie może być następnie wielokrotnie odtwarzane z różną prędkością i w różnych kierunkach (do przodu, do tyłu).

Symulator zaimplementowany został w środowisku MS Visual C++ 6.0 przy pomocy biblioteki *MFC* oraz autorskiej biblioteki *simUnity* zapewniającej komunikację w rozproszonym środowisku HLA. Diagram klas przedstawiony został na rys.7. Klasy *CHelApp*, *CHelView*, *CHelDoc* i *CMainFrame* stanowią szkielet typowej aplikacji zaimplementowanej zgodnie z założeniami architektury *MFC*. Na uwagę zasługują następujące klasy:

- *suExerciseConn* – interfejs komunikacyjny federata,
- *suEntityPublisher* – publikator stanu jednostki wykorzystywany przez Symulator,
- *suEntityStateRepository* – opis stanu jednostki,
- *suArtPart* – reprezentacja ruchomej części jednostki,
- *busTerrainGrid* - model terenu w postaci regularnej siatki węzłów *busGridNode* ładowanej z pliku,
- *busFly Model* – prosty model ruchu sterowany manipulatorem ręcznym,
- *busAAR* – rejestrator i odtwarzacz przebiegu symulacji; zarejestrowane w pliku dane umożliwiają odtwarzanie przebiegu lotu w przód i w tył, z naturalną prędkością, po jednej klatce lub po 10 klatek animacji.

### 3.2. Generatory obrazu

Zadaniem generatorów obrazu jest wizualizacja scen przestrzennych w czasie rzeczywistym. Scenę przestrzenną należy rozumieć jako kompozycję obiektów trójwymiarowych, które zostały utworzone za pomocą specjalistycznego oprogramowania modelującego (np. MultiGen, AutoCAD). Dane wejściowe do procesu wizualizacji generatory otrzymują w sposób ciągły od Symulatora. Oba typy generatorów wykorzystują taki sam algorytm wizualizacji, natomiast różnice występują w sposobie interpretacji odebranych danych oraz zawartości wyświetlanych scen.

Scena Generators Obrazu Przyrządów złożona jest z obiektów reprezentujących wybrane przyrządy pokładowe oraz obiektu tablicy, w której przyrządy są osadzone. Na podstawie zawartości komunikatów GOP określa aktualne wartości wskazań przyrządów i wprowadza niezbędne modyfikacje w scenie polegające na zmianie wzajemnego położenia obiektów lub ich części. Należy zauważyć, że w przypadku GOP umowy obserwator sceny pozostaje

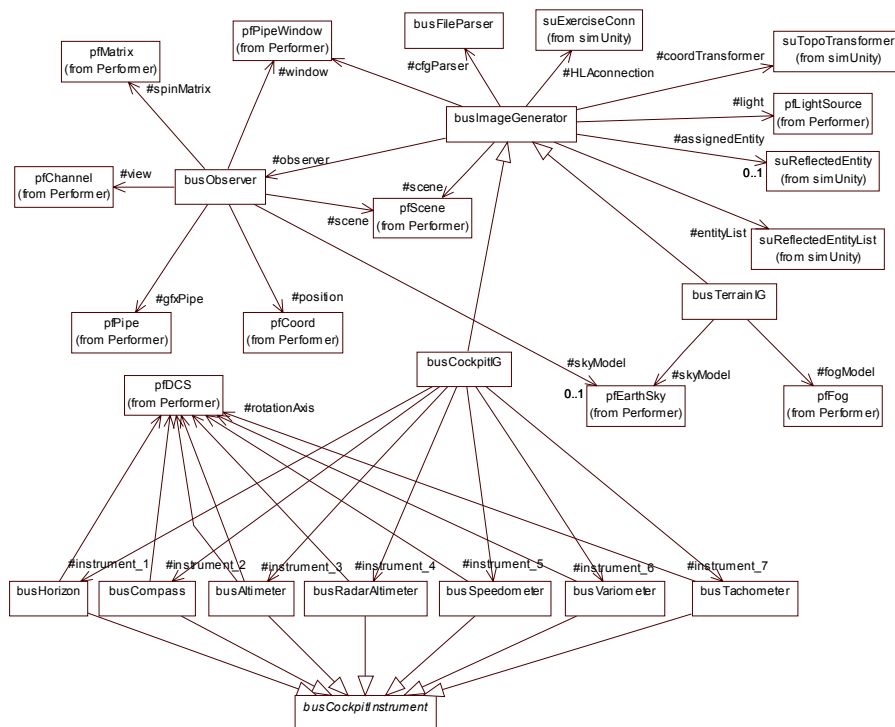
przez cały czas w tym samym położeniu, a parametry wizualizacji pozostają niezmiennie. Zadaniem Generатора Obrazu Terenu jest wizualizacja otoczenia symulowanego śmigłowca. Jego scena zawiera obiekt terenu wraz z pokryciem oraz obiekt symulowanego śmigłowca, który w niektórych trybach obserwacji jest widoczny. Położenie i orientacja obserwatora w GOT podlegają zmianom związanym z przemieszczaniem się śmigłowca. W zależności od wybranego trybu obserwacji, obserwator może znajdować się (w nawiasach podano niezbędne dane do wyznaczenia położenia i orientacji obserwatora) :

- w kabine symulowanego śmigłowca (położenie i orientacja śmigłowca),
- poza kabiną i okrążyć śmigłowca z zadaną prędkością kątową, w ustalonej odległości i wysokości względem środka masy śmigłowca (położenie i orientacja śmigłowca, prędkość kątowna, promień orbity i wysokość względem środka masy śmigłowca),
- poza kabiną i podążać za śmigłowcem w zadanej odległości (położenie i orientacja śmigłowca, liczba klatek animacji określająca opóźnienie obserwatora względem położenia śmigłowca).

Wymienione tryby obserwacji i ich parametry wyznaczone są na podstawie atrybutów symulowanej jednostki i interakcji odbieranych z Symulatora. Za pomocą interakcji przesyłane są również parametry środowiska wizualizacji, takie jak pora dnia czy gęstość mgły.

Oba generatory zaimplementowane zostały w środowisku graficznym Performer firmy SGI będącym zbiorem bibliotek funkcji i klas C++ przeznaczonych do tworzenia symulacji wizualnych działających w czasie rzeczywistym. Środowisko Performer umożliwia:

- tworzenie aplikacji wielookienkowych z możliwością wyświetlania wielu niezależnych widoków sceny w każdym oknie,
- dynamiczne budowanie i modyfikację hierarchicznego grafu sceny,
- zrównoleglenie obliczeń graficznych poprzez rozdzielenie zadań na kilka procesów działających na różnych procesorach,
- dynamiczną reakcję na przeciążenie systemu wizualizacji,
- wykorzystanie realistycznych efektów wizualnych, takich jak: mgła, chmury, punkty świetlne oraz sekwencje animacyjne (np. eksplozje).



Rys. 8. Diagram klas generatorów obrazów

Na rys. 8 przedstawiono diagram klas generatorów obrazu. Wyjaśnienia wymagają następujące klasy:

- *busImageGenerator* – bazowa klasa generatorów obrazu działających w środowisku HLA opartym na modelu federacyjnym RPR-FOM 1.0,
- *busCockpitGenerator* – specjalizacja klasy *busImageGenerator*, której instancja stanowi rdzeń aplikacji uruchamianej na SWP,
- *busTerrainGenerator* – specjalizacja klasy *busImageGenerator*, której instancja stanowi rdzeń aplikacji uruchamianej na SWT,
- *busObserver* – klasa wirtualnych obserwatorów sceny 3D, których położenie, orientacja oraz tryb obserwacji mogą być zmieniane dynamicznie,
- *busCockpitInstrument* – klasa przyrządów pokładowych,

- *busHorizon*, *busCompass*, *busAltimeter*, *busRadarAltimeter*, *busSpeedometer*, *busVariometer*, *busTachometer* – specjalizacje klasy *busCockpitInstrument* opisujące stan i zachowanie konkretnych typów przyrządów pokładowych,
- *busFileParser* – interpreter plików konfiguracyjnych,
- *pfPipe* – potok graficzny,
- *pfPipeWindow* – okno aplikacji,
- *pfChannel* – prostokątny obszar okna stanowiący pojedynczy widok sceny,
- *pfScene* – scena obiektów graficznych,
- *pfDCS*, *pfDoubleDCS* – węzeł grupujący, który posiada własną macierz przekształceń geometrycznych (pełni funkcje dynamicznego układu współrzędnych),
- *pfLightSource* – źródło światła oświetlające obiekty w scenie,
- *pfMatrix* – macierz przekształceń geometrycznych (4x4),
- *pfCoord* – współrzędne obiektu graficznego,
- *suReflectedEntity* – reprezentacja wykrytej w federacji jednostki symulacyjnej; jej stan aktualizowany jest na podstawie danych wysyłanych przez *suEntityPublisher* oraz wbudowanych modeli aproksymacyjnych (ang. *Dead Reckoning*),
- *suReflectedEntityList* – automatycznie aktualizowana lista wykrytych jednostek,
- *suTopoTransformer* – konwerter współrzędnych geocentrycznych na współrzędne topograficzne.

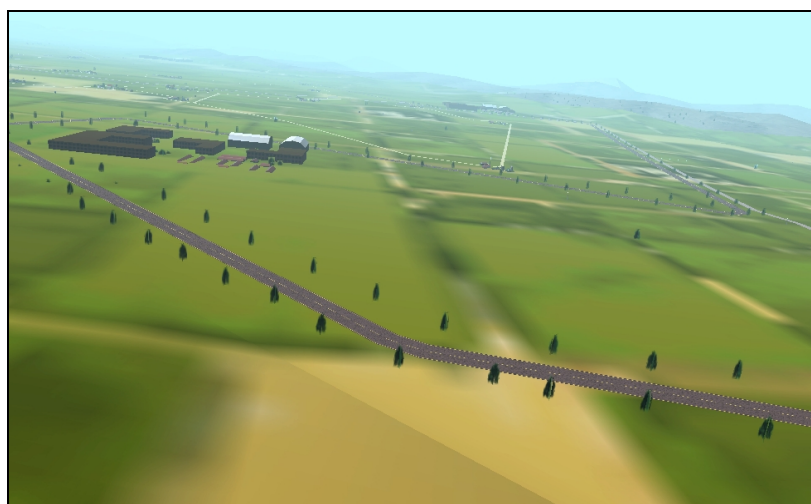
#### 4. Podsumowanie

Opisany w artykule system zobrazowania został zaimplementowany, uruchomiony i przetestowany. Na rys. 9 przedstawiono przykładowe zrzuty ekranowe ze stanowiska SWT i SWP. Na potrzeby badań system został zainstalowany na stanowisku laboratoryjnym wyposażonym w makietę kabiny śmigłowca Mi-2 i manipulatory generujące sygnały elektryczne dla karty przetworników A/C. Monitor SWP został zabudowany w miejscu klasycznej

tablicy przyrządów. Stanowisko Wizualizacji Terenu generuje obraz na monitorze umieszczonym bezpośrednio przed kabiną pilota.



a)



b)

Rys. 9. Przykładowe zrzuty ekranowe: a) SWP, b) SWT

## Literatura

- [1] Department of Defence: *High Level Architecture. Run-Time Infrastructure Programmer's Guide 1.3 Version 4*, 1998, USA.
- [2] Hartman J., Creek P.: *IRIS Performer™ Programing Guide*. Silicon Grahics, 1995, USA.
- [3] *MultiGen II User's Guide*. MultiGen Inc., 1996, USA.
- [4] Suttle C.: *Fundamentals of Real Time Modeling*. MultiGen Inc., 1997, USA.
- [5] Greuel C.: *Creating a Real-Time Database*.  
<http://www.xian.com/realtime/index.html>, 1996.

*Recenzent: prof. dr hab. inż. Marian Chudy*

*Praca wpłynęła do redakcji: 12.11.2004*

