МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНЕ НЕКОМЕРЦІЙНЕ ПІДПРИЄМСТВО
ДЕРЖАВНИЙ УНІВЕРСИТЕТ «КИЇВСЬКИЙ АВІАЦІЙНИЙ ІНСТИТУТ»
ДЕРЖАВНИЙ УНІВЕРСИТЕТ ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ

Кваліфікаційна наукова
праця на правах рукопису

КОВАЛЕНКО ЮЛІЯ БОРИСІВНА

УДК 004.415.2:629.7.05

ДИСЕРТАЦІЯ

ІНФОРМАЦІЙНА ПІДТРИМКА ПРОЦЕСУ ПРОЕКТУВАННЯ ТА ЕКСПЛУАТАЦІЇ КОМПЛЕКСУ БОРТОВОГО ОБЛАДНАННЯ ІНТЕГРОВАНОЇ МОДУЛЬНОЇ АВІОНІКИ

Спеціальність 05.13.06 «Інформаційні технології»

Подається на здобуття наукового ступеня
доктора технічних наук

Дисертація містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело

Юлія КОВАЛЕНКО

Київ – 2025

АНОТАЦІЯ

Коваленко Ю.Б. Інформаційна підтримка процесу проєктування та експлуатації комплексу бортового обладнання інтегрованої модульної авіоніки. – Кваліфікаційна наукова праця на правах рукопису. Дисертація на здобуття наукового ступеня доктора технічних наук за спеціальністю 05.13.06 – «Інформаційні технології». – Державне некомерційне підприємство Державний університет «Київський авіаційний інститут»., Київ, 2025.

Дисертація присвячена вирішенню важливого науково-прикладного завдання, що має технічну спрямованість у галузі інформаційних технологій – розробці моделей, методів та інформаційної технології інформаційної підтримки процесу проєктування та експлуатації комплексу бортового обладнання, побудованого на основі принципів інтегрованої модульної авіоніки (IMA).

Актуальність теми обумовлена наявністю протиріч між:

– високими вимогами до надійності та відмовостійкості бортового обладнання сучасних повітряних суден і обмеженими можливостями традиційних підходів до проєктування та супроводу IMA в умовах складної багатофакторної експлуатації;

– існуючими засобами інформаційної підтримки, які обмежені у своїй функціональності, та необхідністю створення цільової інформаційної підтримки для систем нового покоління, що вимагають використання сучасних технологій і стандартів для забезпечення гнучкості, масштабованості й автоматизації;

– необхідністю гармонізації базових засобів інформаційної підтримки процесів проєктування та експлуатації бортового радіоелектронного обладнання з засадами інтегрованої модульної авіоніки, при цьому висока модульність і гнучкість системи призводить до збільшення складності, вартості та часу на сертифікацію, що негативно впливає на її надійність і продуктивність.

Недостатній рівень інтеграції моделей на етапах проєктування і технічної експлуатації призводить до зниження ефективності процесів діагностики, прогнозування стану та прийняття рішень щодо технічного обслуговування.

Уперше розроблено метод підтримки процесу проєктування ІМА, наукова новизна якого полягає у формалізації визначення множини параметрів функцій бортового обладнання на ранніх стадіях розробки. Запропоновано математичну модель проєктування комплексу бортового обладнання, що забезпечує підвищення його надійності та відмовостійкості завдяки узагальненню функцій верхнього рівня та використанню стаціонарного марківського процесу.

Вперше побудовано графову модель структури функцій комплексу бортового обладнання, яка дозволяє здійснити розрахунок обчислювальних ресурсів, визначити рівень гарантії проєктування та оцінити навантаження на мережу передачі даних. Запропоновано структурну модель проєктування складних автоматизованих систем управління ІМА з урахуванням віртуальних з’єднань, AFDX- та CAN-мережевих технологій, що підвищують ефективність виявлення та ізоляції несправностей.

Вперше запропоновано структурно-логічну модель уніфікованого автоматизованого робочого місця для контролю обчислювальної системи ІМА, що забезпечує підвищення достовірності діагностики шляхом мажорування результатів контролю різними модулями.

Удосконалено інформаційну технологію інформаційної підтримки, яка об'єднує графові, структурні та динамічні моделі у єдиний підхід до підтримки прийняття рішень, діагностики та прогнозування стану технічних систем ІМА. Реалізація запропонованих рішень дозволяє скоротити тривалість та вартість розробки, зменшити кількість помилок та підвищити ефективність функціонування бортового обладнання в умовах експлуатації.

Проведені експериментальні дослідження підтвердили достовірність теоретичних положень та практичних розробок дисертаційного дослідження.

Результати дослідження впроваджені в діяльність ТОВ «СІТОН ДІДЖИТАЛ», а також використовуються в навчальному процесі кафедри телекомунікаційних та радіоелектронних систем Державного некомерційного підприємства Державний університет «Київський авіаційний інститут».

Результати роботи дисертанта знайшли відображення у таких звітах держбюджетних НДР № 5-2024/18.02 «Система забезпечення кібербезпеки та стійкості об’єктів критичної інфраструктури». Термін роботи: 01.03.2024р. – 30.06.2025р. Державного некомерційного підприємства «Державний університет «Київський авіаційний інститут».

Ключові слова: інформаційна технологія, моделі та методи, автоматизовані системи управління, підтримка прийняття рішень, надійність технічних систем, архітектура обчислювальних систем, інтелектуальна діагностика, інтегрована модульна авіоніка.

ABSTRACT

Kovalenko Y.B. Information support for the design and operation of the integrated modular avionics onboard equipment complex. – Qualification research paper as a manuscript.

Dissertation for the degree of Doctor of Technical Sciences in the specialty 05.13.06 – "Information Technologies". – State University "Kyiv Aviation Institute", Kyiv, 2025.

The dissertation considers an important applied scientific problem of a technical nature in the field of information technologies – the development of models, methods and information technologies to support the processes of designing and operating an on-board equipment complex based on the principles of integrated modular avionics (IMA).

The relevance of the topic is determined by the existence of contradictions between:

– high requirements for reliability and fault tolerance of modern aircraft on-board equipment and limited capabilities of traditional approaches to the design

and support of on-board radio-electronic equipment (AEA) in complex multi-factor operating conditions;

– existing information support tools, which are limited in functionality, and the need to develop targeted information support for next-generation systems that require modern technologies and standards to ensure flexibility, scalability and automation;

– the need to harmonize the main information support tools for the design and operation of on-board radio-electronic equipment with the principles of AEA, while high modularity and flexibility of the system lead to an increase in complexity, cost and time of certification, which negatively affects reliability and productivity.

Insufficient integration of models at the stages of design and technical operation reduces the efficiency of diagnostic processes, condition prediction and decision-making in technical maintenance.

A method for supporting the AEA design process has been developed for the first time. Its scientific novelty lies in the formalization of the definition of parameter sets for on-board equipment functions at the early stages of development. A mathematical model for the design of an on-board equipment complex is proposed, which increases reliability and fault tolerance due to the generalization of high-level functions and the use of a stationary Markov process.

A graph model of the functional structure of an on-board equipment complex is developed for the first time. This model allows for the assessment of computing resources, the determination of the level of design assurance, and the assessment of the load on the data transmission network. A structural model for the design of complex automated IMA control systems is proposed, which takes into account virtual connections, AFDX and CAN network technologies, which increases the efficiency of fault detection and isolation.

A structural-logical model of a single automated workplace for controlling an IMA computing system is proposed for the first time, which increases the

reliability of diagnostics by voting on the majority of results from different modules.

Information support technology has been improved by integrating graph, structural and dynamic models into a unified approach to decision support, diagnostics and forecasting of the state of IMA technical systems. The implementation of the proposed solutions allows to reduce the time and cost of development, minimize errors and improve the performance of on-board equipment in operational conditions.

Experimental studies have confirmed the validity of the theoretical provisions and practical developments of the dissertation research.

The results of the study have been implemented in the activities of LLC «SITON DIGITAL», and are also used in the educational process of the Department of Telecommunications and Radioelectronic Systems of the State Non-profit Enterprise "Kyiv Aviation Institute of the State University".

The results of the dissertation are reflected in the reports of the state-funded research project No. 5-2024/18.02 «Cybersecurity and resilience system of critical infrastructure facilities», which was implemented from March 1, 2024 to June 30, 2025 by the State Non-Profit Enterprise «Kyiv Aviation Institute of the State University»..

Keywords: information technology, models and methods, automated control systems, decision support, reliability of technical systems, architecture of computing systems, intelligent diagnostics, integrated modular avionics.

СПИСОК ПУБЛІКАЦІЙ ЗДОБУВАЧА

Наукові праці, в яких опубліковані основні наукові результати дисертації:

1. Y. Kovalenko, «Methods of developing integrated modular avionics systems», Scientific Journal of Polonia University, no. 43(6), pp. 324-334, 2020.

2. Y. Kovalenko, I. Kozlyuk, «Designing of technological processes of decision-making support information system in integrated modular avionics». Problems of Informatization and Management, no. 64, pp. 4-15, 2020.

3. I. Kozlyuk, Y. Kovalenko, «Functional bases of the software development and operation in avionics». Problems of Informatization and Management, no. 63, pp. 49-63, 2020.2

4. Y. Kovalenko, «Experiment test of the reliability of computer systems of integrated modular avionics». Problems of Informatizati on and Management, no. 65, pp. 45-53, 2021.5

5. Y. Kovalenko, « Methodology For Testing Languages For Embedded Avionics Systems ». Problems of Informatizati on and Management, no. 63, pp. 44 – 52 , 2022.1

6. H. Konakhovych, I. Kozlyuk, Y. Kovalenko, «Specificity of optimization of performance indicators of technical operation and updating of radio electronic systems of aircraft», System research and information technologies, no. 3, pp. 41-54, 2020.

7. I. Kozlyuk, Y. Kovalenko, «Reliability of computer structures of integrated modular aviation for hardware configurations», System research and information technologies, no. 3, pp. 84-94, 2021.

8. Y. Kovalenko, I. Kozlyuk, «Implementation of the integrated modular avionics application development complex according to the ARINC653 standard», The Bulletin of Zaporizhzhia National University: Physical and mathematical Sciences, no. 2, pp28-36, 2020.

9. Y. Kovalenko, I. Kozlyuk, «Functional methods of developing integrated modular avionics systems», The Bulletin of Zaporizhzhia National University: Physical and mathematical Sciences, no. 1, pp101-116., 2021.

10. Y. Kovalenko, I. Kozlyuk «Методи оцінки показників надійності авіаційної техніки за результатами випробувань та технічної діагностики», Наукоємні технології, № 3(39), С. 367-375, 2018.

11. V. Lakhno, S. Kazmirchuk, Y. Kovalenko, L. Myrutenko, T. Zhmurko, «Design of adaptive system of detection of cyber-attacks, based on the model of logical procedures and the coverage matrices of features», Eastern-European Journal of Enterprise Technologies, no. 3(9), pp. 30-38, 2016.

12. Ю. Коваленко, «Модель захищеної автоматизованої системи управління на підприємстві», Вісник Інженерної академії України, № 2, С. 68-72, 2017.

13. Ю. Коваленко, «Система тестування на проникнення комп'ютерних мереж та систем», Вісник Інженерної академії України, № 1, С. 167-170, 2015.

14. Ю. Коваленко, «Системний аналіз проблеми багатокритеріального вибору варіанту удосконалення системи захисту інформації», Захист інформації, Т. 17, № 1, С. 38-43, 2015.

15. Ю. Коваленко, «Комплексний підхід до управління корпоративним інформаційним контентом», Безпека інформації, Т. 21, № 1, С.70-76, 2015.

16. В. Козачок, Ю. Коваленко, «Особливості побудови комплексних систем захисту інформації в розподілених корпоративних мережах», Сучасний захист інформації, № 1, С. 41-47, 2015.

17. Ю. Коваленко, «Захишенний застосунок на основі технології VOIP», Вісник Інженерної академії України, № 2, С. 27-30, 2015.

18. А. Корченко, С. Казмирчук, А. Гололобов, Ю. Коваленко, «Метод инкрементирования порядка лингвистических переменных для систем анализа и оценивания риска», Захист інформації, Т. 17, №2, С. 101-111, 2015.

19. Ю. Коваленко, Н. Вишневська, І. Мішина, А. Дзюбаненко, «Структурний аналіз мов програмування захищених застосунків для програмованих логічних контролерів», Безпека інформації, Т. 22, № 1, С. 38-43, 2016.

20. Ю. Коваленко, «Нейросетевые технологии принятия решений управления сложными социотехническими системами (ССТС) в реальном времени, Вимірювальна та обчислювальна техніка в технологічних процесах, № 1, С. 95-100, 2016.

21. Ю. Коваленко, «Комп'ютеризована система захищеного застосування на основі технології VOIP», Одеса: Збірник наукових праць ОДАТРЯ, С. 91-95, 2016.

22. Ю. Коваленко, «Метод захисту інформації в комп'ютерних мережах промислових підприємств», Вісник Інженерної академії України, № 4, С. 93-97, 2016.

23. Kovalenko, Y., «A programmable logic controller (PLC); programming language structural analysis», Advances in Intelligent Systems and Computingthis link is disabled, С.234–242, 2017.

24. Ю. Коваленко, Л. Рибалка, М. Бурлака, «Вдосконалення автоматизованої системи управління технологічної підготовки виробництва в авіаційній галузі», Вісник хмельницького національного університету. Технічні науки, № 4, С. 209-214, 2018.

25. Ю. Коваленко, Л. Рибалка, М. Бурлака, «Аналіз надійності функціонування програмного забезпечення інформаційно-управляючих систем», Вимірювальна та обчислювальна техніка в технологічних процесах, № 4, С. 74-81, 2017.

26. Ю. Коваленко, Козлюк І., «Проектування технологічних процесів інформаційної системи підтримки прийняття рішень», Збірник наукових праць Військового інституту Київського національного університету імені Тараса Шевченка, № 61, С. 14-23., 2018.

27. Ю. Коваленко, Козлюк І., «Структурний аналіз технологічного проектування і управління технологічним проектом у авіаційній галузі», Збірник наукових праць Військового інституту Київського національного університету імені Тараса Шевченка, № 59, С. 22-31, 2018.

28. Kovalenko Yu. B., Kozlyuk I. O. Method of developing automated information support for integrated modular avionics systems: monitoring, diagnostics and reliability improvement // Visnyk of the National Aviation University. – 2024. – Vol. 80. – [40-52pp.]. DOI: https://doi.org/10.18372/2073-4751.80.19768.

29. Shmatko O., Herasymov S., Milevskyi S., Balitskyi N., Pohasii S., Aleksiiev M., Vlasov I., Melenti Y., Kovalenko Yu., Peleshok Y. Development of a method for assessing the efficiency of technical systems' computer dynamic simulators // Eastern-European Journal of Enterprise Technologies. – 2025. – Vol. 2, No. 9 (134). – P. 50–64. – DOI: https://doi.org/10.15587/1729-4061.2025.327558.

30. Benzar A., Kovalenko Yu., Taranenko A., Balynska O., Balynskyi I. Contemporary strategies for managing organisational information security // Journal of Information Systems Engineering & Management. – 2024. – Vol. 10, No. 12(s). 505-513 pp. – DOI: https://doi.org/10.52783/jisem.v10i12s.1859.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

ПС	–	повітряне судно
ОС	–	обчислювальна система
IMA	–	ана модульна авіоніка
ARINC	–	Aeronautical Radio Incorporated (авіаційний радіоелектронний комітет)
ПЗ	–	програмне забезпечення
ASAAC	–	Allied Standard Avionics Architecture Council (об’єднаний комітет по стандартизації архітектури авіаційних комплексів)
БРЕО	–	бортове радіоелектронне обладнання
DAIS	–	Distributed Avionics Information System (розподілена авіонічна інформаційна система)
ІД	–	ідентифікатор
MASA	–	Modular Avionics System Architecture (архітектура модульної авіоніки)
DS	–	відмова в обслуговуванні (Denial of service)
ISS	–	Integrated Sensor System (інтегрована сенсорна система)
SEM	–	Standard Electronic Module (стандартний електронний модуль)
AAAP	–	Advanced Avionics Architecture and Packaging (просунута архітектура та компоновка авіоніки)
ЕО	–	електронне обладнання
КБО	–	комплекс бортового обладнання
ІТ	–	інформаційні технології
DFBW	–	Digital Fly-By-Wire (електродистанційна система керування польотом)
ILS	–	Instrument Landing System (інструментальна система посадки)
DO-178C	–	Design Objectives 178C (стандарт сертифікації авіаційного програмного забезпечення
DO-254	–	Design Objectives 254 (стандарт сертифікації апаратного забезпечення для авіації)
ARP 4761	–	Aerospace Recommended Practice 4761(аналіз безпеки для авіоніки)
ARP 4754A	–	Aerospace Recommended Practice 4754A (процеси розробки комплексів авіаційних систем)
DO-331	–	Design Objectives 331 (додаток до DO-178C для моделювання)
DO-332	–	Des ign Objectives 332 (додаток до DO-178C для об’єктно-орієнтованих технологій)
DO-333	–	Design Objectives 333 (додаток до DO-178C для формальних методів)
DO-330	–	Design Objectives 330 (вимоги до інструментів сертифікації ПЗ)
DO-248C	–	Design Objectives 248C (додаткові роз’яснення до DO-178C)
DO-278A	–	Design Objectives 278A (вимоги до ПЗ для об'єктів наземної інфраструктури авіації)
HLR	–	High-Level Requirements – вимоги високого рівня

LLR – Low-Level Requirements – вимоги низького рівня
PSAC – Plan for Software Aspects of Certification – план сертифікації аспектів програмного забезпечення
SDP – Software Development Plan – план розробки програмного забезпечення
SVP – Software Verification Plan – план перевірки програмного забезпечення
SCMP – Software Configuration Management Plan – план керування конфігурацією програмного забезпечення
SQAP – Software Quality Assurance Plan – план забезпечення якості рограмного забезпечення
SDS – Software Design Standard – стандарт проєктування програмного забезпечення
SRS – Software Requirements Specification – специфікація вимог до програмного забезпечення
SCS – Software Coding Standard – стандарт кодування програмного забезпечення
АЗ – апаратне забезпечення
ОМ – обчислювальний модуль
ТФН – таблиця функцій несправностей
РГП – рівень групової пріоритетності
ГФ – група функцій
БД – база даних
СУБД – система управління базами даних
ДКР – дослідно-конструкторська розробка
САПР – система автоматизованого проєктування
CAD – Computer-Aided Design – автоматизоване проєктування
CAE – Computer-Aided Engineering – інженерний аналіз
PDM – Product Data Management – управління інженерною документацією
GIS – Geographic Information System – геоінформаційна система
EDA – Electronic Design Automation – САПР для електроніки
CFD – Computational Fluid Dynamics – обчислювальна гідродинаміка
MES – Mechanical Event Simulation – симуляція механічних подій
СКД – система контролю і діагностики
БОС – базова операційна система
ОК – обчислювальний комплекс
ЗП – запам’ятовуючий пристрій
ВМК – внутрішній магістральний канал
МЗп – магістраль запису
МЧт – магістраль читання
МА – адресна магістраль
СП – спецобчислювач
ОчП – обчислювальний пристрій
ПО – пристрій обміну

ПК	–	пристрій керування (каналом)
ДЗП	–	довготривалий запам’ятовуючий пристрій
ОЗП	–	оперативний запам’ятовуючий пристрій
КРС	–	коробка розподільча системи
ПОУ	–	пульт оперативного управління
СО	–	спеціальний обчислювач
ФК	–	фільтр Калмана
АФК	–	адаптивний фільтр Калмана
НФК	–	нелінійний фільтр Калмана
НАФК	–	нелінійний адаптивний фільтр Калмана
ІНС	–	інерціальна навігаційна система
GPS	–	Global Positioning System – глобальна система позиціонування
МГУА	–	метод групового урахування аргументів
ДЕС	–	динамічна експертна система
ГА	–	генетичний алгоритм
CLIPS	–	C Language Integrated Production System – експертна система, побудована на базі мови С
ІС	–	інформаційна система
UML		Unified Modeling Language (уніфікована мова моделювання)
MBD		Model-Based Design (модельно-орієнтоване проектування)
RTOS		Real-Time Operating System (операційна система реального часу)
ARINC 653		Aeronautical Radio, Incorporated Standard 653 (стандарт на розділення часу у системах авіоніки)
XtratuM		реалізація гіпервізора з розділенням часу (ARINC 653-сумісна платформа)
UG		Unigraphics (нині NX Siemens PLM)
ADA		структурована мова програмування для систем реального часу
MECANO		система моделювання кінематики і динаміки механічних систем
OrCAD		Oriented Computer-Aided Design (система електронного моделювання схем)
GENESYS		General Engineering System Simulation (інструмент системного моделювання)
АСОНІКА-Т		система аналізу теплових процесів (українська розробка)
Sauna		програмний інструмент для теплового моделювання
BETAsoft		система інженерного аналізу (теплові та електричні процеси)
Simulink		графічне середовище моделювання та симуляції від MathWorks
SolidWorks		програма для тривимірного САПР
Mentor Graphics		компанія-розробник програмного забезпечення САПР

ВСТУП

Обґрунтування вибору теми дослідження. На сучасному етапі розвитку вітчизняна авіаційна промисловість підходить до створення повітряних суден (ПС) нового покоління. Ключова роль у створенні ПС нового покоління відводиться створенню нової універсальної обчислювальної системи (ОС). Універсальна ОС входить до складу ряду авіаційних комплексів (пілотажний, навігаційний та ін.), які керують рухом ПС в польоті, і розробляється відповідно до концепції, що отримала назву «інтегрована модульна авіоніка» (IMA), докладно викладеної в групі стандартів ARINC 651-655.

Стандарти ARINC 651-655а регламентують порядок розробки компонентів апаратного і програмного забезпечення (ПЗ), з яких в подальшому будується універсальна ОС для авіаційного застосування. Реалізація запропонованих в стандартах ідей передбачає надання ОС IMA:

якісно нових експлуатаційних властивостей;

підвищених значень техніко-економічних показників в порівнянні з існуючими сьогодні обчислювальними системами, що знаходяться в експлуатації.

Практичний досвід розробки ОС нового покоління і перспективних ОС IMA показав суттєві відмінності в принципах їх структурної організації та виявив об'єктивну потребу сучасного виробництва в створенні нових методів та засобів автоматичного та автоматизованого контролю, спеціалізованих під обчислювальні системи IMA і забезпечують властивість реконфігурації ОС при виникненні відмов у польоті.

Прийняті в ОС нового покоління технічні рішення в області організації засобів контролю виявляються непридатними для створення засобів контролю перспективних ОС, в зв'язку з чим актуальною є проблема створення і дослідження методів і засобів контролю ОС, спеціалізованих під ОС IMA.

На тлі успіхів перспективних зарубіжних досліджень компаній Astronautics Corporation of America, Aerospace Display Systems Inc., Display and Technologies interface Product, Planar Advance be, Systran Corporation, Thales, Honeywell, Rockwell Collins, Allied Signal та інших, що проводяться під егідою об'єднаного комітету по стандартизації архітектури авіаційних комплексів ASAAC (Allied Stand Avionics Architecture Council) в рамках програм створення в США нових концепцій побудови БРЕО – DAIS, MASA (Module Avionics System Architecture), інтегрованих сістемтчіков – ISS (Integrated Sensor System), стандартних конструкцій електронних модулів-SEM (Standard Electronic Module), і AAAP (Advanced Avionics Architecture And Packaging) у Великій Британії.

В Україні ведеться пошук нових наукових напрямів розвитку БРЕО, дослідження наукових основ проектування та управління якістю проектних робіт, методології побудови і функціонування ЕО, здатних забезпечити якісний і довготривалий пріоритет вітчизняних розробок в цільовій ефективності та конкурентоспроможної боротьбі.

Детальне опрацювання методів і засобів проектування, оптимальна комбінація – сукупності функціональних систем в єдиний комплекс вимагає створення наскрізної технології автоматизованого синтезу, що забезпечує зниження невизначеності в оцінці основних властивостей розроблюваних систем до рівня, що дозволяє здійснити обґрунтований вибір найкращого варіанту структури і параметрів БРЕО, а також окремих видів його підсистем вже на початковому етапі – етапі ескізного проектування.

Зазначена проблема обумовлюється об'єктивним протиріччям, тобто, науково-технічна проблема пов’язана з гармонізацією базових засобів інформаційної підтримки процесів проектування та експлуатації бортового обладнання з засадами інтегрованої модульної авіоніки.

Зв'язок роботи з науковими програмами, планами, темами

Дисертаційну роботу виконано відповідно до планів наукової і науково-технічної діяльності Державного некомерційного підприємства

«Державний університет «Київський авіаційний інститут» та в межах виконання науково-дослідної роботи № 5-2024/18.02 «Система забезпечення кібербезпеки та стійкості об’єктів критичної інфраструктури» (термін виконання: 01.03.2024 – 30.06.2025).

Тематика дослідження відповідає основним науковим напрямам і важливим проблемам фундаментальних досліджень у галузі природничих і технічних наук, затвердженим постановою Кабінету Міністрів України від 11 січня 2018 р. № 13. Результати дослідження також узгоджуються з положеннями Стратегії розвитку вітчизняної авіаційної промисловості на період до 2020 року (розпорядження КМУ від 27 грудня 2008 р. № 1656) та Державної цільової науково-технічної програми розвитку авіаційної промисловості на 2021—2030 роки (постанова КМУ від 01 вересня 2021 р. № 951).

Результати дисертаційного дослідження впроваджено у навчальний процес кафедри телекомунікаційних та радіоелектронних систем Державного некомерційного підприємства «Державний університет «Київський авіаційний інститут» для підготовки здобувачів ступеня бакалавра та доктора філософії за спеціальністю 172 «Електронні комунікації та радіотехніка» (акт від 20.05.2024 р.). Результати інтегровані до викладання дисциплін:

– «Методи забезпечення надійності та ефективності експлуатації сучасних телекомунікаційних та радіотехнічних систем»;

– «Основи теорії надійності, експлуатації та ремонту інформаційно-телекомунікаційних та радіотехнічних систем»;

А також у діяльність товариства з обмеженою відповідальністю «СТОПН ДІДЖИТАЛ», зокрема шляхом практичного застосування результатів дисертаційного дослідження при розробленні апаратно-програмного комплексу підтримки авіаційних систем (акт впровадження від 15.04.2025 р.).

Мета і завдання дослідження. Метою дисертаційної роботи полягає в підвищенні ефективності інформаційної підтримки процесів проектування та експлуатації бортового обладнання гармонізованих з засадами інтегрованої модульної авіоніки.

Для досягнення наведеної мети в рамках цієї роботи розглянуті наступні завдання:

1. Проаналізувати сучасні засоби інформаційної підтримки процесу проектування та експлуатації комплексу бортового обладнання.

2. Розробити метод підтримки процесу проектування інтегрованої модульної авіоніки для формалізації процесу визначення множини параметрів побудови комплексів бортового обладнання нового покоління.

3. Розробити модель процесу проектування комплексу бортового обладнання інтегрованої модульної авіоніки для побудови математичних методів підвищення надійності та відмовостійкості на етапі проектування.

4. Розробити графову модель структури функцій комплексу бортового обладнання інтегрованої модульної авіоніки вхідних і вихідних параметрів для реалізації обчислювальної потужності, навантаження на мережу передачі даних.

5. Розробити структурну модель процесу проектування складних автоматизованих систем управління комплексу бортового обладнання для вирішення проблеми виявлення несправностей автоматизованих систем, встановити взаємозв'язок щодо виявлення аналогічностей та появи помилок такого типу, повідомити та сформулювати звіт.

6. Розробити структурно-логічну модель уніфікованого автоматизованого робочого місця для підвищення достовірність контролю обчислювальної системи інтегрованої модульної авіоніки.

7. Розробити методи оцінки ймовірності безвідмовної роботи структурно-логічної моделі уніфікованого автоматизованого робочого місця для контролю функціональних елементів обчислювальної інтегрованої модульної авіоніки на виробництві.

8. Розробити інформаційну технологію для підвищення відмовостійкості бортового обладнання.

9. Розробити алгоритм контролю обчислювальної системи інтегрованої модульної авіоніки під час польоту повітряного судна для контролю функціональних елементів та подальшої розробки програмного забезпечення у складі уніфікованого робочого місця обчислювальної системи інтегрованої модульної авіоніки.

10. Розробити, експериментально дослідити та впровадити в практику алгоритмічне, програмне забезпечення інформаційної підтримки процесів проектування та експлуатації бортового обладнання інтегрованої модульної авіоніки для верифікації створених методів та моделей.

У процесі вирішення наведених задач використані результати робіт у сфері статистичної обробки даних, теорії та практики експлуатації технічних комплексів багатьох вітчизняних і закордонних вчених, перш за все А. Вальда, М. Де Гроота, М.М. Фішмана, Ю.Г. Сосуліна, А.Н. Ширяєва, А. Тартаковського, І.В. Нікіфорова, Г. Дейвіда, І.А. Ібрагімова, Р.З. Хасьминського, Ш. Закса, Є.Ю. Барзіловіча, В.А. Каштанова, Б.Р. Левіна, Ю.К. Бєляєва, С.О. Дмитрієва, В.П. Харченка, В.С. Дем'янчука, Г.Ф. Конаховича, І.О Козлюк,.І.О. Мачаліна, В.В. Коніна, В.О. Ігнатова, В.С. Новікова, О.Л. Петрашевського, О.І. Запорожця, В.В. Уланського, О.Г. Байбуза, В.Г. Мелкумяна та інших.

Об'єктом дослідження є процес проектування та експлуатації комплексу бортового обладнання інтегрованої модульної авіоніки.

Предметом дослідження є засоби інформаційної підтримки процесу проектування та експлуатації комплексу бортового обладнання.

Методи дослідження ґрунтуються на системному аналізі результатів сучасних теоретичних і прикладних розробок українських і зарубіжних вчених у галузі інформаційних технологій, автоматизованих систем управління та авіоніки. Під час виконання дослідження використано методи системного аналізу та теорії управління для формалізації процесу

проєктування ІМА, логіко-імовірнісні та статистичні методи для оцінки безвідмовності й відмовостійкості елементів обчислювальної системи, методи дискретної математики та графової теорії для побудови структурно-функціональних моделей, методи теорії обробки спостережень для аналізу результатів експериментів, методи аналітичного та математичного моделювання для створення моделей процесів, а також методи програмної інженерії для реалізації інформаційної технології підтримки проєктування та експлуатації бортового обладнання.

Наукова новизна отриманих результатів полягає в наступному:

Розроблено концепцію інформаційної підтримки процесу проектування та експлуатації комплексу бортового обладнання

1. Удосконалено метод підтримки процесу проектування інтегрованої модульної авіоніки, в якому за рахунок виокремлення етапів: ескізного проєктування, формуванні вимог процесу оцінки безпеки, верифікації процесу оцінки безпеки та виділення окремої проектної процедури генерування структури функцій комплексу бортового обладнання, дозволило формалізувати процес визначення множини параметрів для побудови комплексів бортового обладнання нового покоління.

2. Удосконалено модель процесу проектування комплексу бортового обладнання інтегрованої модульної авіоніки, в якій за рахунок узагальнених функцій верхнього рівня та первинних даних про їх структуру, окремої процедури процесу проектування формалізації та типізації структури функцій, на основі Марківських процесів дозволяє підвищити відмовостійкість на ранніх стадіях проектування.

3. Вперше розроблено графову модель структури функцій комплексу бортового обладнання, яка, використовуючи теорію графів, дозволяє пов’язати множини вхідних і вихідних параметрів функцій модулів та їх груп, визначити необхідну кількість обчислювальних модулів, рівні гарантії проєктування кожного модуля, а також обсяг і характер даних, що передаються мережею в межах обчислювальної інфраструктури ІМА.

4. Вперше розроблена структурна модель процесу проектування складних автоматизованих систем управління комплексу бортового обладнання інтегрованої модульної авіоніки, в якій за рахунок модернізації спеціалізованих контролерів універсальними модулями забезпечення незалежної роботи різних авіаційних систем, та інтеграції підсистеми віртуальних з'єднань внутрішньої комутованої мережевої інфраструктури, при підтримці технологій, AFDX (Avionics Full DupleX Switched Ethernet) і CAN (Controller area network), на основі теорії ймовірності, дозволило обґрунтувати основні етапи проектування так експлуатації та компоненти автоматизованих систем управління бортового обладнання, підвищити ймовірність виявлення несправностей автоматизованих систем.

5. Вперше розроблена структурно-логічну модель уніфікованого автоматизованого робочого місця, в якій за рахунок застосування модульного принципу розділення модулів-тестів за функціональною залежністю кожного тестуючого елемента, та процедури мажорування результатів контролю кожного функціонального елемента обчислювальної системи, дозволяє підвищувати достовірність контролю обчислювальної системи інтегрованої модульної авіоніки.

6. Вперше розроблено метод оцінки ймовірності безвідмовної роботи структурно-логічної моделі уніфікованого автоматизованого робочого місця обчислювальної системи інтегрованої модульної авіоніки, в якій за рахунок аналітичних процедур реалізації процесів резервування на рівні однотипних функціональних модулів та окремої процедури ковзного резервування на рівні ідентичних функціональних модулів контролю функціональних елементів обчислювальної інтегрованої модульної авіоніки, дозволило підвищити у сукупності достовірність контролю обчислювальної системи інтегрованої модульної авіоніки.

7. Вперше розроблено інформаційну технологію, в якій за рахунок методу підтримки процесу проектування інтегрованої модульної авіоніки, моделей процесу проектування та графової моделі структури функцій

комплексу бортового обладнання, та враховуючи структурну модель процесу проектування складних автоматизованих систем управління комплексу бортового обладнання інтегрованої модульної авіоніки, з врахуванням методів прогнозування, апріорно-математичної моделі редукованої динамічної експертної системи, дозволила підвищити відмовостійкість та надійність бортового обладнання інтегрованої модульної авіоніки.

Практичне значення отриманих результатів. Отримані в дисертаційній роботі результати можуть бути використані для побудови та оптимізації структури функцій комплексу бортового обладнання, дозволяють ефективно провести роботи по формуванню структури комплексу на ранніх стадіях процесу проєктування, що в результаті призводить до: зниження кількості помилок на ранніх стадіях проєктування; зниження трудовитрат на усунення помилок, які виникають на ранніх стадіях проєктування; підвищення ефективності виконання робіт в рамках ескізного етапу проєктування; скорочення загального часу проєктування комплексу бортового обладнання; зниження вартості процесу проєктування.

На основі запропонованої інформаційної технології, розроблено комплекс алгоритмів, що дозволяють обчислювати прогнозовані значення похибок систем авіоніки бортового обладнання та запобігають виходу з діапазону стійкої роботи, а також забезпечують працездатність і відмовостійкість обладнання повітряних суден;

На основі запропонованої структурно-логічної моделі уніфікованого автоматизованого робочого місця, розроблено апаратно-програмний комплекс, що використовується при розробці відповідне програмне забезпечення, для реалізації алгоритму контролю функціональних елементів операційної системи інтегрованої модульної авіоніки.

Розроблено апаратно-програмний комплекс для контролю і діагностики функціональних елементів обчислювальної системи інтегрованої модульної авіоніки, що функціонують в нормальних і спеціальних умовах експлуатації,

що дає можливість реалізувати програмне забезпечення, інформаційно-вимірювального та алгоритмічного забезпечення.

В цілому, результати дисертаційного дослідження є підґрунтям для формування та реалізації своєчасних та вірних запобіжних та корегувальних дій персоналом експлуатаційних підрозділів наземного забезпечення авіації (НЗА) та органами державного регулювання у сфері транспорту. Результати досліджень упроваджені в навчальному та науковому процесах Державного некомерційного підприємства «Державний університет «Київський авіаційний інститут», що підтверджено відповідними актами впровадження.

Особистий внесок здобувача.

Основні положення та результати дисертаційної роботи, що виносяться на захист, отримані автором самостійно. У роботах, що написані у співавторстві, автору належать: [1-11, 13-16, 18, 20-28] – результати які отримані особисто; [2] – постановка задачі дослідження та аналіз інформаційної підтримки процесу проєктування та експлуатації БРЕО; [8-9] – розробка методів і моделей підтримки процесу проектування інтегрованої модульної авіоніки комплексів бортового обладнання нового покоління; [16-18] – запропоновано інформаційну технологію для підвищення відмовостійкості бортового обладнання; [26-30] – розроблено алгоритм контролю обчислювальної системи інтегрованої модульної авіоніки під час польоту повітряного судна для контролю функціональних елементів та подальшої розробки програмного забезпечення у складі уніфікованого робочого місця обчислювальної системи інтегрованої модульної авіоніки. З робіт, опублікованих у співавторстві, для вирішення проблеми та задач, поставлених у дисертаційному досліджені, використовуються результати отримані особисто здобувачем наукового ступеня.

Апробація результатів дисертації

Основні результати дисертаційних досліджень доповідалися й обговорювалися на таких конференціях і семінарах: Міжнародна науково-практична конференція молодих учених і студентів «Політ. Сучасні

проблеми науки» (Київ, НАУ, 2013 – 2025 рр.); Науково-практична конференція «Актуальні питання забезпечення кібернетичної безпеки та захисту інформації» (с. Верхнє Студене, Європейський університет, 2018 р.); Міжнародна науково-технічна конференція «ITSEC" (Київ, НАУ, 2013 – 2023 рр.); Міжнародна науково-технічна конференція "АВІА» (Київ, НАУ, 2013 – 2019 рр.); Міжнародна науково-практична конференція «Комплексне забезпечення якості технологічних процесів та систем» (Чернігів, Національний університет «Чернігівська політехніка» 2017 – 2021 рр.); Міжнародна науково-практична конференція" Інтегровані інтелектуальні робототехнічні комплекси (ІІРТК)" (Київ, НАУ, 2015 – 2019 рр.); Науково-практичний семінар "Технічні засоби захисту інформації" (Київ, Національна Академія Наук України, 2017 – 2021рр.). Ось перелік дяких з них:

Коваленко Ю.Б Новітні технології управління корпоративним інформаційним контентом. АВІА-2015 : XII міжнар. наук.-техн. конф., 28-29 квітня 2015 р.: тези доп. – К., 2015. – Т.1. – С. 121–122.

Коваленко Ю.Б проблема підбору варіантів удосконалення проектованої системи фаззінгу з використанням генетичних алгоритмів. Інтегровані інтелектуальні робототехнічні комплекси (ІІРТК-2015): VIII міжнар. наук.-практ. конф., 18-19 травня 2015р.: тези доп. – К., 2016. – С. 77–80.

Коваленко Ю.Б Політ. Сучасні проблеми науки: XV міжнар. наук.-практ. конф. молод. учених і студ., 8-9 квітня 2015р.:тези доп. – К., 2015. – С. 112.

Коваленко Ю.Б Методи виявлення уразливостей міжсайтового скриптінгу та SQL-ін’єкції. Матеріали першої всеукраїнської науково-практичної конференції «Перспективні направлення захисту інформації». – Одеса, 2016. – С. 39-43.

Коваленко Ю.Б Міждисциплінарні підходи в процесі дослідження віртуальних спільнот у соціальних мережах Актуальні питання забезпечення кібербезпеки та захисту інформації: ІІ міжнар. наук. конф., 24-27 лютого 2018. – тези доп.: – К., 2016. – С. 74-76.

Коваленко Ю.Б Системний аналіз комбінованого аналізатора бінарного коду bitblaze. Комплексне забезпечення якості технологічних процесів та систем: VIII міжнар. Наук.-техн. Конф.,26-29 квітня 2018 р.: тези доп. – Чернігів., 2016. – С. 30

Коваленко Ю.Б Структурна схема системи elite keylogger. Приладобудування:стані перспективи: XVII міжнар. Наук.-техн. Конф.,17-19 травня 2016 р.: тези доп. – К., 2019. – С. 86-87. Коваленко Ю.Б Організація безпеки інформаційних ресурсів корпоративних мереж. Міжвідомчий міжрегіональний семінар Наукової Ради НАН України.«Технічні засоби захисту інформації».: тези доп.- – К., 22.12.2016р. – С.43.

Kovalenko Yuliia, Kovalchuk Olena. Zero Trust Security: Treats to Information Security in the Context of National Security // International and National Security: Theoretical and Applied Aspects: Proceedings of the 9th International Scientific and Practical Conference (Dnipro, March 14, 2025). – Dnipro: Dnipropetrovsk State University of Internal Affairs, 2025. – [P. 41–43].

Публікації. Результати дисертаційного дослідження опубліковано у 41 науковій праці. Із них 30 статей – опубліковано у наукових журналах, з яких: 25 статті [2–6, 8–10, 12–22, 24–28] – у наукових фахових виданнях України, що включені до переліку, затвердженого МОН України; 5 статей [7, 11, 23, 29, 30] – у виданнях, що індексуються у міжнародній наукометричній базі даних Scopus. За матеріалами доповідей на науково-технічних та науково-практичних конференціях опубліковано 11 публікацій.

Структура та обсяг роботи. Дисертаційна робота складається зі вступу, 5 розділів, висновків, списку використаних джерел (130 найменувань на 14 сторінках), 2 додатків (на 8 сторінках). Основний текст роботи викладено на 300 сторінках, рисунків – 58, таблиць – 28. Загальний обсяг роботи становить 320 сторінок.

РОЗДІЛ 1. ДОСЛІДЖЕННЯ ПРОЦЕСУ ІНФОРМАЦІЙНОЇ ПІДТРИМКМ ПРОЕКТУВАННЯ ТА ЕКСПЛУАТАЦІЇ КОМПЛЕКСУ БОРТОВОГО ОБЛАДНАННЯ

Умови експлуатації сучасних ПС характеризуються високими швидкостями, великими висотами та дальністю польоту, дією різноманітних зовнішніх збурень. В таких умовах ймовірність відмов систем зростає. Тому необхідно жорстко контролювати характеристики КБО і окремих систем авіоніки. Для цього застосовуються різні системи контролю на різних етапах експлуатації КБО.

Бортове радіоелектронне обладнання, що використовуються на ПС, виконує безліч складних функцій в польоті. А от же, їх працездатність потрібно відстежувати бажано в кожен момент часу. Використовувані засоби для контролю та діагностики, де системи попередніх поколінь не можуть бути використані для контролю та діагностики перспективних обчислювальних систем. Провідною задачею систем проектування та експлуатації є інформаційна підтримка високого рівня заданого показника ефективності використання обладнання за призначенням. Тому метою цього розділу є аналіз існуючих засобів інформаційної підтримки проектування та експлуатації БРЕО аеронавігаційного обслуговування України та постановка задачі дослідження під час проектування та модернізації систем експлуатації.

Для досягнення поставленої мети в цьому розділі були розглянуті такі основні задачі: проаналізовано сучасні засоби інформаційної підтримки процесу проектування та експлуатації комплексу бортового обладнання; виконано аналіз радіоелектронного обладнання, що використовується в цивільній авіації України, а також визначені його основні визначальні параметри та показники надійності; проаналізовано наукові результати у сфері інформаційної підтримки та технічних систем проектування та експлуатації, виконано аналіз нормативно-правової документації проектування та експлуатації; виконано постановку задач дисертаційного дослідження.

1.1 Класифікація моделей і методів інформаційної підтримки процесів проектування та експлуатації повітряних суден.

Сучасні повітряні судна, оснащені різними радіоелектронними бортовими пристроями, які допомагають пілоту виконувати складні повітряні завдання. Стрімкий розвиток цифрової електроніки та інформаційних технологій, що стає все більш виразним протягом останніх років, вимагає від пілота сучасного літака підтримки в польоті бортового обладнання, яке, по суті, вже є «розумними» комп’ютерами зі складною та комплексною авіонікою програмного забезпечення. Така комп’ютеризація сучасного ПС має забезпечити необхідну його експлуатаційну надійність і безпеку як екіпажу, так і пасажирів [1–5].

Починаючи з 1970-х років, науково-дослідницька програми з цифровим керуванням польотами досліджувала заміну механічних первинних систем керування польотом на електронні системи за допомогою електричних сигналів.

Boeing 747, Lockheed L-1011 і Douglas DC-10 використовували різні реалізації для забезпечення функцій автопосадки, які вимагали ймовірності відмови під час приземлення. 747 використовував аналогові комп’ютери з потрійним резервуванням. L-1011 використовував цифрові комп'ютери в подвійній архітектурі, а DC-10 використовував два ідентичних канали, кожен з яких складався з подвійного резервного аналогового комп'ютера з відключенням і відключенням для кожної осі.

З того часу Airbus A-320 використовує штатну систему управління польотом DFBW. Він використовує різноманітність проектування програмного забезпечення для захисту від збоїв загального режиму. Архітектура комп'ютера управління польотом Boeing 777 використовує матрицю 3 на 3 з 9 процесорів 3 різних типів. Також використовується багатоверсійне програмне забезпечення.

Процес виробництва цифрових бортових пристроїв і систем забезпечує їх надійну роботу в усіх критичних для них умовах польоту, зокрема через

зміни тиску і температури навколишнього повітря та виникнення лінійних перевантажень, наприклад, під час маневрового польоту. Збій програмного забезпечення системи посадки ILS під час заходу літака на посадку в аеропорт у складних метеоумовах .

Щоб задовольнити суворі вимоги щодо надійності в процесі розробки електронного повітряного обладнання та спеціального програмного забезпечення для авіоніки, спеціалісти склали відповідні стандарти, такі як стандарт DO-178C, що містить вимоги до програмного забезпечення, та стандарт DO-254, що містить апаратне забезпечення. вимоги, підкріплені додатковими документами стандартизації, включаючи ARP 4761 і ARP 4754A (рис. 1.1).

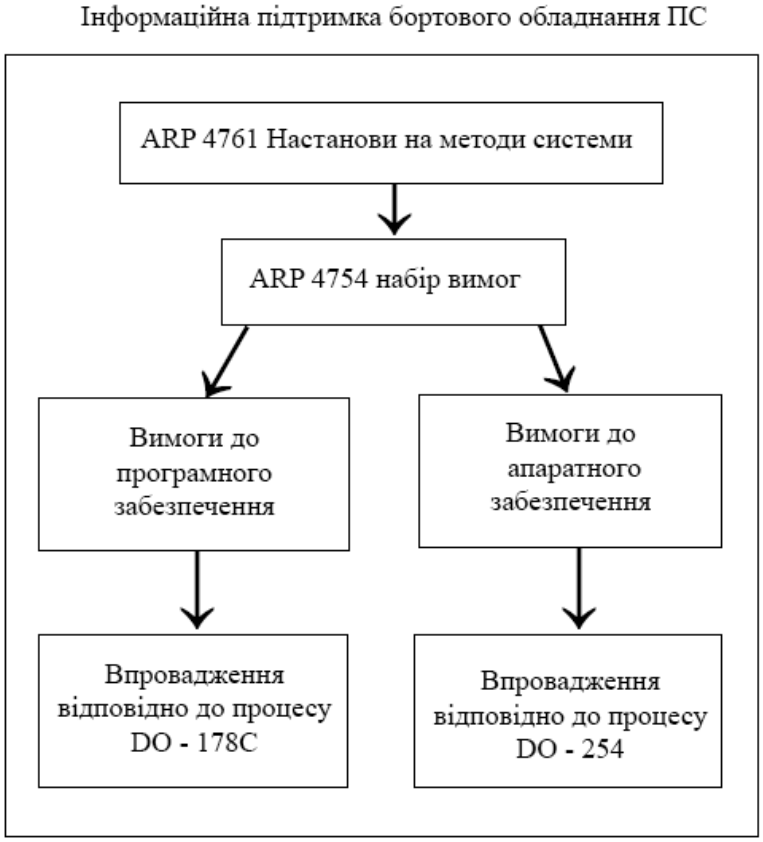


Рис. 1.1 Інформаційна підтримка процесу проектування авіоніки та програмного забезпечення, критичного для безпеки польотів, відповідно до стандарту DO-178C.

Стандарт ARP 4761 — це набір процесів і методів, які використовуються для забезпечення необхідного рівня надійності

розроблених пристроїв і систем; призначений для встановлення на борту ПС. Цей стандарт визначає п'ять рівнів пошкодження бортового обладнання та їх наслідки [6]. Рівень А охоплює збитки, виникнення яких може призвести до серйозних жертв і матеріальних втрат. Рівень В покриває значні збитки, які можуть призвести до серйозних травм. Рівень С охоплює значні пошкодження нижчого ступеня опромінення, що означає відмову основних бортових систем, але дозволяється продовження польоту . Рівень D покриває незначні пошкодження, наприклад, коли дозволяється продовжувати політ без певної функції. Рівень Е охоплює пошкодження, виникнення яких є незначним для процесу продовження польоту та дій пілота. Стандарт ARP 4754A — це набір функціональних системних вимог щодо програмного та апаратного забезпечення.

Стандарт DO-178C є основним документом, що застосовується на міжнародному авіаційному ринку до процесу виробництва та сертифікації програмного забезпечення бортових електронних пристроїв. Він складається з базової частини та додаткових стандартів DO-331, DO-332 та DO-333, а також DO-330, DO-248C та DO-278A, які визначають додаткові вимоги та дії, які мають бути передбачені програмним забезпеченням. процесу розробки в залежності від прийнятої методики реалізації (рис. 1.2).

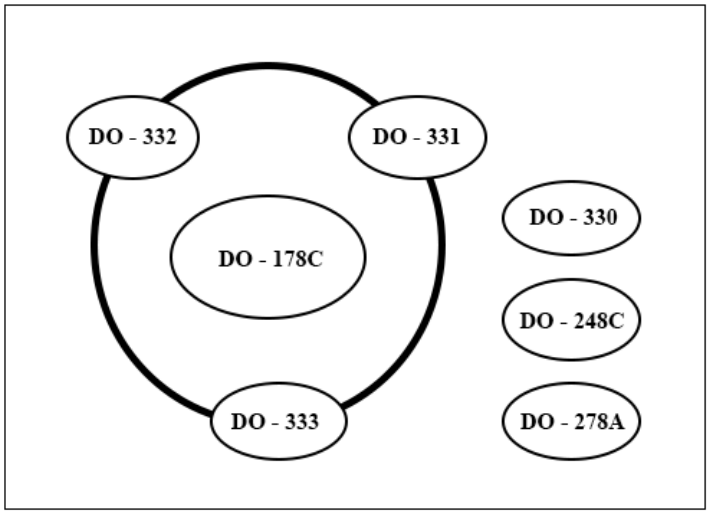


Рис. 1.2. Модель взаємозв’язків між додатковими стандартами, які визначають допоміжні вимоги та види діяльності відповідно до стандарту DO-178C.

Стандарт DO-331 описує розробки програмного та апаратного забезпечення з використанням техніки моделювання (проектування на основі моделі). Обчислювальні алгоритми або їх фрагменти реалізуються за допомогою зовнішніх засобів, а процес проектування виконується шляхом об’єднання блоків, які виконують різноманітні математичні операції. Реалізована таким чином обчислювальна діаграма підлягає тестуванню та перевірці. Заключним етапом реалізації алгоритму є автоматичне формування діаграми на основі розробленого вихідного коду програмного забезпечення. При використанні цього методу обчислювальні алгоритми реалізуються візуально з автоматичною генерацією результуючого програмного коду.

Стандарт DO-332 застосовується при реалізації проектів з використанням об'єктно-орієнтованих мов програмування (включаючи C++, Java, Ada). Фундаментальним питанням у випадку об’єктно-орієнтованих мов є перевірка такого програмного забезпечення, яка дотримується трьох основних методів, а саме: успадкування, поліморфізму та динамічного зв’язування. Втіленням такого стандарту може бути наявність ще одного методу з такою ж назвою в межах тієї самої ієрархії класів, що гарантує, що під час виклику відповідного класу виконується відповідний метод.

Стандарт DO-333 застосовується, коли в процесі проектування використовуються формальні методи. Формальні методи – це математичні прийоми, які застосовуються в процесі розробки та перевірки програмного забезпечення. Завдання формальних методів полягає в тому, щоб розробити математичну систему для побудованої системи та перевірити її поведінку, щоб довести, що система, що будується, є функціональною та задовольняє передбачуваним вимогам безпеки. Стандарт DO-333 описує процес кваліфікації програмних засобів, які використовуються в процесі розробки, тестування, запуску та модифікації програмного забезпечення. Процес кваліфікації розуміється як оцінка програмних засобів, результати яких не перевіряються і які спрощують, прискорюють і автоматизують стандартний

процес DO-178C.

Стандарт DO-248C є додатковим і додатковим документом до стандарту DO-178C, що пояснює його незрозумілі та проблемні аспекти.

Вищезазначені вимоги стандарту DO-178C стали основою для побудови комп’ютерної системи, що підтримує управління життєвим циклом програмного забезпечення авіоніки.

Життєвий цикл програмного забезпечення авіоніки пов’язаний із виконанням численних різноманітних типів взаємопов’язаних завдань, метою яких є проектування та розробка програмного забезпечення відповідної якості, що відповідає вимогам замовника, а також забезпечення необхідного рівня безпеки розроблені авіонічні системи та пристрої [1–9].

Стандарт DO-178C визначає три основні проекти, а саме планування, розробку та інтеграцію програмного забезпечення, які є взаємно паралельними у вибраних областях. Це означає, що початок одного процесу необов’язково пов’язувати із завершенням попереднього. Прикладом може бути початок тестування закодованого програмного фрагмента до етапу повного завершення кодування .

Існує багато моделей життєвого циклу програмного забезпечення, які включають каскадну модель. У цій моделі розрізняють наступні основні кроки, які необхідно виконати, щоб відповідати вимогам стандарту DO178C. Вони передбачають збір вимог і їх аналіз, а також проектування, розробку, тестування та впровадження програмного забезпечення.

Кожен із цих кроків дозволяє нам повернутися до попереднього стану, забезпечуючи взаємодію між виконуваними завданнями, що особливо важливо у разі виявлення програмних помилок, які необхідно виправити. Впровадження розробленого програмного забезпечення авіоніки також охоплює його експлуатацію до моменту утилізації.

Планування програмного забезпечення включає процеси, пов’язані з плануванням і стандартами, після яких слідує розробка програмного забезпечення. На цьому етапі стандарт DO-178C визначає п'ять основних

етапів і три стандарти програмного забезпечення. Він також виділяє інші документи, які необхідні для сертифікації будь-якого розробленого програмного забезпечення [18–20]. Основні етапи програмного забезпечення авіоніки такі:

– Етап сертифікації аспектів програмного забезпечення (PSAC): свого роду «контракт» між підрядником і органом сертифікації.

– Етап розробки програмного забезпечення (SDP): містить вимоги щодо планування програмного забезпечення, кодування та етапів інтеграції. SDP був написаний для розробників програмного забезпечення і є свого роду керівництвом щодо розробки програмного забезпечення, щоб воно відповідало прийнятим вимогам.

– Етап перевірки програмного забезпечення (SVP): містить аспекти, пов’язані з перевіркою функціональності програмного забезпечення, і призначений для тестувальників програмного забезпечення. SVP асоціюється з SDP, тому що на цьому етапі перевіряються припущення, які були продумані на етапі розробки програмного забезпечення.

– Етап керування конфігурацією програмного забезпечення (SCMP): визначає процедури, інструменти та методи, спрямовані на досягнення цілей, пов’язаних із керуванням вимогами протягом усього життєвого циклу програмного забезпечення. Він охоплює процедури з точки зору визначення базової версії та ідентифікації версій програмного забезпечення, звітування про проблеми, контролю та перегляду модифікацій, архівування, контролю завантаження та відновлення програмного забезпечення.

– Етап забезпечення якості програмного забезпечення (SQAP): визначає процедури та методи, які мають застосовуватися, щоб задовольнити вимоги до якості, пов’язані зі стандартом DO-178C. Він визначає процедури з точки зору управління якістю, виконання аудиту, дії, пов’язані зі звітуванням про проблему, і методологію коригувальних дій.

– Основні стандарти розробки програмного забезпечення БРЕО:

– Стандарти вимог до програмного забезпечення (SRS), які визначають принципи, методи та інструменти, що застосовуються для розробки вимог високого рівня. Вони включають методи, що використовуються для розробки програмного забезпечення, позначення для реалізації вимог (алгоритми, блок-схеми), обмеження інструментів проекту, які будуть використовуватися для розробки програмного забезпечення, а також критерії, пов’язані з вимогами.

– Стандарти розробки програмного забезпечення (SDS): вони визначають методи, інструменти та обмеження в процесі розробки програмного забезпечення. Вони включають вимоги низького рівня та архітектуру програмного забезпечення. Вони призначені для команди розробників програмного забезпечення та пояснюють, як ефективно реалізувати проект. Вони охоплюють, серед іншого, методи номенклатури, обмеження інструментів проектування та складність програмного забезпечення (наприклад, тривалість процедури).

– Стандарти програмного кодування (SCS); Вони визначають методи, інструменти та обмеження в процесі кодування програмного забезпечення. Вони включають, серед іншого, стандарти кодування, використовувані мови програмування, стандарти представлення коду, стандарти номенклатури, обмеження компілятора та обмеження, що випливають із стандартів програмування.

– Вищезазначені документи, як еталонні шаблони, реалізовані в рамках побудованої комп’ютерної системи, яка підтримує процес управління розробка та сертифікація програмного забезпечення авіоніки.

– Процес розробки програмного забезпечення авіоніки складається з чотирьох детальних процесів:

– Процес вимог до програмного забезпечення, який призводить до розробки вимог високого рівня (HLR);

– Процес проектування програмного забезпечення, який розробляє вимоги низького рівня (LLR) і архітектуру програмного забезпечення на основі HLR;

– Процес кодування, який призводить до створення вихідного коду та неінтегрованого об'єктного коду;

– Процес інтеграції програмного забезпечення, який передбачає консолідацію програмного забезпечення у формі виконуваних програм та його інтеграцію із зовнішніми пристроями.

Вимоги високого рівня (HLR) реалізуються на основі архітектури системи та системних вимог. Вони включають часові сигнали, керування пам'яттю, заплановані зв'язки із зовнішніми пристроями, методи реагування та виявлення помилок, моніторинг роботи системи та розділення програмного забезпечення. HLR є основою для розробки вимог низького рівня, які використовуються в процесі проектування програмного забезпечення, які включають описи зв’язків із зовнішніми пристроями, визначення та спосіб потоку даних, механізми зв’язку та компоненти програмного забезпечення.

Процес кодування включає переклад LLR у вихідний код і попередньо скомпільований об’єктний код. Це пов'язано з процесом верифікації, оскільки на цьому етапі відбувається попереднє виконання частково розробленого коду. Процес інтеграції передбачає компіляцію та об’єднання скомпільованого коду у виконувані програми і вбудовування цього програмного забезпечення на цільову платформу.

Процеси розробки програмного забезпечення визначають один або багато рівнів системних вимог. Вимоги високого рівня визначаються безпосередньо на основі архітектури системи та системних вимог. Вони розробляються в процесі проектування програмного забезпечення, таким чином створюючи взаємопов’язані вимоги низького рівня. Однак, коли вихідний код створюється безпосередньо на основі вимог високого рівня, він також відповідає вимогам низького рівня та підлягає рекомендаціям щодо

вимог низького рівня. Процеси вимог до програмного забезпечення використовують вихідні дані процесу життєвого циклу програмного забезпечення для створення вимог високого рівня. Основним результатом цього процесу є дані про вимоги до програмного забезпечення.

Дані про вимоги до програмного забезпечення визначають вимоги високого рівня, включаючи вимоги, надані замовником. Дані повинні включати опис розподілу вимог до програмної системи, враховуючи вимоги безпеки та потенційні умови помилок, функціональні та робочі вимоги для кожного режиму роботи, критерії продуктивності (наприклад, точність і точність), вимоги та обмеження, пов’язані з часом, обмеження розміру пам’яті, апаратні та програмні інтерфейси (наприклад, протоколи, формати, частота вводу/виводу), виявлення помилок, моніторинг безпеки, а також вимоги до розділення програмного забезпечення (як окремі компоненти програмного забезпечення взаємодіють) і рівні програмного забезпечення для кожного компонента.

Вхідні дані в процес розробки програмного забезпечення авіоніки включають вимоги до програмного забезпечення, план розробки програмного забезпечення та стандарти розробки програмного забезпечення. Якщо заплановані критерії переходу виконуються, вимоги високого рівня використовуються в процесі виробництва для створення архітектури програмного забезпечення та вимоги низького рівня. Вони можуть включати один або декілька рівнів вимог.

Основним результатом цих процесів є опис проекту, який містить архітектуру програмного забезпечення та вимоги низького рівня. Дані повинні включати детальний опис того, як програмне забезпечення задовольняє вимоги високого рівня, включаючи алгоритми та структуру даних, і як вимоги до програмного забезпечення відповідають процесам і завданням. Він також повинен надавати опис архітектури програмного забезпечення, що визначає структуру програмного забезпечення, із реалізованими вимогами, описами вводу/виводу (наприклад, словник даних,

дані та потік керування в рамках проекту), обмеження ресурсів, стратегію управління ресурсами та їх обмеженнями, а також запаси як а також методи вимірювання цих запасів. Наприклад, час і пам’ять, процедури послідовності, опис має включати внутрішньопроцесорні та внутрішньозадачні механізми зв’язку, включаючи фіксовані часові послідовності переривання, методи проектування та деталі їх реалізації, наприклад завантаження програмного забезпечення. Важливим елементом опису є програмне забезпечення, модифіковане користувачем, методи розділення та заходи щодо запобігання злому розділу, а також опис компонентів програмного забезпечення незалежно від того, чи є вони новими чи раніше виготовленими і посилання на базову версію, з якої вони були створені. завантажено. Цей опис має також включати похідні вимоги, що випливають із процесу розробки програмного забезпечення. Якщо система містить неактивний код, опис заходів безпеки для активації коду на цільовому комп’ютері та обґрунтування проектних рішень безпосередньо включені до системних вимог, пов’язаних із її безпекою.

Розглянемо процес автоматичного контролю який складається з двох основних етапів. Перший етап полягає в сприйнятті інформації про стан об'єкта і зовнішні умови (первинне перетворення) і в перетворенні її на вигляд, зручний для подальшої обробки (проміжне перетворення). Другий етап полягає у виявленні одержаної інформації ознак контрольованої події, тобто тих специфічних особливостей, які відрізняють цю подію від всіх інших, і в формуванні сигналу про настання цієї події при наявності повної сукупності, що характеризують його ознаки.

У [10] розглядається метод аналізу функціонально пов'язаних безперервних систем з метою визначення мінімального числа точок контролю. За блочно-функціональною схемою будується логічна модель, строгий математичний опис у термінах алгебри логіки. Наводиться алгоритм визначення мінімального числа точок контролю. Наводяться приклади складання таблиці несправностей і визначення мінімальних сукупностей

виходів. Ступінь зв'язаності елементів в системі є одним із чинників, що визначають методологію випробувань [16]. Одним із недоліків передчасного переходу до випробувань системи є те, що в цьому випадку буде дуже багато незрозумілих відмов елементів, а це дуже ускладнює дослідження статистики відмов. Інший недолік – відмови одних елементів зроблять жорсткішим режим роботи інших.

У [11] наводиться методика аналізу одного класу систем для визначення мінімального числа контрольованих виходів, що забезпечують перевірку працездатності або діагностику ушкоджень. Модель також має передбачає блочно-функціональну структуру. На розбиття системи на блоки накладається низка обмежень. Зокрема, йдеться про системи без зворотного зв'язку і без резервування, оскільки в іншому випадку в системі будуть невиразні несправності.

У [12] розглядається задача виділення невідомого автомата з заданого класу автоматів шляхом подачі послідовностей вхідних символів і спостереження реакцій. Завдання розглядається для випадку, коли автомати з істотними несправностями відрізняються.

Ручні засоби контролю мають недоліки: низьку швидкість, недостатню достовірність і в певних випадках високу вартість перевірок [13]. Автоматичні системи контролю дозволяють в значній мірі усунути ці недоліки. Складність процесу контролю визначається складністю об'єкта. Для опису функціонування об'єкта створюється математична функціонально-статистична модель. Модель задається системою рівнянь, що описують залежність параметрів об'єкта від зовнішніх і внутрішніх впливів при функціонуванні.

Загальна ймовірність працездатності об'єкта після його r-й перевірки і настройки в [14] визначається за формулою 1:

$$P^{(r)} = \prod_{i=1}^{n_1} D_i^{(r)} \prod_{i=1}^{n_1} P_{\sigma i/i-1} \prod_{i=n_1+1}^{n} P_{\sigma i/i-1}, \qquad (1)$$

де $\boldsymbol{D_i^{(r)}}$ – достовірність *r*-й перевірки і налаштування *i*-го параметра об'єкта, $\boldsymbol{P_{\sigma i/i-1}}$ – умовна ймовірність безвідмовної роботи по *i*-му контрольованим параметром, *n* – загальне число параметрів, що характеризують працездатність об'єкта, n_1 – число параметрів, що контролюються.

У загальному випадку достовірність *r*-й перевірки і налаштування *i*-го контрольованого параметра об'єкта залежить від ймовірності невиявлених і тих неправдивих відмов, які при налаштуванні переходять до невиявлених відмов. Достовірність перевірки і налаштування визначається за формулою 2:

$$D_i^{(r)} = 1 - P_{hi}^{(r)} - P_{f \to hi}^{(r)}, \tag{2}$$

де $\boldsymbol{P_{hi}^{(r)}}$ – ймовірність існування невиявленої відмови, $\boldsymbol{P_{f \to hi}^{(r)}}$ – імовірність невиявлених і тих неправдивих відмов, які при налаштуванні переходять до невиявлених відмов.

У [15] також наголошується, що одним із шляхів підтримки необхідного рівня надійності складних обчислювальних пристроїв у процесі їхньої експлуатації є контроль працездатності та пошук несправностей. Важливе значення має також контроль в процесі виробництва складних пристроїв. Незалежно від того, якими способами і засобами мусить здійснюватися контроль, завжди важливо мати якомога менше витрат часу і апаратури контролю. У зв'язку з цим велике значення має впровадження методів отримання мінімальних сукупностей впливів і контрольних параметрів, а також методів оптимізації програм перевірок, таблиця 1.1.

У [16] розглядається метод побудови мінімальних і мінімізованих тестів. Наводяться оцінки існування тестів меншої довжини щодо мінімізованих тестів, а також імовірні оцінки близькості отриманих тестів до мінімальних.

Таблиця 1.1

Комплексний аналіз методів підвищення надійності мікропроцесорів

Методи	Можливий вплив на інші властивості мікропроцесорів
1. Раціональне проектування системи.	Зменшення загального числа елементів, габаритів, ваги.
2. Введення надлишкового резервування.	
2.1. Структурне резервування.	Значне збільшення габаритів маси, енергоспоживання..
2.2. Часове резервування.	Значне зменшення продуктивності..
2.3. Інформаційне резервування	Збільшення габаритів, маси, енергоспоживання, зменшення швидкодії.
2.4. Функціональне резервування.	Збільшення габаритів, ваги, енергоспоживання.
2.5. Навантажувальне резервування.	Збільшення габаритів, ваги, енергоспоживання.
3. Створення багатоальтернативних (багатоверсійних) систем.	Збільшення термінів і вартості проектування, обсягу апаратури (або зменшення продуктивності).
4. Підвищення відмовостійкості типових елементів.	Збільшення термінів і вартості проектування.
5. Поліпшення технології виробництва.	Збільшення вартості та строків виробництва.
6. Захист від несприятливих зовнішніх факторів.	Збільшення габаритів, ваги, енергоспоживання.
7. Контроль та діагностування в процесі експлуатації.	Збільшення обсягу апаратури. Зменшення продуктивності.
8. Прогнозування та профілактика відмов.	Збільшення вартості експлуатації. Необхідність проведення профілактики.
9. Управління відмовостійкістю міропроцесорів за фактичним технічним станом..	

Розвиток апаратури призводить до ускладнення вирішуваних завдань, що спричиняє ускладнення схем. Для правильного визначення працездатності важлива кількісна оцінка з високою точністю, а для локалізації несправності – більша кількість діагностичної інформації. З'явилася блочно-модульна побудова, яка забезпечує більш швидке усунення несправностей. Успішне вирішення завдань забезпечувалося для невеликих пристроїв із малим числом функціональних блоків.

У [17] пропонується знайти функціональну залежність між складовими вектора і параметрами елементів пристрою, яка є одним з m співвідношень, що характеризують складові вектора, отримані при випробуваннях. Результат будь-якого вимірювання вихідного параметра містить в собі зміну його значення як через випадкові зміни параметрів елементів у межах поля допусків, так і внаслідок пошкоджень елементів. Для вирішення завдання пошуку елемента, що відмовив, необхідно досліджувати всі можливі відмови елементів з тим, щоб визначити, яке ушкодження найтісніше пов'язані з отриманою величиною вихідного параметра. Згідно [18] автоматичний

контроль в поєднанні з можливістю усунення несправностей є одним із способів підвищення надійності апаратури.

У [19] зазначається, що класичні методи діагностики розглядають комбінаційні і послідовні схеми з метою визначення мінімального набору тестів. Метод чорного ящика дає лише функціональну перевірку справності, без можливості пошуку місця несправності. Інший підхід заснований на створенні діагностичних словників. Пропонується розглядати систему окремо на поведінковому і структурному рівнях. На структурному рівні зручно застосувати відображення у вигляді графів. Будь-яка дискретна послідовна система ізоморфна орієнтованому графу. Вершини відображають функціональні елементи (комбінаційні або послідовні), дуги з'єднання між ними. За графу можна побудувати матрицю суміжності, а з неї, відповідно, отримати матрицю досяжності. У складній системі проводиться сегментація, цикли і ланцюги графа замінюються узагальненими вершинами. Контрольні точки мають обиратися таким чином, щоб тестова процедура ізолювала підсистему від її сусідів з дотриманням умов спостереження та управління.

У [20] наводиться метод побудови програм перевірки дискретних пристроїв з пам'яттю, робота яких описується моделлю кінцевого автомата у вигляді таблиць переходів і виходів. Відповіді автоматів на тестові впливи є значеннями ознак, за якими проводиться розбиття множини автоматів на класи. Наводиться алгоритм побудови діагностичної програми.

У [21] розглядається задача контролю працездатності і діагностики несправностей в дискретних блокових об'єктах. При цьому об'єкти представляються логічними моделями, що відображають причинно-наслідкові зв'язки, що існують в об'єкті при наявності несправностей. Досліджуються питання побудови універсальних діагностичних пристроїв для локалізації несправностей з точністю до змінних блоків. Створені раніше методи належать переважно до комбінаційних схем. Тестовий контроль дискретних об'єктів вимагає виконання великого числа обчислювальних операцій. Перспективним шляхом слід вважати апаратні способи

діагностики. Наводяться моделі для синхронних і асинхронних дискретних об'єктів. Розглядаються питання реалізації універсального пристрою діагностики для застосування його в якості зовнішнього пристрою, як вбудованого пристрою для об'єктів з резервуванням, як вбудованого пристрою для об'єктів, що мають вбудований контроль в блоках.

У [22] розглядається методика побудови оптимальних універсальних систем контролю на основі теорії статистичних рішень. Структурно система контролю складається з формувача (перетворює простір станів на простір сигналів), класифікатора (що вказує прийнятим за отриманими сигналами приналежність стану до відповідного класу) і інформатора (видає повідомлення на основі прийнятих сигналів і вказівок класифікатора).

У [23] розглядається діагностика несправностей комбінаційних схем на основі еквівалентної нормальної форми.

Згідно [24] основними характеристиками якості розпізнавання є помилки розпізнавання і середні втрати. Там же наводяться статистичні критерії виявлення: мінімального ризику Байеса, ідеального спостерігача Зигерта-Котельникова, максимальної правдоподібності Фішера, мінімаксний, спостерігача Неймана-Пірсона, мінімальної тривалості експерименту Вальда; а також статистичні критерії розпізнавання: мінімального ризику і сумарного перевищення.

У [25] дається класифікація методів технічного діагностування обчислювальних пристроїв. Розрізняють універсальні та спеціальні методи. Універсальні методи побудови перевіряючих і діагностичних тестів розбиваються на три основних види:

– метод теорії експериментів;
– метод таблиць функцій несправностей;
– метод істотних шляхів.

У [26] розглядається оптимальний пошук несправностей. Система, що розглядається як об'єкт контролю, це сукупність складових її елементів, з'єднаних між собою функціональними зв'язками. Кожен з елементів системи

може знаходитися в двох станах: справності або відмови. Контроль системи полягає в послідовному застосуванні спеціальних тестів, кожен з яких перевіряє справність цілком певної підмножини елементів. Перевірка проводиться або для підтвердження справності, або для віднайдення несправності. Тест, який перевіряє всю систему , буває неможливий або недоцільний, з цього вигідніше використовувати послідовність декількох простих тестів. У загальному випадку підмножини елементів, що перевіряються різними тестами, можуть перетинатися [27].

Наводяться алгоритми віднайдення єдиного несправного елемента, а також невідомого числа несправних елементів при довільних пересічних тестах і при поелементній перевірці. У загальному вигляді тестова стратегія ґрунтується на впорядкування тестів по тимчасових витратах і по можливостям відмов елементів. Аналогічний підхід застосовується і для виявлення несправності.

У [28] наводиться метод побудови повних перевіряючих і діагностичних тестів комбінаційних пристроїв. Пропонується спосіб редукції повного діагностичного тесту до повного перевіряючого тесту. При побудові перевіряючого тесту очікувана реакція випробуваного пристрою вважається такою, яка збігається з реакцією справного пристрою. В іншому випадку перевірка закінчується і пристрій вважається несправним. Для діагностичного експерименту реакція випробуваного пристрою заздалегідь невідома.

У [29] наводиться зв'язок достовірності контролю з імовірністю безвідмовної роботи. Достовірність як правильне відображення реального стану об'єкта контролю вважається основним критерієм оцінки якості контролю. Поряд з економічними і експлуатаційними характеристиками достовірність визначає ефективність контролю.

У [30] першочергова увага приділяється логічним аспектам технічної діагностики. По-перше, для більшості завдань побудови алгоритмів діагнозу, так само як і методів їхнього вирішення, притаманний комбінаторний

характер. По-друге, допустимі способи діагнозу технічного стану безперервних об'єктів завдяки простоті, яка властива для цих способів реалізації, набули найширшого розповсюдження. По-третє, дискретні об'єкти – це великий клас пристроїв і систем, в основу якого має бути покладений логічний підхід. Більшу частину технічних засобів діагнозу представляють пристрої дискретної дії.

В якості універсальної математичної моделі розглядаються таблиця функцій несправностей, де за рядками відкладаються елементарні перевірки, за стовпцями – технічні стани об'єкта або, що те ж саме, – функції, що реалізуються об'єктом в різних станах.

Логічна схема зіставляється з орієнтованим графом, вершинам якого відповідають логічні елементи, вхідні і вихідні полюси, а дугам – з'єднання, причому впорядковані, тому що функція, що реалізується елементом, в загальному випадку несиметрична щодо перестановок вхідних змінних. Логічна мережа може бути представлена в дужковій формі, а також в еквівалентній нормальній формі. Під фізичною несправністю можна розуміти наслідки певної події, яка перетворює справний пристрій на несправний. Можливі наслідки, які є несправностями при структурному описі і не є такими при його функціональному описі. Серед найбільш відомих несправностей з'єднань елементів виділяють наступні:

– обрив з'єднання;
– замикання з'єднання з шиною живлення;
– переплутування зв'язків;
– поява зайвих зв'язків;
– замикання декількох зв'язків між собою;
– сукупність несправностей.

Дані несправності можуть бути еквівалентні подачі на вхід постійного сигналу 0 або 1, зміні функції елемента, зміні класу пристрою.

Таблиця функцій несправностей є явною моделлю об'єкта діагнозу. При роботі з ТФН багато завдань діагнозу вирішуються просто в обмін на

великі обсяги обчислень і пам'яті.

Іншим способом пошуку несправностей є використання діагностичного словника. Повний діагностичний словник має містити значення виходів комбінаційного пристрою при всіх можливих несправностях і на всій множині вхідних наборів пристрою. За збігом фактичних вихідних значень випробуваного пристрою зі значеннями зі словника визначається технічний стан.

У [31] описана автоматизована система забезпечення надійності функціонування технологічних установок безперервної дії. Автоматизації зазнали операції діагностування, визначення надійності та прийняття рішення про корегувальні впливи. Застосування подібних систем стрімко знижує час відновлення, виключає небезпечні наслідки і причини відмов апаратури або зменшує їхню ймовірність.

Завдання побудови діагностичної моделі зводиться до оптимізації процесу розпізнавання несправностей в об'єкті і визначенню множин: діагностичних і режимних параметрів, класів несправностей, описів несправностей в просторі ознак, алгоритмів формування ознак і розпізнавання класів несправностей.

У [32] наводиться методика динамічного контролю. Динамічний контроль – певні дії, спрямовані на визначення технічного стану системи за результатами вимірювання істотно змінних у часі сигналів. Динамічний контроль знаходить основне застосування для визначення працездатності інерційних об'єктів.

Згідно [33] метою технічної діагностики є підвищення надійності ресурсу технічних систем. Основне завдання – розпізнавання стану технічної системи в умовах обмеженої інформації. Теоретичний фундамент – теорія розпізнавання образів, завдання класифікації. Алгоритми розпізнавання ґрунтуються на діагностичних моделях, що встановлюють зв'язок між станами технічної системи і їхнім відображеннями у просторі діагностичних сигналів.

У [34, 35] алгоритми виявлення несправностей є частиною методики реалізації відмовостійкості. Вони мають забезпечувати виявлення всіх необхідних класів несправностей, а також достатню ймовірність своєчасного виявлення. Перший крок забезпечення відмовостійкості – чітка ідентифікація несправностей, що становлять інтерес. Перший етап виявлення несправностей – початковий контроль, що здійснюється перед нормальною експлуатацією і служить для виявлення несправних елементів апаратури, дефекти яких виникли під час виробництва або збірки. Повний контроль на логічному рівні економічно неможливий. Зростає роль імовірнісних підходів і комплексного логіко-функціонального контролю. Вартість початкового контролю становить значну частину загальної вартості цифрових схем.

У [36] наводиться аналіз впливу допусків пасивних елементів ланцюгів на ненадійні характеристики системи. У роботі застосовуються методи теорії графів і матриць.

У [37] в якості прикладу типових задач дослідження операцій наводиться організація вибіркового контролю продукції для забезпечення заданого рівня якості при мінімальних витратах на контроль. Щоб порівнювати між собою по ефективності різні рішення, потрібно мати певний кількісний критерій, показник ефективності, цільову функцію. Для вибіркового контролю такий показник – середні очікувані витрати R на контроль за одиницю часу за умови, що система контролю забезпечує заданий рівень якості, наприклад, середній відсоток браку не вище заданого. При цьому R потрібно мінімізувати.

У [38] акцентується на тому, що навіть неповний літературний огляд показує, що запропонована досить велика кількість методів оцінки працездатності та ступеня працездатності об'єктів і систем управління. Однак практичне втілення цих методів стикається з такими труднощами, як значне число непрямих вимірювань з обробкою для визначення поточних значень параметрів системи. При великому числі контрольованих параметрів обсяг обчислень надзвичайно великий, а процес обробки громіздкий і непридатний

для реалізації за прийнятний час. Зменшення числа контрольованих параметрів і використання наближених залежностей дає лише наближене уявлення про працездатність системи.

У [39] розглядається задача перевірки правильності функціонування безперервних динамічних об'єктів в робочому режимі. Основну увагу приділено групі методів, в яких діагностування виконується шляхом перевірки алгебраїчних співвідношень між вихідними сигналами об'єкта і пристрою діагностування.

У [40, 41] методи виявлення помилок в цифрових схемах об'єднуються в наступні групи:

– модифікація схем дублювання і порівняння,
– використання спеціальних кодів,
– алгебраїчний метод,
– контроль за перебігом процесу.

У [42] викладаються питання автоматизації контролю цифрових функціональних модулів засобів обчислювальної техніки та радіоелектронної апаратури. Розглядаються технологічні процеси виробництва цифрових функціональних модулів, методи і процедури пошуку несправностей, принципи побудови автоматизованих систем контролю. Підвищення надійності виробів, що виготовляються при одночасному зниженні трудомісткості їхнього контролю, можливо тільки на основі автоматизації операцій контролю і налагодження. Частка трудомісткості контрольних операцій збільшується, що пояснюється вдосконаленням елементної бази, підвищенням її ступеня інтеграції, зростанням складності електронних блоків і модулів.

1.2. Аналіз сучасних засобів інформаційної підтримки процесу проектування та експлуатації комплексу бортового обладнання.

До складу інформаційної підтримки входить множина таких засобів, як методи, моделі, критерії, системи, алгоритмічне, програмне та апаратне

забезпечення, що використовуються для проектування та експлуатації КБО ІМА.

Засоби інформаційна підтримка процесів проектування та експлуатації:

методичні засоби – документи, що характеризують склад, правила відбору та експлуатації засобів автоматизованого проектування та експлуатації;

лінгвістичні засоби – сукупність мов, які застосовуються для опису процедур автоматизованого проектування та проектних рішень в експлуатації.

математичні засоби – математичні моделі проектованих об'єктів, методи та алгоритми проектних процедур, що використовуються при автоматизованому проектуванні та експлуатації;

програмні засоби – об'єднують власне програми для систем обробки даних на машинних носіях та програмну документацію, необхідну для проектування та експлуатації;

інформаційні засоби – поєднує всі можливі дані, необхідні для виконання автоматизованого проектування та експлуатації;

організаційні засоби – включають положення, інструкції, накази, штатні розписи, кваліфікаційні вимоги та інші документи, що регламентують організаційну структуру підрозділів проектної організації та їхню взаємодію з комплексом засобів автоматизованого проектування та експлуатації.

Отже нами проведено аналіз сучасних засоби інформаційної підтримки процесу проектування та експлуатації комплексу бортового обладнання, встановлено, що проблеми незалежної і федеративної архітектури привели до створення та розвитку концепції інтегрованості модульної авіоніки, показано в таблиці 1.

Зауважимо, що основним методом підвищення відмовобезпеки і надійності на сьогоднішній день є резервування. Застосування резервування дозволяє ефективно підвищити показники надійності слабких механізмів і вузлів до необхідного рівня. Метод резервування добре вивчений і існує

безліч різних способів його реалізації.

Основними недоліками методу резервування є:

– зниження ефективності при збільшенні кратності резервування;

– сильне зростання вартості обладнання зі збільшенням кратності резервування;

– сильне зростання вагогабаритних характеристик обладнання зі збільшенням кратності резервування;

– основний і резервний канал як правило ідентичні і мають однаковий час наробітку до відмови, термін служби та ін. Це спричиняє високу ймовірність синхронної відмови всіх резервних каналів.

Проведено аналіз сучасних засобів інформаційної підтримки процесу проектування та експлуатації комплексу бортового обладнання таблиця 1.2

Таблиця 1.2

Аналіз сучасних засоби інформаційної підтримки процесу проектування та експлуатації комплексу бортового обладнання

№	Засоби	Критерії																											
		За ступенем автоматизації		За універсальності		За ступенем уніфікації		За видами каналів зв'язку з об'єктом		За видами обробки інформації		За способом обробки інформації		По виду на об'єкт		За видами контролю			За способом контролю параметрів		За місцем розташування		За кількістю різних класів станів в об'єкті		За способом представлення результатів контролю		За способом прийому інформації		По виду програмного пристрою	
		Неавтоматичні	Автоматичні	Універсальні	Спеціалізовані	Уніфіковані	Неуніфіковані	Проводні	Бездротові	Інформаційно - логічні	Інформаційно - обслуговуючі	З первинною реєстрацією інформації	Без первинної реєстрації інформації	Активні	Пасивні	Прогнозуючі	Діагностичні	Прецедентності	Цифрові	Аналогові	Вбудовані	Зовнішні	Багатозначні	Двозначні	З кількісною оцінкою	З якісною оцінкою	Паралельно	Послідовно	З зовнішнім програмуванням	З внутрішнім програмуванням
1	Методичні засоби	-	±	-	+	+	-	+	-	-	-	-	-	+	-	-	+	-	-	-	+	+	-	-	-	-	+	+	-	-
2	Лінгвістичні засоби	-	+	-	+	-	+	+	-	-	-	+	-	-	+			-	-	-	-	+	+			-	-	+	-	-
3	Математичні засоби	-	+	-	+	-	-	+	-	-	-		+	-	+		+	+	-	-	-	-	-	-	-	-	-	-	-	-
4	Програмні засоби	-	+	+	+	-	-	+	-	+	-	-	-	+	+		-	-	-	-	-	-	-	-	-	-	-	-	-	-
5	Інформаційні засоби	-	+	-	-	+	+	-	-	+	-	-	-	-	+							+	+		-	-		+	-	-
6	Організаційні засоби	+	+	-	+	+	-	-	-	-	-	+			+		-	-	-	-	-	-	-	+	+	+	-	-	-	-
7	Інформаційна підтримка ПІпаЕ КБО ІМА	-	+	+	+	+	-	+	+	+	+	+	+	+	-	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+

Сукупність цих недоліків веде до того, що застосування резервування є небажаним і дорогим методом підвищення надійності.

Архітектура ІМА дозволяє значною мірою знизити кількість резервного обладнання в складі КБО. Це властивість ІМА є одна з найбільш значущих і обумовлена концепцією застосування уніфікованих ОМ в поєднанні з властивістю незалежності ПЗ і АЗ [32-35].

Однак, аналізуючи структурну схему типового крейта ІМА, приходимо

до висновку, що резервуванню підлягає весь крейт цілком. При такому підході резервуються не тільки критичні функції комплексу, а й некритичні. Тобто резервування надмірне.

Причиною, яка зумовлює доцільність такого підходу до резервування, є розподіл критичних функцій комплексу між ОМ крейта. У підсумку надмірність ПЗ і АЗ КБО ІМА надвисока.

Ґрунтуючись на принципах роботи алгоритму проєктування структури функцій, можна запропонувати метод зниження надмірності ПЗ і АЗ комплексу.

РГП функцій є одним з критеріїв проєктування структури функцій. На етапі визначення складу ГФ облік РГП дозволяє максимально сконцентрувати критичні функції комплексу в рамках однієї групи. Тобто з отриманої множини ГФ лише мала частина матиме в своєму складі критичні функції.

Тобто при реалізації ПЗ і АЗ крейта вийде, що лише один ОМ і реалізовані на ньому програмні застосунки будуть вимагати резервування. Це означає, що можна мінімізувати комплектність резервного крейта.

Тепер розглянемо механізм реконфігурації КБО ІМА. Цей механізм є основою і методом скорочення кількості резервного обладнання в складі крейта при значному підвищенні його відмовобезпеки [100, 101, 118, 119]. Реконфігурація заснована на процедурі перерозподілу програмних за стосунків між справними ОМ крейта [102-104]. Підставою для виконання реконфігурації є відмова одного або більше ОМ [105, 106]. Метою реконфігурації при критичних режимах експлуатації ПС, в разі пошкодження великого числа ОМ, є перерозподіл програмних застосунків таким чином, щоб забезпечити працездатність найбільш критичних функції [107, 118].

Для ефективної роботи алгоритмів реконфігурації потрібно розбиття функцій КБО на якомога більш дрібні функціональні застосунки. Однак при цьому складність визначення оптимальної конфігурації багаторазово зростає, аж до неприйнятного рівня. Це є перешкодою на шляху реалізації

високоефективного реконфігуруючого КБО ІМА. Крім того, сам алгоритм реконфігурації при такому підході вимагає його реалізації як програмного за стосунку комплексу. А значить, до нього пред'являються ті ж вимоги, що і до всіх інших функцій КБО, що робить реалізацію властивості реконфігурації вкрай складним завданням [105-109].

Однак, якщо реалізувати кожну ГФ як один програмний додаток і призначити відповідні пріоритети між ними, то можна максимально спростити реалізацію властивості реконфігурації комплексу. При виконанні цих умов сутність реконфігурації полягатиме у виконанні максимально можливої кількості програмних застосунків відповідно до їхнього пріоритету.

Аналізуючи масив даних, які формуються при проєктуванні структури функцій, можна зробити висновок про те, що зберігання цих даних можна здійснювати в базі даних (БД) реляційного типу, які широко застосовуються в наш час [110, 117].

Для роботи з реляційними БД існує безліч систем управління БД (СУБД): IMS, DB2, Informix, Oracle, Database, Microsoft SQL Server, Adaptive Server Enterprise, Teradata Database, Firebird, Postgre SQL, MySQL, SQLite, Microsoft Access, Visual FoxPro, Линтер, CouchDB, MongoDB, Cache та ін. Вибір конкретної СУБД визначається можливостями і потребами підприємства. Всі СУБД відрізняються один від одної набором функцій для обслуговування та управління БД. Тому доцільно визначити низку вимог, яким мусить відповідати СУБД [20, 27, 111, 112].

Для формування масиву вихідних даних і побудови графа первинної структури функцій потрібна участь чималої кількості фахівців. Це фахівці різних рівнів проєктування – ПС, комплекс, система, модуль. Фахівці вузького і широкого профілів, які займаються розробкою програмних застосунків, функціональних вузлів, друкованих плат тощо. Кількість фахівців, залучених до процесу проєктування структури функцій, визначається складністю КБО і глибиною розкладання комплексних функцій

на підфункції. При проєктуванні складних КБО ІМА може виникнути необхідність залучення і сторонніх фахівців. Крім цього, очевидно, що необхідно забезпечити доступ до БД як замовника, так і виконавця ДКР. А це вже є питанням міжміської взаємодії. Тоді вже відіграє важливу роль питання збереження таємниці розробки і підприємств.

Таким чином, СУБД має забезпечуватися можливість мережевого доступу до БД хоча б у рамках мережі підприємства і мати вбудовані інструменти для управління політикою доступу до даних.

Другим важливим аспектом є питання надійності зберігання даних. Інформація про структуру функцій проєктованого КБО дуже цінна і об'ємна. Крім того, масиви вихідних даних про деякі функції і фрагменти графа первинної структури функцій є з великою ймовірністю типовими. У зв'язку з чим згодом на підприємстві буде сформована бібліотека даних про функції комплексу. Втрата даних є важко відновлюваною і вимагає великих трудовитрат. Тому вкрай важливо, щоб СУБД містила стандартні інструменти підвищення надійності зберігання даних, перевірки коректності внесених змін та ін.

Спеціалізоване ПЗ як правило пишеться для конкретних, вузькоспеціалізованих завдань, з жорстким математичним і алгоритмічним набором функцій. Ступінь складності завдань, що реалізуються таким методом, може бути досить висока. Кількість операторів при роботі з таким ПЗ як правило не перевищує однієї людини [113].

На сьогоднішній день САПР є найефективнішим інструментом для вирішення трудомістких завдань проєктування. Найбільш значущі ефекти від застосування САПР наступні:

- Скорочення трудомісткості проєктування.
- Скорочення циклу проєктування.
- Скорочення собівартості проєктування.
- Покращення якості проєктування.
- Скорочення витрат на натурне моделювання та випробування.

- Скорочення трудомісткості розробки.
- Скорочення трудомісткості адаптації до умов експлуатації.
- Скорочення трудомісткості супроводу.

Високу ефективність САПР отримали в процесі інтенсивного розвитку упродовж останніх двох десятиліть. Перші САПР з'явилися на початку 60-х років XX століття і були набором частково пов'язаних між собою прикладних програм вузької спеціалізації. Тоді були розроблені програми для вирішення завдань будівельної механіки, аналізу електронних схем і проєктування друкованих плат.

Висока ефективність і величезні перспективи розвитку САПР залучили велику кількість компаній-розробників в дану сферу. Незабаром вигляд САПР змінився докорінно. Вже до кінця XX століття системи САПР зазнали якісних змін і мали вигляд мобільних і чітко організованих систем, здатних до оптимізації, в залежності від функціоналу.

Сьогодні САПР є складними математичними об'єктами, заснованими на поєднанні безлічі прикладних наук. Розробка математичного апарату САПР вимагає використання величезного розмаїття методів обчислювальної математики, статистики, дискретної математики, математичного аналізу, методів математичного програмування, імовірнісних розрахунків, застосування складних алгоритмів та ін.

Сучасні САПР можна класифікувати за безліччю ознак. Класифікація відповідно до ISO/IEC 19501 UML:

- типом об'єкта проєктування;
- складністю об'єкта проєктування;
- рівнем автоматизації проєктування;
- комплексністю автоматизації проєктування;
- характером документів, що випускаються;
- кількістю випущених документів за рік;
- кількістю рівнів технічного забезпечення.

Відповідно до цієї класифікації завдання проєктування структури КБО

є комплексною і за ознакою типу об'єкта проєктування належить до проєктування виробів приладобудування. За ознакою складності об'єкта проєктування, в залежності від конкретного виробу, може належати як до об'єктів середньої складності, так і до об'єктів високої складності.

При проєктуванні КБО застосовується безліч САПР, різних за своїм цільовим призначенням:

– Засоби проєктування CAD (Computer Aided Design).

– Засоби інженерного аналізу CAE (Computer Aided Engineering).

– Засоби управління документообігом PDM (Product Document Management).

– Геоінформаційні системи GIS (Geo informatics Systems).

У свою чергу, серед усіх CAD-рішень за галузевим призначенням вони поділяються на:

– Машинобудівні CAD – MCAD (Mechanical Computer Aided Design).

– САПР електронних пристроїв, EDA (Electronic Design Automation).

А за системами CAE наступні:

– Засоби виконання розрахунків на міцність.

– Засоби виконання теплових розрахунків.

– Засоби побудови гідроаеродинамічних моделей (CFD, Computational Fluid Dynamics).

– Засоби виконання кінематичного аналізу.

– Засоби виконання механічної симуляції (MES, Mechanical Event Simulation).

– Засоби виконання електромагнітних і електродинамічних розрахунків.

Нижче представлені приклади САПР, які отримали на сьогоднішній день найбільш широке поширення:

– Для проєктування електричних систем: AutoCAD Electrical, E3.series, CADdy++

– Для тривимірного проєктування конструкцій апаратури БРЕО:

Autodesk Inventor, SolidWorks, UniGraphics, КОМПАС, CATIA та ін.

– Для інженерних розрахунків і моделювання теплових полів, створюваних апаратурою БРЕО: BETAsoft, Sauna, АСОНІКА-Т та ін.

– Для моделювання електромагнітних полів, що створюються апаратурою БРЕО: OrCAD Family Release, GENESYS та ін.

– Для проведення спектрального аналізу радіочастотних сигналів, що генеруються і прийнятих апаратурою БРЕО – TESLA та ін.

– Для моделювання міцності і резонансних характеристик конструкцій авіаційних виробів: Samcef, Mecano, Boss Quattro та ін.

– Для автоматичного трасування друкованих плат з урахуванням тривимірного компонування елементів монтажу: ACCELEDA, P-CAD, Altium Designer та ін.

– Для конструювання і розводки джгутів в тривимірному просторі корпусів апаратури БРЕО: UG/Wiring, Autodesk Inventor та ін.;

– Для моделювання гідродинамічних процесів у системах охолодження авіаційних виробів з урахуванням тепловиділення електронних радіоелементів і фізичних процесів теплопереносу: Fine/Turbo та ін.

– Для електронного моделювання радіотехнічних сигналів і ланцюгів: Cadence Design, Mentor Graphics, XILINX Foundation, ALTERA, MicroCap .

– Для проєктування програмного забезпечення: C/C++/C#, ADA .

– Для аналізу і розрахунку надійності, готовності та ремонтопридатності: АРБІТР, АРМ, АСОНІКА-К, AnyGraph, CRISS, Block Sim, ITEM Software, RAM Commander, Reliability Workbench, Windchill.

Наведені приклади програмних пакетів САПР за рівнем комплексності є одноетапними. Якщо класифікувати їх за рівнем автоматизації в рамках відповідного етапу проєктування КБО, то їх можна віднести до класу високоавтоматизованих САПР. Якщо розглядати дані САПР у межах всього процесу розробки, то загальний рівень автоматизації буде середнім.

Впровадження САПР дозволяє скоротити час процесу проєктування приблизно на 20%. А впровадження принципів уніфікації та стандартизації

апаратно-програмних розробок підприємства на засадах принципів інтегрованого середовища САПР дозволяє пришвидшити процес проєктування десь нана 50%.

Провівши порівняльний аналіз САПР в основних існуючих аналогів сформувався перелік програм, що дозволяє досягти оптимального рівня автоматизації процесу проєктування КБО [35-41]. Перелік програм представлений в табл. 1.3.

Таблиця 1.3

Перелік САПР для автоматизації процесу проєктування КБО

Етап проєктування	**Завдання проєктування**	**Засіб САПР**	**Ступінь автоматизації**
Ескізне проєктування	Розробка архітектури виробу	–	Низька
	Схемотехнічне опрацювання виробу	AutoCAD Electrical	Висока
	Проєктування друкованих плат	Altium Designer	Висока
	Тривимірне проєктування конструкції виробу	Autodesk Inventor	Висока
	Аналіз і розрахунок надійності	RAM Commander	Висока
Технічне проєктування	Доопрацювання архітектури виробу	AutoCAD Electrical	Низька
	Схемотехнічне доопрацювання виробу	AutoCAD Electrical	Висока
	Доопрацювання друкованих плат	Altium Designer	Висока
	Доопрацювання тривимірних моделей конструкції виробу	Autodesk Inventor	Висока
	Аналіз і розрахунок надійності	RAM Commander	Висока

Виходячи з даних, представлених у табл. 1, можна зробити висновок про те, що сучасний рівень розвитку САПР дуже високий [42]. При виборі САПР надано перевагу продукції компанії Autodesk. Це пов'язано з високим ступенем популярності продукції даної компанії на українському ринку, якістю технічної підтримки, пристосованістю до роботи за стандартами України, високим авторитетом виробника.

САПР на сьогоднішній день дуже сильно інтегровані в усі типові процеси життєвого циклу виробу та їхня роль вкрай велика. Розробка більшості сучасних виробів була б неможлива без САПР. До таких виробів належить і КБО. Аналіз ступеня автоматизації проєктування КБО показав, що найменш автоматизованим і складним завданням проєктування є розробка структури комплексу.

Типовий процес розробки завжди починається з вибору аналогів і прототипу нового виробу [43]. Таким чином, виробник використовує вже готову структуру виробу, лише допрацьовуючи її. Ступінь доопрацювання прототипу вкрай рідко перевищує 20%, що дозволяє в повному обсязі скористатися принципом успадкування найкращих технічних рішень наявних розробок.

Такий підхід є актуальним, коли йдеться про невеликий пристрій з типовим набором функцій. Мета розробки такого пристрою, як і раніше, це покращення його якісних характеристик.

При розгляді КБО як об'єкта проєктування очевидно, що лише частина його функцій є типовими. Кожен новий КБО відмінний від аналогів не лише за якісними показниками, а й функціонально. Розглядаючи архітектуру ІМА, завдання розробки структури комплексу ускладнюється. Виробники випускають КБО ІМА, маючи в своєму розпорядженні величезний досвід розробок лише федеративних комплексів. Водночас певні особливості архітектури змушують виробника кожен КБО ІМА розробляти практично «з нуля».

Для створення реконфігурованих багатопроцесорних бортових

обчислювальних систем з характеристиками, які будуть відповідати вимогам перспективних комплексів бортового обладнання, розробляються бортові системи, структура яких заснована на використанні уніфікованих функціональних модулів різного призначення, причому їх тип і кількість не повинні впливати на принципи організації обчислювальної системи і принципи її функціонування. При створенні засобів і методів контролю КБО класу IMA необхідно використовувати принципи уніфікації і стандартизації, засновані на внутрішніх функціональних вузлах компонентів КБО.

Роботи з проектування функціональних модулів для КБО перспективних ПС в даний час ведуться за двома різними підходами до апаратної реалізації, які визначають внутрішню структуру функціональних модулів класу IMA.

Обчислювальні модулі мають відкриту масштабовану архітектуру, що дозволяэ програмно реконфігурувати обчислювальну систему. Основу модулів становить елементи (в залежності від модуля кількість варіюється від 16 до 64 шт.), які з'єднані швидкодіючими каналами зв'язку. При розташовані в вузлах плоскої сітки і з'єднані «прямокутної» системою зв'язків. При цьому з'єднуються тільки сусідні елементи ПЛІС, а дані між несуміжними елементами ПЛІС передаються «транзитом» через проміжні елементи.

На основі базового модуля може бути побудована обчислювальна система, що містить кілька базових модулів та запрограмована для вирішення функціональної задачі авіоніки.

Запропонована розробниками в модулі апаратна реалізація має очевидні переваги у вигляді високої швидкодії, малих габаритних розмірів базових модулів, легко реалізовану програмно-керовану реконфігурацію.

Разом з цим підхід на основі використання тільки апаратного обладнання має суттєві недоліки у вигляді великого енергоспоживання, що відіб'ється на значне збільшення габаритних розмірів блоку в зв'язку з необхідністю введення складної системи охолодження, та у вигляді істотної

залежності проектного рішення від використовуваної імпортної елементної бази, так як вибір вітчизняних аналогів обмежений. Крім того, потрібна розробка складових проекту: нового програмного забезпечення, операційної системи, компіляторів і комплексів ПЗ для програмування обчислювальної системи.

Відповідно до другого підходу елементом, який виконує основні функціональні завдання, покладені на функціональний модуль, є процесор. Додатково до нього при необхідності можуть бути встановлені співпроцесори і / або мікросхеми ПЛІС.

З огляду на той факт, що в даний час основна увага приділяється на реалізації заміни імпорту в вітчизняних виробах (в тому числі модулів бортових обчислювальних систем ІМА), даний підхід є більш перспективним, ніж реалізація модулів ІМА виключно на елементах.

Обчислювальні системи, що використовуються на ПС, виконують безліч складних функцій в польоті. Таким чином, їх працездатність потрібно відстежувати в кожен момент часу. Використовувані засоби для контролю та діагностики для систем попередніх поколінь не можуть бути використані для контролю та діагностики перспективних обчислювальних систем. Таким чином, необхідно розвивати напрямки дослідження засобів контролю та діагностики бортових цифрових обчислювальних систем класу ІМА та розробляти нові алгоритми для цих цілей.

Вимірювальні системи, що входять до складу бортових обчислювальних систем ІМА, мають похибки, обумовлені їх конструкціями та умовам, в яких функціонує ПС. Для компенсації похибок КБО використовуються відомі алгоритмічні методи, а також можливості реконфігурування структури та адаптації до зовнішніх умов і внутрішньому стану КБО.

До складу бортового обладнання ПС входять базові автономні навігаційні системи, зокрема ІНС, СНС, радіонавігаційні системи, геофізичні та оптичні системи, а також комплексна обробка інформації. Однією з

важливих задач при експлуатації КБО ПС є збереження його працездатності та високої ефективності. Для визначення працездатності та якості функціонування КБО застосовуються різні системи контролю.

Умови експлуатації сучасних ПС характеризуються високими швидкостями, великими висотами та дальністю польоту, дією різноманітних зовнішніх збурень. В таких умовах ймовірність відмов систем зростає. Тому необхідно жорстко контролювати характеристики КБО і окремих систем авіоніки. Для цього застосовуються різні системи контролю на різних етапах експлуатації КБО.

Автоматизовані бортові системи контролю, що включають вбудовані засоби інструментального контролю та системи інформаційного контролю. Використовуються ієрархічні системи засобів контролю, які добре зарекомендували себе на практиці, в яких оцінюється працездатність і достовірність інформації окремих систем і комплексу бортового обладнання в цілому.

Однак при вирішенні завдання контролю бортового обладнання доцільно знати не тільки момент відмови бортових систем, але і передбачити момент виникнення аварійної ситуації, а також інтервали недостовірної роботи обладнання. Вирішення цього завдання за допомогою апріорних прогнозують моделей вимагає проведення тривалих дорогих експериментів, не дозволяє враховувати особливості конкретних систем і здійснювати ефективний контроль високоманеврових ПС.

Тому для здійснення контролю бортового устаткування перспективних ПС доцільно використовувати комплексні системи контролю на базі ДЕС, які дозволяють враховувати режими польоту ПС, мають велику базу даних і ансамбль оціночних критеріїв. Застосування ДЕС на борту ПС пов'язане зі складнощами реалізації, вимогами підвищеної продуктивності КБО. КБО ПС повинні мати бути помехо- та відмовостійкими, видавати достовірну навігаційну інформацію в усіх режимах роботи.

Таким чином, виділений комплекс проблем, який доцільно вирішити в процесі дисертаційного дослідження: розробити структуру системи контролю працездатності КБО ПС на основі принципів функціонування ДЕС, а також високоточні алгоритми контролю стану КБО і окремих навігаційних систем ПС. При синтезі систем контролю та діагностики необхідно враховувати можливість їх реалізації в умовах обмежень потужності КБО.

На основі проведеного аналізу систем контролю та їх алгоритмічного забезпечення зроблено висновок про необхідність розробки універсального алгоритмічного засобу контролю та діагностики стану сучасних вимірювальних комплексів КБО ПС.

1.3. Висновки до першого розділу

Проведений аналіз інформаційної підтримку процесу проектування та експлуатації комплексу бортового обладнання (аналіз сучасних інструментів, програмного забезпечення, баз даних, методів аналізу даних, систем моніторингу та інше) який показав їх обмежені можливості відносно функціоналу, що спрямоване на забезпечення доступності, актуальності, цілісності, комплексності, надійності, зручності використання та захищеності інформації, таким чином, дозволив визначити задачі дослідження, які орієнтовані на побудову ефективних систем інформаційних технологій для авіаційної промисловості та засобів інформаційної підтримки для оптимізації процесів проектування та експлуатації комплексу бортового обладнання.

РОЗДІЛ 2. МЕТОДИ І МОДЕЛІ ІНФОРМАЦІЙНОЇ ПІДТРИМКИ ПРОЦЕСУ ПРОЄКТУВАННЯ ТА ЕКСПЛУАТАЦІЇ КОМПЛЕКСУ БОРТОВОГО ОБЛАДНАННЯ ІНТЕГРАЛЬНОЇ МОДУЛЬНОЇ АВІОНІКИ

IMA сприяв документ «Інтегрована модульна авіоніка (IMA) Розробка та керівництво та міркування щодо сертифікації» (RTCA DO-297) в листопаді 2005 р. і зосередив увагу на нових галузевих стандартах. На той час концепція CPM була добре встановлена. “Aircraft Data Network, Part 7, Avionics Full-Duplex Switched Ethernet Network” (ARINC 664 Part 7) стандарт приніс адаптацію мережевих технологій на основі COTS з підтримкою QoS. APPplication EXecutive (APEX) як ARINC 653, забезпечив структуру для розробки додатків із розділеним часом і простором у середовищі гіпервізора з високою зрілістю продукту операційної системи реального часу. ARINC 653 APEX було вперше опубліковано в 1996 році і оновлено в 2003 році. У військовій авіоніці були розроблені стандарти, пов’язані з запланованим високопродуктивним середовищем авіоніки Fibre Channel – анонімний обмін повідомленнями абонента (FC-AE-ASM) для архітектур IMA з більшою пропускною здатністю, опублікований у 2007 році.

Для досягнення поставленої мети в цьому розділі були розглянуті такі задачі: розробили метод підтримки процесу проектування інтегрованої модульної авіоніки для формалізації процесу визначення множини параметрів побудови комплексів бортового обладнання нового покоління; розробили модель процесу проектування комплексу бортового обладнання інтегрованої модульної авіоніки для побудови математичних методів підвищення надійності та відмовостійкості на етапі проектування; розробити графову модель структури функцій комплексу бортового обладнання інтегрованої модульної авіоніки вхідних і вихідних параметрів для реалізації обчислювальної потужності, навантаження на мережу передачі даних; розробити структурну модель процесу проектування складних автоматизованих систем управління комплексу бортового обладнання для

вирішення проблеми виявлення несправностей автоматизованих систем, встановити взаємозв'язок щодо виявлення аналогічностей та появи помилок такого типу, повідомити та сформулювати звіт.

2.1. Нормативна документація процесу проєктування комплексу бортового обладнання

Нормативна документація для процесу проєктування комплексу бортового обладнання в авіації може включати наступні компоненти:

Технічні вимоги: це документ, що визначає функціональні та технічні вимоги до бортового обладнання. Він описує функціональні можливості, технічні характеристики, інтерфейси з іншими системами, вимоги до надійності, безпеки та забезпечення документацією.

Стандарти: стандарти забезпечують конкретні вимоги щодо дизайну, виготовлення, випробувань, сертифікації та експлуатації бортового обладнання. Ці стандарти можуть бути встановлені організаціями, такими як Міжнародна організація цивільної авіації (МОСА), Європейське агентство з безпеки авіації (EASA), Федеральна авіаційна адміністрація (FAA) США або іншими національними або міжнародними організаціями.

Регуляторні вимоги: ці вимоги встановлюються регуляторними органами і включають в себе правила та положення, що стосуються сертифікації, безпеки, екології та інших аспектів експлуатації бортового обладнання.

Документи проєктування: це конкретні документи, які використовуються для опису процесу проєктування, включаючи технічні специфікації, креслення, діаграми блоків, плани випробувань та інші технічні матеріали. Ця документація є важливою для забезпечення високої якості та безпеки бортового обладнання в авіаційній промисловості.

Boeing 777 став першим цивільним транспортним літаком що використовував множинний доступ (ARINC 629) на цивільному транспортному ПС для підтримки мережі IMA, рисунок 2.1.

Boeing 787:

архітектура АКУ: 787 також використовує IMA, але з оновленою архітектурою, яка дозволяє більшу гнучкість в конфігурації систем;

- функціональні можливості: система управління 787 включає в себе передові системи автоматичного керування та управління польотом, що забезпечує ефективність та економію пального;

- взаємозв'язки: АКУ 787 взаємодіє з різними системами літака, такими як системи електроживлення, авіонічні системи та системи безпеки:

- особливості: Boeing 787 має високо інтегровану систему управління, що забезпечує покращену ефективність та надійність польоту.

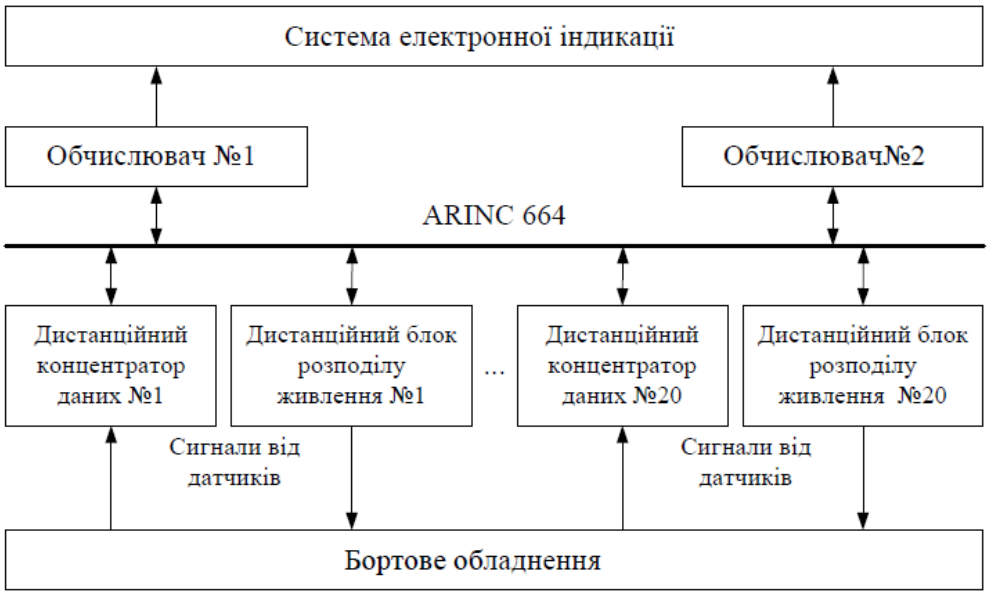


Рис. 2.1 – Архітектура комплексу управління БО ПС Boeing 787

ARINC 629 детермінований і кожен термінал має високу цілісність над асоційованим доступом пристрою до шини, правильним використанням смуги пропускання та відбором переданих і отриманих даних. Хоча це було критичне покращення на 2 Мбіт/с порівняно з мережами ARINC 429 точка-точка, які раніше працювали на 12,5 і 1000 Кбіт/с, ARINC 629 необхідні промислові ASIC і з'єднувачі.

Хоча в IMA був досягнутий прогрес завдяки таким зусиллям, як AIMS, впровадження було оптимізовано для конкретних доменів і досягли лише частини намічених галузевих цілей IMA.

Аналіз архітектури комплексу управління (АКУ) бортового обладнання (БО) пасажирських літаків включає огляд структури систем управління, їх функціональні можливості, взаємозв'язки та особливості кожного літака, рисунок 2.2. Airbus A380:

– архітектура АКУ: A380 використовує IMA, але з більшою кількістю модулів та функцій;

– функціональні можливості: система управління A380 включає в себе продвинуті системи керування польотом, включаючи системи автоматичного керування та управління триманням курсу;

взаємозв'язки: АКУ A380 інтегрується з великою кількістю систем літака, таких як системи керування двигунами, системи безпеки та системи комфорту пасажирів.

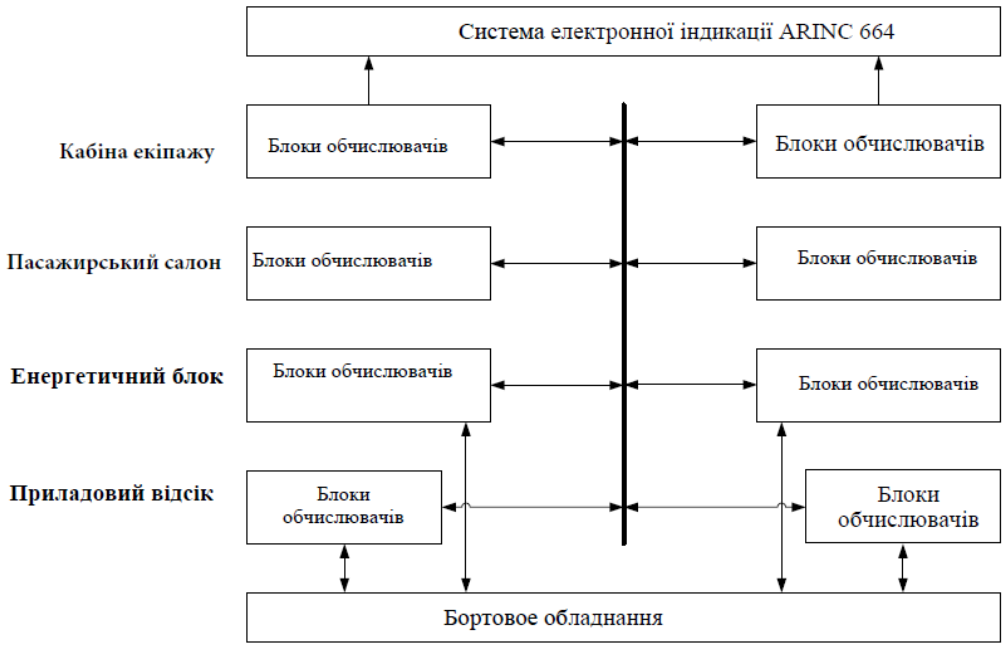


Рис. 2.2 – Архітектура комплексу управління БО ПС Airbus A-380

Кожен з цих літаків має свої унікальні особливості та переваги у системах управління, які відображають вимоги та стратегії виробників в галузі авіаційних технологій.

Спільним для архітектури комплексу управління (АКУ) бортового обладнання (БО) пасажирських літаків, Airbus A380 і Boeing 787 може бути використання інтегрованої модульної авіоніки. Ця концепція дозволяє

об'єднувати різні функції управління в єдиному комплексі, що спрощує розробку, впровадження та підтримку систем управління літака.

Крім того, всі ці літаки використовують передові системи автоматичного керування та управління польотом, які забезпечують безпеку та ефективність польоту. Також, вони мають високо інтегровані системи управління, що дозволяє підвищити автоматизацію та зменшити навантаження на екіпаж.

Взагалі, розробка АКУ для сучасних пасажирських літаків включає в себе використання передових технологій, стандартів безпеки та вимог ефективності, що робить їх відмінними прикладами сучасних авіаційних технологій.

У вітчизнянгому авіапромі дотримання встановленого набору загальних галузевих апаратних модулів може створити перешкоди для отримання переваг від швидкого розвитку інформаційних та електронних технологій.

Існує можливість створити нову програму ПС спільність між платформами різних областей застосування завдяки співпраці між постачальниками платформ.

Ключовими інтерфейсами, які потребують керування, є міжмодуль шини даних (об’єднувальні панелі), мережа та програма програмний інтерфейс (APEX) і більш широке використання розумних периферійних пристроїв і віддалених концентраторів даних значно зменшить потребу в аналогових і дискретних ввід/вивід у шафах IMA та блоках, замінених комп’ютерною лінійкою (LRU).

Аппаратне забезпечення. Уніфікований набір CPM використовується в підсистемах комерційного транспорту в повній конфігурації IMA з використанням стандарту «Керівництво з проектування інтегрованої модульної авіоніки» (ARINC 651) для визначення загальної архітектури апаратного забезпечення, змішаного з апаратним забезпеченням на основі спеціальної стійки або шафи [11] . В Військові IMA, як правило, базуються

на серії COTS і надійних готових (ROTS) OpenVPX або VMEbus CPM на основі ATR або апаратного забезпечення на основі стійки. Через поступові спіралі військових і потребу в розширеній обробці для нових функцій у вибраних підсистемах розгортається кілька поколінь CPM в даній системній архітектурі платформи. Також у військовій авіоніці деякі підсистеми базуються на IMA, деякі використовують спільні кластери обробки IMA, а деякі зберігають об'єднані рішення. Однією з областей небажання використовувати IMA як у цивільних, так і у військових системах є системи керування польотом. Занепокоєння викликає додаткова потреба в більш жорсткій синхронізації каналів і складність керування резервуванням для більш розподілених конфігурацій.

Програмне забезпечення. Додатки авіоніки складаються з набору функцій або завдань, які виконуються на одному або кількох розподілених CPM.

ARINC 653 і його стандарти програмного забезпечення інтерфейсу APEX використовуються як на комерційних, так і на військових платформах, щоб уможливити просторовий і часовий розподіл функцій авіоніки в архітектурі платформи. Стандарт ARINC 653 був представлений у 2005 році. APEX використовується для встановлення тимчасового розділення шляхом введення фіксованих періодично запланованих розділів з обмеженими часовими вікнами, у яких програми можуть виконуватися. Основний кадр представляє основну періодичну послідовність виконання розділу. Основний кадр розбивається на цілу кількість другорядних кадрів однакового періоду та тривалості. Просторове розділення реалізується шляхом визначення віртуального адресного простору для кожного розділу статично під час запуску. Це логічне розділення дозволяє програмам зі змішаним рівнем критичності (як визначено в ARP-4754) працювати на одному CPM. У військових системах для критичних функцій, що не стосуються безпеки, також використовуються інші операційні системи реального часу на основі міркувань переносимості застарілих версій, а також драйвер OSA для

використання Linux як основного інтерфейсу.

Типовий процес проєктування КБО регламентується безліччю різних нормативних документів міжнародного, державного та галузевого рівнів. Тому при розробці методики проєктування структури функцій КБО IMA необхідно врахувати вимоги даних документів.

Виконаємо аналіз нормативної документації, що регламентує типовий процес проєктування КБО.

Відповідно до стандартів, процес проєктування авіоніки поділяється на кілька етапів [53]:

– формування вимог до КБО.
– Розробка концепції проєктування.
– Формування технічного завдання.
– Ескізне проєктування виробу.
– Технічне проєктування виробу.
– Формування робочої документації.
– Введення в дію (підготовка і проведення випробувань КБО).
– Гарантійний супровід і післягарантійне обслуговування.

Кожен етап закінчується формуванням відповідної документації з фіксацією результатів виконаної роботи. Перехід від одного етапу до іншого здійснюється за підсумками оцінки виконаної роботи під час завершеного етапу і ухвалення основоположних рішень про проведення наступного.

Проєктування авіоніки має здійснюватися на основі стандартів і нормативної документації, до яких належать:

– Документація EASA і FAA з норм льотної придатності і сертифікації авіаційного обладнання.

– Документація FAA, EASA, ARINC, RTCA і SAE з процесів розробки, випробувань, сертифікації та виробництва бортового устаткування.

– Національні стандарти України.

– Стандарти ANSI/VITA і нормативні документи, що розробляються ОАК для НДР «Конструктор КБО», «IMA-Конструктор», «IMA-Інтеграція».

– Докладний список даних нормативних документів наведено нижче:

– Авіаційні правила. Частина 21. Процедури сертифікації авіаційної техніки .

– Авіаційні правила. Частина 25. Норми льотної придатності літаків транспортної категорії (АП-25).

– Керівництво з сертифікації систем електронної індикації літаків транспортної категорії.

– Технічні вимоги до літаків транспортної категорії, що виконують всепогодні польоти.

– RTCA DO-248B / EUROCAE ED-94B, Final Report for Clarification of DO- 178B «Software Considerations in Airborne Systems and Equipment Certification».

– EASA Certification Memorandum (SWCEH-002). Software Aspects of Certification.

– ARINC 653. Avionics Application Software Standard Interface.

– IEEE Std 830-1998. Recommended Practice for Software Requirements Specification.

– IEEE Std 1233-1998. Guide for Developing System Requirements Specification.

– ISO/IEC 15289 Systems and software engineering – Content of systems and software life cycle process information products (Documentation).

– ДЕСТ 19.201-78. Технічне завдання, вимоги до змісту та оформлення.

– ДЕСТ 19.202-78. Специфікація. Вимоги до змісту та оформлення.

– ДЕСТ 2.004-88. Єдина система конструкторської документації. Загальні вимоги до виконання конструкторських і технологічних документів на друкованих і графічних пристроях виведення ЕОМ.

– ДЕСТ 2.102-68. Єдина система конструкторської документації. Види і комплектність конструкторських документів.

– ДЕСТ 2.104-68. Єдина система конструкторської документації. Основні написи.

– ДЕСТ 2.105-95. Єдина система конструкторської документації. Загальні вимоги до текстових документів.

– ДЕСТ 2.106-96 Єдина система конструкторської документації. Текстові документи.

– ДЕСТ 7.32-2001. Система стандартів з інформації, бібліотечної та видавничої справи. Звіт про науково-дослідну роботу. Структура і правила оформлення.

– ДЕСТ 2.124-85. Єдина система конструкторської документації. Порядок застосування покупних виробів.

– ДЕСТ ВТ 0019-001-2006. Програмне забезпечення вбудованих систем. Вимоги до змісту та оформлення документів.

– DOC 9613 AN/937. Керівництво з навігації, заснованої на характеристиках (PBN).

– DOC 8168 OPS/611. Здійснення польотів повітряних суден.

– DOC 4444 ATM/501. Керівництво з розробки функцій авіоніки. Організація повітряного руху.

– DOC 8071. Керівництво з випробувань радіонавігаційних засобів. Том II. Випробування супутникових радіонавігаційних систем (GNSS).

– DOC 9863 AN/461. Керівництво з бортової системи попередження зіткнень (БСПЗ).

– При розробці КБО ІМА додатково набувають чинності такі документи:

– ANSI/VITA 46.0-2007. VPX Baseline Standard (Базовий стандарт VPX).

– ANSI/VITA 48.0-2010. Mechanical Specification for Microcomputers Using Ruggedized Enhanced Design Implementation (REDI) (Конструкції мікрокомп’ютерів для жорстких умов експлуатації).

– VITA 46.9 PMC/XMC/Ethernet Signal Mapping to 3U/6U on VPX User IO (Призначення сигналів вводу-виводу і Ethernet-мезонін PMC/XMC для стандарту VPX з форматом модулів 3U і 6U).

– IEEE 1386.1-2001. Standard Physical and Environmental Layers for PCI Mezzanine Cards: PMC (Стандарт на фізичний рівень і навколишні умови для

мезонінних плат с інтерфейсом PCI).

Представлений перелік нормативної документації дозволяє типізувати процес розробки авіоніки і зробити його повністю контрольованим. З цією метою у процес проєктування впроваджуються необхідні для контролю якості процеси. Одним з основних таких процесів є оцінка безпеки. Роботи з оцінки безпеки проєктування КБО відбуваються відповідно за такими документами:

– Окремо процеси, які є частиною процесу оцінки безпеки, регламентуються документами:

– ISO/IEC 12207. Вироби авіаційної техніки. Загальні вимоги до комплексних програм гарантування безпеки польоту, надійності, контролю придатності, експлуатаційної та ремонтної технологічності.

– ISO 62061:2005. Вироби авіаційної техніки. Номенклатура показників безпеки польоту, надійності, контролю придатності, експлуатаційної та ремонтної технологічності.

– ISO 13849-1:2020. Надійність в техніці. Розрахунок надійності. Основні положення.

– ДЕСТ 1 02776-2001. Експлуатація технічна авіаційної техніки за станом. Основні положення.

Перейдемо до нормативної документація для процесу експлуатації (технічного обслуговування) комплексу бортового обладнання в авіації, виділили наступні компоненти:

Технічні стандарти і нормативи: ці документи визначають вимоги до технічного обслуговування, включаючи періодичність, методи та процедури виконання обслуговування, вимоги до перевірки, випробувань та заміни компонентів.

Інструкції з експлуатації: ці документи надають вказівки щодо коректної експлуатації обладнання, включаючи процедури запуску, вимоги до експлуатаційних параметрів, вказівки з виконання регулярного обслуговування та процедури взаємодії з іншими системами.

Плани технічного обслуговування (ТО): ці документи містять розклади та програми для проведення регулярного обслуговування, включаючи періодичність, обсяг робіт, необхідність заміни деталей та інше.

Процедури та стандарти безпеки: ці документи встановлюють вимоги щодо безпеки під час технічного обслуговування, включаючи заходи безпеки для обслуговуючого персоналу, вимоги до обладнання та матеріалів, використовуваних під час обслуговування.

Звіти та документація після обслуговування: ці документи містять інформацію про виконані роботи, результати перевірок, виявлені несправності та заходи, які були вжиті для їх усунення.

Ця документація є важливою для забезпечення безпеки, ефективності та надійності експлуатації бортового обладнання в авіаційній промисловості. Нормативна документація процесу проєктування та комплексу бортового обладнання може бути представлена у вигляді таблиці 2.1

Таблиця 2.1

Класифікація вимог, стандартів і документації у процесі проєктування бортового авіаційного обладнання»

Компонент	Опис	Вимоги	Стандарти	Регуляторні вимоги	Документи проєктування
Технічні вимоги	Функціональні та технічні вимоги до бортового обладнання	Призначення, характеристики, інтерфейси, надійність, безпека, документація	Стандарти МОСА, EASA, FAA	Вимоги авіаційних вад	Технічні специфікації, креслення, діаграми блоків
Стандарти	Вимоги до дизайну, виготовлення, випробувань, сертифікації та експлуатації	Документи стандартів	МОСА, EASA, FAA	Нормативні акти	Стандарти проєктування, випробувань
Контролюючи вимоги	Правила та положення щодо сертифікації, безпеки, екології та інше	Документи регуляторних органів	Авіаційні вади	Закони, правила, інструкції	Регуляторні документи
Документи проєктування	Опис процесу проєктування та технічні матеріали	Технічні специфікації, креслення, діаграми блоків	Стандарти, внутрішні процедури	Внутрішні документи	Плани, специфікації, креслення

До сучасних КБО в обов'язковому порядку пред'являється низка вимог з уніфікації виробів і їхньої взаємозамінності, тому в технічному завданні (ТЗ) на компльекс закладаються вимоги зі стандартизації, уніфікації і каталогізації.

Такий підхід стримує темпи впровадження нових технологій в авіаційній промисловості, однак мінімізує ймовірність помилок і ризиків на стадії проєктування виробів.

Уніфікація мережі залежить від архітектури IMA, але стандарт ARINC 664, частина 7, використовується в комерційному просторі IMA для забезпечення якості обслуговування (QoS) між розподіленими CPM. Важливою перевагою ARINC 664 Part 7 є те, що він може сформувати безперервну мережу, яка охоплює як рівень літака (зв’язок між LRU, центрами IMA та іншими інтелектуальними периферійними пристроями), так і рівень системного домену (зв’язок між модулями, пов’язаними з певною групою систем).

Розроблено метод підтримки поетапного процесу технічної розробки КБО, рисунок 2.3. Етапи взаємодіють та враховуються один одним для забезпечення високої якості та надійності комплексу бортового обладнання.

Розробка \ Етапи	Етап 1	Етап 2	Етап 3	Етап 4	Етап 5	Етап 6	Етап 7
	Аналіз	Розробка та оцінка концепцій	Ескізне проектування	Технічне проектування	Перша поставка	Перший виліт	Сертифікація
Розробка обладнання	Аналіз задачі (MDR)	Аналіз поставленої задачі (PR)	- Аналіз вимог до АЧ (PRR) - Визначення вимог до АЧ - Попереднє проектування (PDR)	Технічне проектування (CDR)	- Реалізація (TRR) - Випробовування	Валідація АЧ (FFR)	Сертифікація АЧ (CR)
Розробка КБО/Систем		Розробка концепцій КБО/системи (PRR)	- Розробка вимог (SRR) - Попереднє проектування (PDR)	Детальне проектування (CDR)	- Моделювання - Інтеграція та випробовування (TRR) - Верифікація системи (QR) - Інтеграція і верифікація першого льотного зразка КБО (AR)	Валідація КБО (FFR)	Сертифікація систем та компонентів КБО (CR)
Розробка програмного забезпечення		Аналіз поставленої задачі (PR)	- Аналіз вимог до ПЗ (PRR) - Розробка вимог високого рівня до ПЗ (ADR)	Розробка вимог низького рівня до ПЗ (архітектура) (DDR)	- Кодування (TRR) - Інтеграція ПЗ і верифікація	Валідація ПЗ (FFR)	Сертифікація ПЗ (CR)

AR – Acceptance
ADR – Architecture Design Review
CDR – Critical Design Review
CR – Certification Review
DDR – Detail Design Review
FFR – First Flight Review
MDR – Mission Definition Review
PDR – Preliminary Design Review
PR – Planning Review
PRR – Preliminary Requirements Review
SRR – System Requirements Review
QR – Qualification Review
TRR – Test Readiness Review

Рис 2.3 Метод підтримки поетапного процесу технічної розробки КБО.

Аналіз вимог і концепція: починаючи з цього етапу, необхідно розглядати вимоги до підтримки (експлуатації) комплексу. Це включає

урахування функціональних та технічних вимог щодо підтримки в процесі роботи та технічного обслуговування обладнання.

Проєктування з урахуванням підтримки: на цьому етапі враховуються вимоги щодо доступності для технічного обслуговування, можливості діагностики та моніторингу стану обладнання, а також способи підтримки в разі виникнення несправностей.

Випробування з урахуванням підтримки: під час випробувань важливо перевіряти не лише функціональність самого обладнання, але й його можливості для ефективної підтримки та обслуговування. Це включає перевірку систем діагностики, резервних блоків та можливостей заміни елементів.

Сертифікація та супровід: після успішного завершення випробувань важливо отримати сертифікацію, яка підтвердить не лише відповідність стандартам якості та безпеки, але й врахує аспекти підтримки та обслуговування.

Експлуатація та підтримка: після введення комплексу в експлуатацію важливо забезпечити ефективну систему підтримки, яка включає в себе регулярне технічне обслуговування, навчання персоналу, постачання запасних частин та інші аспекти, що забезпечують надійну та безперебійну роботу обладнання.

Стандарт ARINC 664, частина 7, був представлений у 2005 році. Віддалені завдання спілкуються через порти APEX на каналі APEX. Дві розподілені комунікаційні задачі належать до розділу джерела та розділу призначення відповідно. На стороні передачі швидкість передачі даних пов’язана з одним віртуальним багатоадресним односпрямованим каналом зв’язку, який називається «Віртуальний канал». Налаштований комутатор ARINC 664 Part 7 виконує переадресацію пакетів, а також забезпечує контроль трафіку на своїх вхідних портах і працює на швидкості 10 або 100 Мбіт/с. У військових архітектурах IMA широко використовується стандартне з’єднання Ethernet у системах місій, де адекватного керування смугою

пропускання достатньо для задоволення вимог QoS датчиків і каналів передачі даних без потреби в спеціалізованому ARINC 664 Частина 7 перемикачі.

Мережі IMA є надзвичайно гнучкими, інтелектуальними магістральними мережами, які з'єднують зв'язок між різними форматами та структурами даних. Типова структура мережі IMA показана на рисинку 2.4.

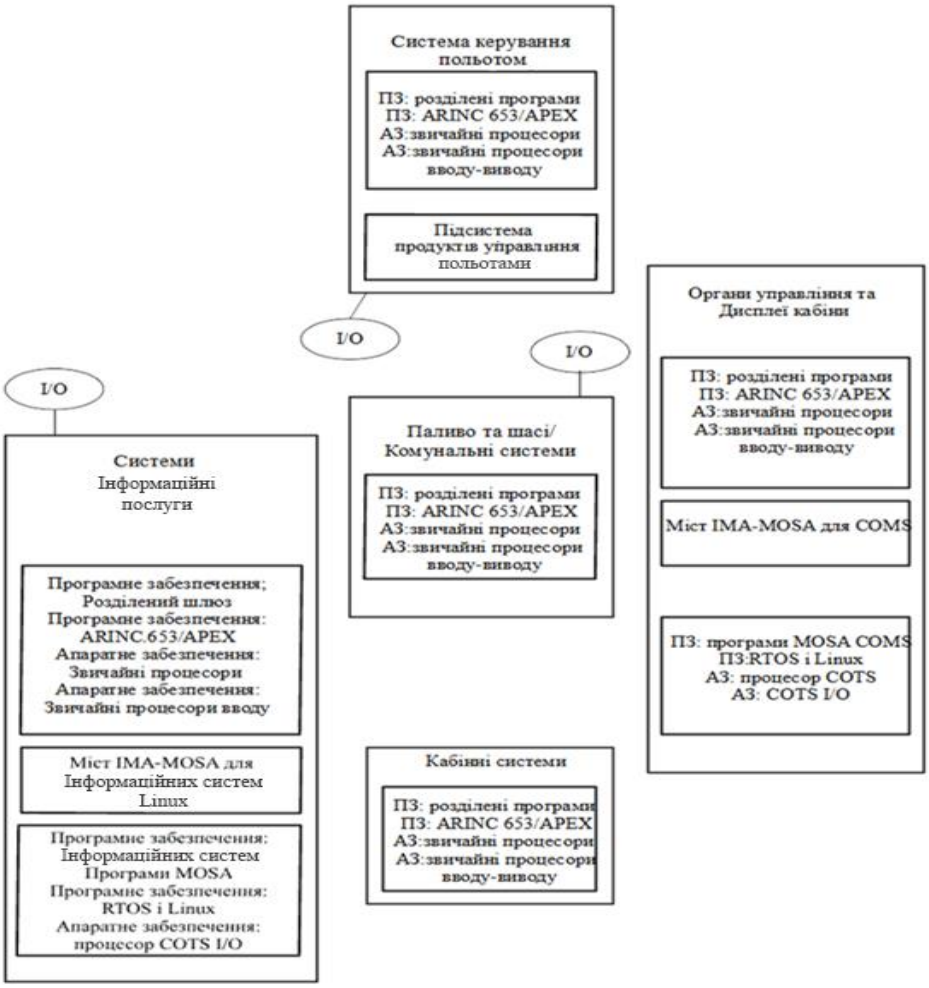


Рис. 2.4. Інформаційна підтримка IMA уніфікованої мережі взаємозвязку з резервною топологією ARINC 664

Інформаційна підтримка IMA (Integrated Modular Avionics) уніфікованої мережі взаємозв'язку з резервною топологією ARINC 664 (Avionics Full-Duplex Switched Ethernet) включає в себе ряд функцій та характеристик, що забезпечують надійну та ефективну роботу системи. Основні аспекти цієї підтримки включають:

1. Діагностика та моніторинг: система забезпечує можливість

виявлення та діагностику несправностей в мережі ARINC 664, щоб забезпечити швидке виявлення та виправлення проблем.

2. Автоматичне переключення на резервну топологію: в разі виявлення несправностей або втрати зв'язку система автоматично переключатися на резервну топологію ARINC 664, щоб забезпечити безперервність обміну даними.

3. Конфігурування та керування мережею: забезпечує зручний інтерфейс для конфігурування та керування параметрами мережі, такими як швидкість передачі даних, призначення портів та інші.

4. Захист від несанкціонованого доступу: забезпечення безпеки мережі шляхом встановлення заходів захисту від несанкціонованого доступу до даних та контролю за доступом до системи.

5. Підтримка резервування ресурсів: система забезпечує можливість резервування ресурсів мережі (наприклад, багатоканальний обмін даними) для забезпечення надійності та безперервності роботи.

Ці функції та характеристики допомагають забезпечити надійну та ефективну роботу уніфікованої мережі взаємозв'язку з резервною топологією ARINC 664 у системах IMA.

Ключовим елементом IMA є відокремлення прямого вводу-виводу для кожної функції та створення елемента даних, доступного в мережі. У сучасних реалізаціях деякий тип вводу/виводу або віддаленого терміналу є другим найважливішим апаратним елементом для CPM, з третім — мережевий комутатор. Крім того, діаграма показує похідну від чистої IMA, де такі функції, як керування польотом, інформаційно-розважальна система та зв’язок, можуть використовувати лінійку продуктів OSA, а не звичайні будівельні блоки IMA. Органи управління польотом є можливі в чистому IMA, але зазвичай їх уникають через проблеми загального режиму (за винятком випадків, коли архітектура IMA включає різний дизайн, який не є стандартним для цивільних ринків). Інформаційно-розважальні системи та засоби зв’язку – це той випадок, коли може знадобитися мережа IP більш

широка підтримка IP і більше стандартів щодо інфраструктури комутатора. Однак навіть у цих випадках потрібен якийсь міст до мережі ARINC 664 Part 7.

Трафік ARINC 664, частина 7, обмежений затримкою та має витримувати значне тремтіння, яке вимагає асинхронного архітектурного інтерфейсу між системами. На відміну від цього, Ethernet із синхронізованим часом забезпечує критичний до часу потік даних для синхронних архітектур.

Це забезпечує детерміновану мережу Ethernet, яка може бути більш придатною для критичних за часом функцій літака, таких як управління польотом. Синхронна природа Ethernet, що запускається за часом, також підходить для архітектур розподіленої обробки, оскільки забезпечує жорсткий контроль тремтіння даних і суворий детермінізм між елементами обробки. Завдання обробки синхронізуються з періодичними мережевими передачами даних із множинним доступом із розділенням часу (TDMA) [13]. Подібно до того, як ARINC 653 забезпечує розподіл за допомогою розподілу часу на процесорі, протокол Ethernet із синхронізацією за часом забезпечує аналогічний розподіл між критичними та некритичними зв’язками через попередньо визначені часові інтервали в мережі.

Це тимчасове розділення дозволяє кільком класам трафіку співіснувати в одній мережі без перешкод. Багатоцільовий транспортний засіб екіпажу NASA, що розробляється, Orion, використовує технологію Ethernet із синхронізованим часом як магістраль даних, що дозволяє співіснувати трьом класам трафіку в одній мережі [14]. У військових системах авіоніки FC-AE-ASM використовувався для реалізації запланованих мереж як з періодичними, так і з асинхронними операціями як попередник повної можливості SAE AS6802, але з використанням Fibre Channel, а не Ethernet. Там, де плановані мережі не потрібні, у військових системах місії OSA широко використовується стандартна мережа з комутацією GigE разом із додаванням шлюзів безпеки та маршрутизаторів.

Розрахунок якості та надійності комплексу бортового обладнання

протягом всього циклу (проектування з урахуванням експлуатації) включає етапи:

1. Визначення вимог до якості та надійності: на початковому етапі необхідно чітко визначити вимоги до якості та надійності комплексу. Це може включати вимоги до безвідмовної роботи протягом певного часу, вимоги до часу відновлення в разі виникнення несправностей, вимоги до міжремонтного часу та інші.

2. Аналіз ризиків і відмов: проведення аналізу можливих ризиків та відмов, що можуть виникнути в процесі експлуатації обладнання. Це допоможе визначити потенційні проблемні місця та розробити стратегії їх уникнення чи мінімізації впливу.

3. Проєктування з урахуванням надійності: під час проєктування комплексу необхідно враховувати принципи надійності, такі як вибір надійних компонентів, резервування систем, розробка систем діагностики та управління несправностями.

4. Випробування та валідація: проведення випробувань для перевірки відповідності розробленого комплексу вимогам до якості та надійності. Це включає в себе випробування на витривалість, стійкість до зовнішніх впливів, перевірку на роботу в екстремальних умовах тощо.

5. Експлуатація та моніторинг: після введення в експлуатацію важливо проводити моніторинг роботи комплексу для виявлення можливих проблем та їх вирішення. Це допоможе забезпечити постійну високу якість та надійність в роботі обладнання.

6. Аналіз після експлуатації: після закінчення експлуатації проводиться аналіз її результатів для визначення можливих покращень в процесі розробки та експлуатації наступних комплексів.

Отже, розрахунок якості та надійності комплексу бортового обладнання включає в себе ці етапи та вимагає комплексного підходу для забезпечення високої якості та надійності впродовж усього життєвого циклу обладнання.

Розрахунок якості та надійності комплексу бортового обладнання при проектуванні з урахуванням експлуатації включає наступні етапи:

1.Визначення вимог до якості та надійності: на цьому етапі важливо чітко визначити вимоги до якості та надійності комплексу, включаючи вимоги до безвідмовної роботи, часу відновлення в разі виникнення несправностей, часу міжремонтного обслуговування та інші параметри.

2. Аналіз ризиків та відмов: проведення аналізу можливих ризиків та відмов, що можуть виникнути в процесі експлуатації. Це допомагає визначити потенційні проблемні місця та розробити стратегії для їх уникнення або мінімізації впливу.

3. Проєктування з урахуванням надійності: під час проєктування комплексу важливо враховувати принципи надійності, такі як вибір надійних компонентів, резервування систем, розробка систем діагностики та управління несправностями.

4. Випробування та валідація: проведення випробувань для перевірки відповідності розробленого комплексу вимогам до якості та надійності. Це включає в себе випробування на витривалість, стійкість до зовнішніх впливів, перевірку на роботу в екстремальних умовах тощо.

5. Моделювання та прогнозування надійності: використання математичних моделей та статистичних методів для прогнозування надійності комплексу в різних умовах експлуатації.

6. Аналіз та оптимізація підтримки: проведення аналізу ефективності та оптимізація системи підтримки обладнання для забезпечення високої якості та надійності впродовж усього циклу його життєвого циклу.

Ці етапи допомагають забезпечити високу якість та надійність комплексу бортового обладнання в умовах його експлуатації.

Для математичного розрахунку якості та надійності комплексу бортового обладнання при проектуванні з урахуванням експлуатації можна використовувати різні методики та моделі. Ми пропонуємо метод використання надійності систем (наприклад, метод «безвідмовності на час»

та модель «експоненційного розподілу»). Ці моделі використовуємо для розрахунку таких параметрів як середній час до відмови (MTTF - Mean Time To Failure), ймовірність безвідмовної роботи протягом певного часу (reliability), час міжремонтного обслуговування (MTTR - Mean Time To Repair) та інші.

Процедура розрахунку виглядає наступним чином:

1. Визначення параметрів методуі надійності: вибір відповідної моделі надійності та визначення параметрів цієї моделі (середній час до відмови MTTF).

2. Розрахунок ймовірності безвідмовної роботи: за допомогою обраного методу можна розрахувати ймовірність того, що комплекс не вийде з ладу протягом певного періоду часу.

3. Оцінка MTTR та часу відновлення: оцінка середнього часу до відновлення роботи комплексу після виникнення несправності.

4.Аналіз чутливості: проведення аналізу чутливості параметрів моделі для визначення впливу різних факторів на надійність та якість системи.

Отже, розрахунок надійності системи розглянемо метод "безвідмовності" для простої системи з одним компонентом. Припустимо, що цей компонент має експоненційний розподіл часу до відмови з середнім часом до відмови (MTTF) рівним 1000 годинам.

1. Ймовірність безвідмовної роботи протягом певного періоду: ймовірність того, що компонент працюватиме без відмови протягом першого року (8760 годин) можна розрахувати за формулою:

$$\text{P(безвідмовної роботи)} = e^{-\frac{8700}{1000}} \approx 0{,}4$$

Тобто, є приблизно 40% шансів, що компонент не вийде з ладу протягом першого року.

2. Середній час між відмовами: середній час між відмовами (MTTF) можна розрахувати за формулою:

MTTF= $\frac{1}{\lambda}$, де де λ –- параметр інтенсивності відмов, який в даному

випадку дорівнює : $\frac{1}{MTTF} = \frac{1}{1000}$ годин.

Для порівняння продуктивності інтегрованої модульної авіоніки з попередніми системами можна використати такі показники, як надійність, швидкодія, масо-габаритні характеристики, вартість і можливості; таблиц 2.2.

Таблиця 2.2

Порівняльний аналіз характеристик IMA та попередніх систем авіоніки

Показник	**Попередня система**	**IMA**	**% покращення**
Надійність	90%	95%	5.56%
Швидкодія	1000 операцій/с	1200 операцій/с	20%
Масо-габаритні х-ки	100 кг, 1 м3	90 кг, 0.8 м3	10%
Вартість	$1,000,000	$900,000	10%
Можливості	Немає	Розширені	-

1. Надійність: порівняти MTBF та MTTF IMA з попередніми системами. Вищі значення вказують на покращення надійності.

2. Швидкодія: порівняти швидкість обробки та реакції нової системи з попередніми. Якщо IMA пропонує швидші обчислення або реакцію, це може бути позитивним показником продуктивності.

3. Масо-габаритні характеристики: якщо IMA має менші розміри або вагу порівняно з попередніми системами, це може бути плюсом для продуктивності, особливо в авіації.

4. Вартість: порівняти вартість впровадження та експлуатації IMA з попередніми системами. якщо нова система виявляється більш вигідною з економічної точки зору, це може бути плюсом для продуктивності.

5. Можливості: оцінити нові можливості, які надає IMA порівняно з попередніми системами. наприклад, покращена інтеграція, більша масштабованість або підтримка нових стандартів.

Ці показники можна порівняти для оцінки того, наскільки ефективною є нова інтегрована модульна авіоніка порівняно з попередніми системами.

Для порівняння продуктивності IMA з попередніми системами у відсотках можна використаємо наступні кроки:

1. Визначення базового рівня: оберіть одну з попередніх систем як базовий рівень продуктивності і встановіть його на 100%.

2. Визначення покращень: наскільки кращою є IMA порівняно з базовим рівнем для кожного показника (надійність, швидкодія, масо-габаритні характеристики, вартість і можливості).

3. Візуалізація: представимо отримані відсотки покращень для кожного показника у вигляді графіку або таблиці для зручності аналізу.

Цей підхід дозволяє чисельно оцінити, наскільки ефективніше працює нова інтегрована модульна авіоніка порівняно з попередніми системами.

Для порівняння продуктивності IMA з попередніми системами у відсотках побудуємо таблицю, де будуть відображені показники кожної системи та відсоток покращення IMA порівняно з попередніми системами.

У цій таблиці кожен рядок представляє показник (надійність, швидкодія, масо-габаритні характеристики, вартість, можливості) і відображає значення для попередньої системи та IMA. Останній стовпчик показує відсоток покращення IMA у порівнянні з попередньою системою.

Алгоритм роботи інтегрованої модульної авіоніки (IMA) може бути поділений на кілька основних етапів. Ось загальний опис такого алгоритму:

1. Ініціалізація системи: під час запуску системи виконується ініціалізація всіх модулів, перевірка їх працездатності та встановлення зв'язків між ними.

2. Виконання основних функцій: IMA виконує основні функції, для яких вона призначена, наприклад, управління авіонікою, обробка даних від датчиків, навігація тощо. Ці функції можуть бути розділені на модулі для більшої ефективності та гнучкості.

3. Моніторинг та діагностика: IMA постійно моніторить стан своїх модулів та системи в цілому, виявляючи можливі проблеми або відхилення в роботі.

4. Управління відмовами: у разі виявлення відмови в одному з модулів або підсистем, IMA може виконати переналаштування або перенаправлення

завдань на інші модулі або резервні системи, щоб забезпечити продовження роботи.

5. Відновлення після відмови: після усунення проблеми ІМА може відновити роботу та перейти в нормальний режим роботи.

6. Завершення роботи: при завершенні роботи або вимкненні системи ІМА може виконати процедури завершення роботи, зберегти дані та відключити модулі від живлення.

2.2. Методи процесу проєктування та оцінка безпеки комплексу бортового обладнання.

Методи процесу проєктування та оцінки безпеки комплексу бортового обладнання включають в себе ряд підходів та інструментів, які допомагають забезпечити високий рівень безпеки при розробці та експлуатації обладнання:

1.Аналіз функціональних вимог та ризиків: аналіз функціональних вимог до обладнання та визначення потенційних ризиків, які можуть виникнути при їх виконанні. Цей аналіз допомагає визначити ключові аспекти безпеки, які потрібно враховувати під час проєктування.

Визначення функціональних вимог та ідентифікація потенційних ризиків є важливими етапами в процесі проєктування комплексу бортового обладнання. Розглянемо кожен з них докладніше:

– визначення функціональних вимог: цей етап включає в себе ідентифікацію та формулювання функціональних вимог до обладнання, тобто того, що воно повинно робити. Це включає такі аспекти, як функції, процеси, взаємодія з іншими системами та інтерфейси користувача. Ці вимоги слід визначити вичерпно і зрозуміло для всіх учасників проекту;

– ідентифікація потенційних ризиків: на цьому етапі визначаємо можливі ризики, які можуть виникнути в процесі проєктування та експлуатації обладнання. це можуть бути технічні, організаційні, фінансові або інші ризики. Ідентифікація ризиків допомагає управляти ними в

майбутньому шляхом їхнього аналізу та планування заходів щодо запобігання або зменшення їх впливу на проєкт [45].

Ці два етапи є ключовими для успішного процесу проєктування та розробки комплексу бортового обладнання, оскільки вони визначають основні вимоги та ризики, які слід враховувати під час всього процесу.

2. Методи оцінки ризиків: використання різних методів, таких як аналіз ризиків та небезпек, величина ймовірності виникнення події та її наслідки, для оцінки ризиків, пов'язаних з безпекою обладнання [46-48].

Оцінка ризиків є важливим етапом в процесі проєктування та розробки комплексу бортового обладнання. Вона дозволяє ідентифікувати, аналізувати та управляти ризиками, які можуть виникнути під час експлуатації обладнання. Для оцінки ризиків ми пропонуємо:

– матрицю ймовірності та впливу: цей метод використовується для оцінки ризиків на основі їхньої ймовірності виникнення та потенційного впливу на проєкт. Ризики оцінюються за шкалою від низького до високого для ймовірності та впливу, і їхній рівень ризику визначається як добуток цих двох значень;

– аналіз важливих впливів: цей метод визначає потенційний вплив ризиків на ключові аспекти проєкту, такі як бюджет, терміни виконання, якість та інші. Ризики ранжируються за їхнім потенційним впливом на ці аспекти, що допомагає визначити пріоритетність управління ними;

– метод дерева вирішення: цей метод використовується для моделювання ризиків та їхніх можливих наслідків у вигляді Кожен вузол дерева представляє можливий результат події, яка може відбутися, та її ймовірність. Цей метод дозволяє оцінити вплив різних ризиків та вибрати стратегію управління ними;

– метод аналізу ризиків і критичних точок (Hazard Analysis and Critical Control Point, HACCP): цей метод визначення критичних точок у виробничих процесах, де можуть виникнути ризики для якості або безпеки продукції.

Ці методи застосовані окремо або в комбінації для оцінки ризиків у процесі проєктування та розробки комплексу бортового обладнання.

3. Функціональна безпека (Functional Safety): використання стандартів, таких як IEC 61508, для забезпечення функціональної безпеки обладнання. Ці стандарти надають методи та вимоги щодо розробки безпечних систем.

Функціональна безпека (Functional Safety) – це концепція, яка описує заходи та процеси, спрямовані на запобігання небезпечним ситуаціям, які можуть виникнути внаслідок невідповідності функціонування автоматизованих систем. Виокремо ключові аспекти функціональної безпеки включають в себе:

– ідентифікацію небезпечних ситуацій: аналіз системи для виявлення можливих небезпечних ситуацій, які можуть виникнути через невідповідність функцій системи;

– оцінка ризику: визначення ймовірності виникнення небезпечної ситуації та її потенційного впливу на безпеку людей, довкілля та майна;

– управління ризиками: розробка та впровадження заходів для зниження ризику до прийнятного рівня шляхом вибору відповідних технологій, методів та процесів;

– документація та стандартизація: створення документації, яка описує процеси функціональної безпеки та відповідні стандарти, які регулюють цей процес;

– тестування та верифікація: проведення тестів та верифікації для підтвердження відповідності системи вимогам функціональної безпеки.

Функціональна безпека є важливим аспектом при проєктуванні та розробці автоматизованих систем, особливо в авіаційній галузі, де безпека має велике значення.

4. Валідація та верифікація: проведення процедур валідації та верифікації, що включає в себе перевірку та тестування обладнання на відповідність вимогам щодо безпеки.

5. Аналіз безпеки за допомогою математичних моделей: Використання математичних моделей, таких як динамічні та статистичні моделі, для аналізу ризиків та визначення ймовірності виникнення небезпечних подій [51].

Аналіз безпеки за допомогою математичних моделей є важливим інструментом в процесі розробки та експлуатації авіоніки в авіації. Виокримемо основні з методів аналізу безпеки за допомогою математичних моделей:

– аналіз надійності: використання статистичних методів для оцінки ймовірності безвідмовної роботи системи на протязі певного часу. Це дозволяє виявити можливі проблеми та недоліки, які можуть призвести до відмови системи;

– наліз ризиків: використання динамічних моделей для моделювання різних сценаріїв небезпечних подій та оцінки їхнього потенційного впливу на безпеку системи. Це дозволяє ідентифікувати ключові ризики та розробляти стратегії управління ними;

– аналіз вимог безпеки: використання математичних моделей для перевірки відповідності системи авіоніки вимогам безпеки, встановленим регуляторами та стандартами;

– симуляційне моделювання: використання комп'ютерних симуляцій для відтворення різних сценаріїв функціонування системи та оцінки їхнього впливу на безпеку та надійність.

Це дозволяє розробникам та експлуататорам систем авіоніки в авіації проводити комплексний аналіз безпеки та надійності системи з використанням математичних моделей для забезпечення її високого рівня безпеки.

6. Системи управління безпекою (Safety Management Systems): впровадження систем управління безпекою для постійного контролю та управління ризиками в експлуатації обладнання.

Ці методи забезпечують високий рівень безпеки комплексу бортового

обладнання та враховуються під час усього процесу проєктування та експлуатації [56].

Ми розробили методологічний підхід до оцінки ризиків та функціональної безпеки бортових систем, таблиця 2.3

Таблиця 2.3

Методологічний підхід до оцінки ризиків та функціональної безпеки бортових систем

Метод	Процес	Відсоток
Аналіз функціональних вимог та ризиків	Визначення функціональних вимог та ідентифікація потенційних ризиків	20%
Методи оцінки ризиків	Використання аналізу ризиків та небезпек для оцінки ризиків безпеки	15%
Функціональна безпека	Застосування стандартів, таких як IEC 61508, для забезпечення безпеки функціональності обладнання	25%
Валідація та верифікація	Перевірка відповідності обладнання вимогам щодо безпеки через тестування та перевірку	20%
Аналіз безпеки за допомогою математичних моделей	Використання математичних моделей для аналізу ризиків та ймовірності виникнення небезпечних подій	20%
Системи управління безпекою	Впровадження систем управління безпекою для контролю та управління ризиками	20%

Безпека має першорядне значення для економічного успіху авіаційної системи. Розробник систем авіоніки повинен переконатися, що система забезпечує необхідний рівень безпеки для пасажирів, екіпажу та технічного персоналу. Відмовостійкі системи є важливими з тенденцією до все більш складних цифрових систем [58].

Багато факторів вимагають відмовостійкості в системах, які виконують функції, які повинні підтримуватися без значних перерв. У системах авіоніки такі функції мають вирішальне значення для продовження безпечного польоту або для задовільного виконання завдання; звідси терміни критичний

для польоту та критичний [59]. Отже, що фізичні компоненти є неідеальними, тобто вони неминуче схильні до фізичного погіршення або відмови. Отже, компоненти за своєю суттю мають кінцевий термін використання, який залежить від окремих екземплярів одного типу компонента. Отже, на певному етапі будь-який фізичний компонент демонструватиме раптову несправність або надмірне погіршення, так що несправність може бути виявлена на певному рівні роботи системи.

Другим фактором, що сприяє фізичним несправностям, є неідеальне середовище, в якому працює система авіоніки. Локальні вібрації, вологість, зміна температури, перехідні процеси в електромережі, електромагнітні перешкоди тощо, як правило, спричиняють навантаження на фізичний компонент, що може призвести до раптової відмови або поступового погіршення. Результатом може бути тимчасова або постійна зміна виходу залежно від характеру та тяжкості стресу. Ступінь викликаного зносу може суттєво вплинути на термін служби компонента. На щастя, можна вжити конструктивних заходів, щоб зменшити сприйнятливість до різних впливів навколишнього середовища. Відповідно, для забезпечення надійності системи необхідний досить комплексний підхід до розробки, хоча відмовостійкість є найбільш помітним аспектом, оскільки вона керує організацією та логікою архітектури системи [79].

Основним фактором, який обумовлює необхідність відмовостійкості, є недоліки конструкції. Для досягнення цілісності, необхідної для систем, важливих для польоту, необхідна стійкість до проектних помилок у апаратному та програмному забезпеченні, а також у загальному потокі даних. Виявилося, що покладатися на апаратну мікросхему, яка видає правильний результат за відсутності фізичної несправності, ризиковано, як продемонструвала помилка конструкції, виявлена в блоці з плаваючою комою високопродуктивного мікропроцесора, який широко використовується. Через труднощі з усуненням усіх проектних помилок використовується різнорідна надлишковість для отримання виходів, які

мають бути ідентичними, навіть якщо вони обчислюються різними комп’ютерами. Використання різнорідного резервування є одним із підходів до збоїв загального режиму. Синфазний збій (CMF) виникає, коли копії резервної системи зазнають збоїв майже одночасно, як правило, через одну причину [80-82].

Цифрова авіоніка – це система «жорсткого режиму реального часу», яка видає критичні за часом вихідні дані, які використовуються для керування польотом літака. Ці критичні результати мають бути надійними. Надійна система одночасно надійна і безпечна. Надійність має багато визначень, і часто виражається як ймовірність того, що не буде збою. Іншим визначенням надійності є ймовірність отримання «правильного» результату. Безпека була визначена як ймовірність того, що вихідні дані системи є правильними, або помилка в результатах може бути виявлена. Правильність була визначена як вимога, щоб вихід усіх каналів узгоджувався біт за біт за умови відсутності помилок. Інший підхід до проектування, наближений консенсус, вважає систему правильною, якщо результати збігаються в межах певної межі .

Процедури верифікації оцінки безпеки та виділення окремої проектної процедури генерування структури функцій комплексу бортового обладнання є важливими кроками у формалізації процесу визначення множини параметрів для побудови комплексів бортового обладнання нового покоління.

Процедура верифікації оцінки безпеки включає в себе перевірку та підтвердження відповідності процесу оцінки безпеки встановленим стандартам та вимогам. Це може включати аналіз методів оцінки ризиків, використання стандартів безпеки, таких як IEC 61508, та перевірку відповідності результатів оцінки безпеки вимогам безпеки та нормативам.

Проектна процедура генерування структури функцій комплексу бортового обладнання спрямована на розробку системи функцій та взаємозв'язків між ними для досягнення визначених цілей проекту. Це

включає в себе визначення потреб користувачів, аналіз вимог та розробку структури, яка забезпечить необхідні функції та властивості обладнання.

Ці процедури допомагають систематизувати та уніфікувати процес визначення параметрів для побудови комплексів бортового обладнання, а саме:

– визначення потреб в новому обладнанні або модифікації існуючого на основі аналізу ринку, вимог клієнтів, технічних та експлуатаційних потреб;

– аналіз вимог, розробка документу, що визначає вимоги до нового обладнання, включаючи технічні характеристики, функціональні вимоги, вимоги до безпеки та експлуатації;

– технічне завдання, створення технічного завдання, яке містить вимоги до обладнання та опис процесу його розробки;

– розробка концепції нового обладнання на основі вимог та технічного завдання;

– проєктування та розробка нового обладнання відповідно до визначених вимог та концепції;

– валідація та верифікація нового обладнання для підтвердження відповідності вимогам та визначення параметрів;

– технічне обслуговування та експлуатація нового обладнання з урахуванням визначених параметрів.

Це дозволяє систематизувати та уніфікувати процес визначення параметрів для побудови комплексів бортового обладнання, забезпечуючи систематичний та структурований підхід до розробки нових продуктів, що може значно полегшити розробку та забезпечити високу якість та безпеку обладнання.

Провівши аналіз в першому розділі ми пропонуємо метод який безпосередньо пов'язаний з процесом проєктування та оцінки безпеки комплексу бортового обладнання, що допомагає забезпечити комплексний підхід до розробки та впровадження безпечного обладнання.

Стуктура та порядок методу:

1. Аналіз ризиків: використання аналізу ризиків для ідентифікації потенційних небезпек та визначення заходів для їх управління в процесі проєктування та експлуатації.

2. Функціональна безпека: застосування принципів функціональної безпеки для визначення та контролю функцій обладнання, які впливають на безпеку.

3. Валідація та верифікація: проведення процедур валідації та верифікації, що включають перевірку відповідності обладнання вимогам безпеки та перевірку правильності роботи за допомогою тестування та аналізу.

4. Системи управління безпекою: впровадження систем управління безпекою для забезпечення постійного контролю та управління ризиками в експлуатації обладнання.

Цей метод дозволяє забезпечити повний цикл безпеки від початкового проектування до експлуатації, забезпечуючи високий рівень безпеки та надійності комплексу бортового обладнання.

Виокримили етапи методу інформаційної підтримки процесу проєктування при врахуванні недоліків експлуатації та оцінці безпеки комплексу бортового обладнання:

1. Аналіз інцидентів та відмов: вивчення інцидентів та відмов обладнання під час експлуатації для ідентифікації можливих недоліків та проблем, які потребують уваги під час проєктування нового обладнання.

2. Врахування досвіду експлуатації: використання досвіду попередніх проектів та експлуатації для уникнення повторення помилок та вдосконалення нового обладнання.

3. Аналіз вимог безпеки та стандартів: врахування вимог безпеки та стандартів під час проєктування для забезпечення відповідності безпеці під час експлуатації.

4. Системи управління ризиками: впровадження систем управління

ризиками для ідентифікації та управління ризиками, пов'язаними з недоліками експлуатації та безпекою.

5. Технічні заходи безпеки: розробка та впровадження технічних заходів безпеки, таких як аварійні зупинки, системи автоматичного виявлення та управління, що забезпечують безпеку в разі виникнення небезпек.

Ці методи дозволяють врахувати недоліки експлуатації та забезпечити високий рівень безпеки під час проєктування комплексу бортового обладнання.

Отже інформаційна підтримка процесу проєктування при врахуванні недоліків експлуатації та оцінці безпеки комплексу бортового обладнання складається:

Несправності апаратних компонентів класифікуємо за ступенем, значенням і тривалістю. Ступінь залежить від того, чи помилки, породжені несправністю, локалізовані чи розподілені; значення вказує, чи генерує помилка фіксовані чи змінні помилкові значення; тривалість означає, чи є помилка тимчасовою чи постійною. Кілька досліджень показали, що постійні несправності спричиняють лише невелику частку всіх виявлених помилок порівняно з тимчасовими несправностями. Перехідний збій, що повторюється, часто називають періодичним. Рисунок 2.5 зображує ці класифікації в дереві несправностей.

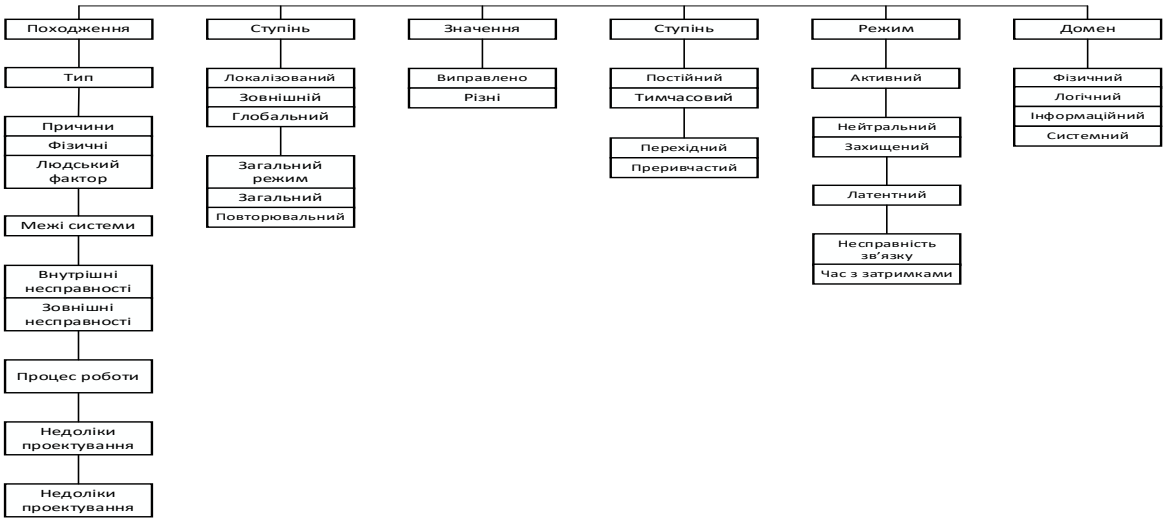


Рис. 2.5 Класифікація несправностей авіоніки бортового обладнання.

Недоліки, які виникають під час розробки системи (від специфікації вимог до впровадження), наступних модифікацій, встановлення процедур експлуатації чи підтримки системи, або операційні помилки, які з’являються під час роботи системи.

Помилки походження можуть виникати внаслідок фізичної несправності апаратного компонента системи або можуть виникати внаслідок помилок, спричинених людиною. Внутрішні помилки меж системи — це ті частини стану системи, які, викликані обчислювальною діяльністю, призведуть до помилки, тоді як зовнішні помилки є результатом втручання системи, спричиненого її фізичним середовищем, або взаємодії системи з людським середовищем. Помилки походження, класифіковані за фазою часу створення, включають помилки конструкції, що є результатом характеристики несправностей.

Оскільки реалізація програми для інтегрованої модульної авіоніки може бути досить складною та включає в себе багато компонентів, побудуємо структуру програми, яка враховує основні етапи роботи IMA:

```
#include <stdio.h>

// Функція для ініціалізації системи
void initializeSystem() {
    // Код для ініціалізації
}

// Функція для виконання основних функцій
void performFunctions() {
    // Код для виконання функцій
}

// Функція для моніторингу та діагностики
void monitorAndDiagnose() {
    // Код для моніторингу та діагностики
}

// Функція для управління відмовами
void manageFailures() {
    // Код для управління відмовами
}

// Функція для відновлення після відмови
void recoverAfterFailure() {
    // Код для відновлення після відмови
}
```

```
// Функція для завершення роботи
void shutdownSystem() {
   // Код для завершення роботи
}

// Основна функція
int main() {
   // Ініціалізація системи
   initializeSystem();

   // Основний цикл роботи системи
   while (1) {
      // Виконання основних функцій
      performFunctions();

      // Моніторинг та діагностика
      monitorAndDiagnose();

      // Управління відмовами
      manageFailures();

      // Відновлення після відмови
      recoverAfterFailure();

      // Перевірка на необхідність завершення роботи
      if (needToShutdown()) {
         break;
      }
   }

   // Завершення роботи
   shutdownSystem();

   return 0;
}
```

Це лише «уривок» програми, яка відображає загальну структуру роботи ІМА. Конкретна реалізація буде залежати від конкретних вимог та функціональних можливостей системи.

При відмові одного з блоків ІМА, система повинна взяти це до уваги та забезпечити безперервну роботу інших блоків або відновити роботу з резервних блоків (якщо такі є). Ось загальна структура програми для обробки відмови одного з блоків ІМА:

```
#include <stdio.h>

// Функція для обробки відмови блоку ІМА
void handleFailure(int failedBlock) {
   // Видалення блоку з системи або перенаправлення завдань на інші блоки

   printf("Блок %d ІМА відмовив. Обробка відмови...\n", failedBlock);
   // Код для відновлення роботи системи після відмови блоку
```

```
    // Якщо є резервні блоки, можливо їх активація або переналаштування на них завдань
    if (hasRedundantBlocks()) {
        activateRedundantBlocks();
    } else {
        // Інше відновлення роботи системи без резервних блоків
        recoverSystem();
    }
}

// Функція для перевірки наявності резервних блоків
int hasRedundantBlocks() {
    // Якщо є резервні блоки, повертаємо 1, інакше 0
    return 1; // Припустимо, що у нас є резервні блоки
}

// Функція для активації резервних блоків
void activateRedundantBlocks() {
    // Код для активації резервних блоків
    printf("Активація резервних блоків...\n");
}

// Функція для відновлення роботи системи
void recoverSystem() {
    // Код для відновлення роботи системи без резервних блоків
    printf("Система відновлена після відмови...\n");
}

// Основна функція
int main() {
    // Спробуємо відобразити обробку відмови блоку 2
    handleFailure(2);

    return 0;
}
```

У цій програмі `handleFailure()` викликається при відмові блоку ІМА з номером, переданим у якості аргументу. Вона перевіряє наявність резервних блоків та відновлює роботу системи залежно від наявності резервних блоків.

У реальному часі обробка відмови одного з блоків ІМА вимагає швидкої реакції та відновлення роботи системи безперервної роботи. Для цього можна використовувати систему реального часу (Real-Time Operating System -RTOS) та механізми, що забезпечують відновлення після відмови. Ось загальна структура програми для обробки відмови у реальному часі:

```
#include <stdio.h>

// Функція для обробки відмови блоку ІМА у реальному часі
void handleFailureRealTime(int failedBlock) {
    // Код для відновлення роботи системи після відмови блоку у реальному часі
    printf("Блок %d ІМА відмовив. Обробка відмови у реальному часі...\n", failedBlock);
```

```
    // Якщо є резервні блоки, можливо їх активація або переналаштування на них завдань у
реальному часі
    if (hasRedundantBlocks()) {
        activateRedundantBlocksRealTime();
    } else {
        // Інше відновлення роботи системи без резервних блоків у реальному часі
        recoverSystemRealTime();
    }
}

// Функція для перевірки наявності резервних блоків
int hasRedundantBlocks() {
    // Якщо є резервні блоки, повертаємо 1, інакше 0
    return 1; // Припустимо, що у нас є резервні блоки
}

// Функція для активації резервних блоків у реальному часі
void activateRedundantBlocksRealTime() {
    // Код для активації резервних блоків у реальному часі
    printf("Активація резервних блоків у реальному часі...\n");
}

// Функція для відновлення роботи системи у реальному часі
void recoverSystemRealTime() {
    // Код для відновлення роботи системи без резервних блоків у реальному часі
    printf("Система відновлена після відмови у реальному часі...\n");
}

// Основна функція
int main() {
    // Спробуємо відобразити обробку відмови блоку 2 у реальному часі
    handleFailureRealTime(2);

    return 0;
}
```

У цій програмі `handleFailureRealTime()` викликається при відмові блоку ІМА з номером, переданим у якості аргументу. Вона перевіряє наявність резервних блоків та відновлює роботу системи залежно від наявності резервних блоків у реальному часі.

Експлуатація інтегрованої модульної авіоніки включає в себе ряд дій, спрямованих на забезпечення безперебійної та ефективної роботи авіонічної системи. Основні аспекти експлуатації ІМА спрямовані на забезпечення безпеки та ефективності експлуатації авіонічної системи з використанням ІМА. В таблиці 2.4 представленні експлуатаційні процеси та заходи забезпечення надійності ІМА.

Таблтця 2.4

Експлуатаційні процеси та заходи забезпечення надійності ІМА

Аспект експлуатації	**Опис**
Перевірка перед польотом	Перевірка працездатності ІМА, включаючи перевірку датчиків і систем
Моніторинг у режимі польоту	Постійний моніторинг стану систем та датчиків під час польоту
Управління відмовами	Автоматичне перенаправлення завдань на інші блоки або активація резервів
Відновлення після відмови	Автоматичне відновлення роботи після усунення відмови
Технічне обслуговування	Регулярне технічне обслуговування та підтримка для забезпечення роботи
Оновлення ПЗ	Регулярні оновлення програмного забезпечення для вдосконалення
Документація та звітність	Ведення відповідної документації та звітність про експлуатацію

Несправність знаходиться в активному режимі, якщо вона дає помилковий стан, як в апаратному, так і в програмному забезпеченні, тобто стан, який відрізняється від нормальних очікувань за існуючих обставин. Крім того, несправність описується як прихована, коли вона не дає помилкового стану. Заходи для виявлення помилок, які можна використовувати у відмовостійкій системі, підпадають під таку широку класифікацію: перевірки повторів; перевірки термінів; зворотні перевірки; перевірки кодування; перевірки обґрунтованості; конструктивні перевірки; діагностичні перевірки.

У процесі розробки систем доцільно підтримувати чітке уявлення про домени помилок і їхні характеристики, як це наведено на рисунку 8.

Поняття "домен" охоплює множину абстракцій помилок, що охоплюють різні рівні функціонування системи і дозволяють конструкторам вирішувати проєктні завдання без зайвих ускладнень. Значення помилки визначає, чи має помилковий стан фіксований характер, чи перебуває у нестабільному, змінному стані.

Розробники, як правило, правильно вибирають рівень абстракції відповідно до типу задач, проте для забезпечення відмовостійкості важливо чітко ідентифікувати основні домени помилок. Фізичний домен охоплює

елементарні апаратні несправності, такі як короткі замикання, обриви чи замикання на землю. Логічний домен стосується збоїв у поведінці пристроїв, наприклад, застрягання сигналу в певному логічному стані або його інверсії. Інформаційний домен відображає помилки в обробці чи інтерпретації даних — зокрема, неправильні значення, зміну знака або помилки парності. Нарешті, системний домен охоплює збої, що впливають на якість або безперервність функціонування системи, включаючи взаємоблокування, системні збої та критичні апаратні відмови.

Зазначені домени утворюють основу ієрархії проєктної відповідальності, визначають механізми забезпечення надійності та порядок дій при реалізації відмовостійкості. Наприклад, фізичний домен є найбільш релевантним для діагностики й локалізації апаратних збоїв. У логічному домені ефективно вирішуються задачі виявлення та аналізу відмов, пов’язаних із функціонуванням елементів. У межах інформаційного домену визначаються методи ідентифікації та виправлення помилок даних, а системний домен забезпечує цілісне функціонування критичних служб навіть за умов часткових збоїв.

Для безпекокритичних застосувань сучасних авіаційних систем найбільшу загрозу становлять не окремі фізичні збої, а так звані синфазні (загальномодові) збої (Common Mode Failures, CMF), що виникають внаслідок несправностей, які одночасно впливають на декілька елементів або підсистем через спільну причину.

Ефективне протидіяння CMF вимагає комплексного підходу, що включає уникнення помилок на етапі проєктування, усунення помилок шляхом тестування, аналізу, моделювання відмов, а також впровадження засобів відмовостійкості, таких як обробники винятків, контрольні точки, відновлення та перезапуск системи. У таблиці 2.5 наведено класифікацію синфазних несправностей, де символ «x» позначає потенційно небезпечні комбінації збоїв, які слід враховувати навіть за відсутності прямого їх проєктування.

Таблиця 2.5

Типологія помилок загального режиму в авіоніці за джерелами та характеристиками прояву

Тип								
Причина		Межа системи		Фаза створення		Тривалість		Помилка загального режиму
Фізичний	Людський фактор	Внутрішній	Зовнішній	Проектування	Оперативний	Постійний	Тимчасовий	
X			X		X			Перехідний (Зовнішній)
X			X		X	X		Постійний (Внутрішній)
	X	X		X			X	Прерівчастий (Проект)
	X	X		X		X		Постійний (Проект)
	X		X		X		X	Взаємодія

Фізичні, внутрішні та операційні збої в технічних системах можуть бути компенсовані за допомогою апаратного резервування. Проте інші типи несправностей здатні впливати одночасно на декілька областей локалізації, що значно ускладнює забезпечення відмовостійкості. Зокрема, слід враховувати чотири основні джерела збоїв загального режиму.

Першим джерелом є перехідні (зовнішні) несправності, які виникають під впливом короткочасних фізичних перешкод із навколишнього середовища, таких як блискавка, радіочастотні впливи високої інтенсивності (HIRF), перегрів, тощо. Другим джерелом є постійні (зовнішні) несправності, які є наслідком тривалого негативного впливу середовища — це, зокрема, висока температура, пісок, солона вода, пил, вібрації, механічні удари.

Ще одним важливим типом несправностей є періодичні помилки, що виникають через недоліки у специфікації вимог, помилки на етапах проєктування, реалізації або інтеграції системи. Вони проявляються нерегулярно, але можуть серйозно вплинути на функціональність. Постійні помилки, на відміну від періодичних, мають стійкий характер прояву, хоча їх джерела аналогічні — вони виникають на тих самих етапах життєвого циклу системи.

Фізична несправність, як подія, що безпосередньо призводить до відмови одного або кількох компонентів, є ключовим джерелом збоїв, які

найчастіше ініціюють інші несприятливі стани. Визначення таких несправностей узагальнено на рисунку 9, де представлено діаграму переходів станів несправності. Ця діаграма ілюструє чотири основні стани: початкову справність, приховану несправність, активну несправність і стан збою. Наприклад, перехід від прихованої до активної несправності може бути зумовлений потенціюючою подією, такою як зміна функціонального режиму системи, що призводить до виявлення наявної помилки.

У випадку виникнення серйозної активної несправності, яка не підлягає відновленню, система переходить у стан збою, коли очікувана функціональність не може бути забезпечена. Якщо наслідки такої несправності не є критичними, система може продовжити функціонування, хоча й із певним погіршенням якості наданих послуг. Водночас реалізація механізмів відмовостійкості дозволяє уникнути повного збою та знизити ризик деградації обслуговування.

Рисунок 2.6 також містить перехід, що описує спонтанне відновлення, коли несправність зникає без зовнішнього втручання. Це свідчить про тимчасовий характер деяких несправностей, які можуть зникати внаслідок, наприклад, самостабілізації електроніки або зміни зовнішніх умов.

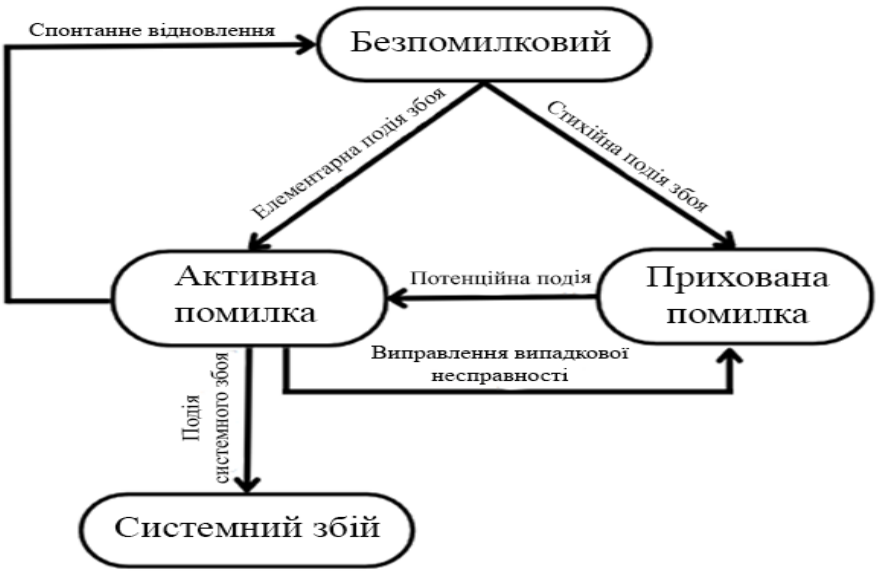


Рис. 2.6 Структура апаратно-програмного стану обладнання .

Це явище може виникнути після того, як зовнішнє «занепокоєння» вщухає або періодична фізична аберація припиняється. Дещо схоже явище

представлено через подію усунення випадкової несправності в таблиці 2.6.

Таблиця 2.6

Розмежування умов несправності

Відновлений режим	Латентний режим	Активний режим
Мимовільне відновлення після збою або граничного фізичного відновлення	-	Помилковий стан, спричинений тимчасовими збуреннями або минущим проявом граничної несправності
-	-	-
-	Ремісія серйозної фізичної несправності або латентність серйозної фізичної несправності	Перехідний прояв серйозної помилки або постійний прояв серйозної помилки

У контексті аналізу динаміки технічного стану системи виокремили три основні режими, що характеризують еволюцію несправностей: відновлений, латентний та активний. Кожен з них має визначений зміст та логічно пов’язані переходи, що відображають процеси деградації або відновлення функціональності.

Відновлений режим описує стан системи після мимовільного усунення несправності або в результаті граничного фізичного відновлення. У цьому стані система вважається працездатною, хоча в минулому могла мати прояви відмов. Відновлення може бути як наслідком зовнішнього втручання (ремонт, перезапуск), так і результатом спонтанного усунення збурення (наприклад, стабілізації електричних параметрів).

Латентний режим відповідає прихованому стану несправності, яка ще не проявила себе у вигляді функціонального порушення, але вже наявна в системі. Такий стан часто є наслідком ремісії серйозної фізичної несправності або її тимчасової локалізації. Латентні відмови є особливо небезпечними, оскільки можуть не бути виявленими під час звичайної експлуатації, але здатні активізуватися під впливом потенціюючих факторів (зміна режиму роботи, стресові умови, вібрація тощо).

Активний режим характеризується відкритим проявом помилки, що впливає на функціональність системи. Помилковий стан може бути

викликаний як тимчасовими збуреннями (наприклад, електромагнітними перешкодами або тепловими імпульсами), так і постійними фізичними пошкодженнями. Перехід у цей режим означає, що несправність більше не є прихованою і призводить до часткової або повної втрати функціональності.

Логіка переходів між режимами наведена на рисунку 2.7, який репрезентує діаграму станів технічної справності. Наприклад, перехід від латентного до активного режиму може відбутися внаслідок зміни експлуатаційного профілю або зовнішнього впливу, що спричиняє реалізацію несправності.

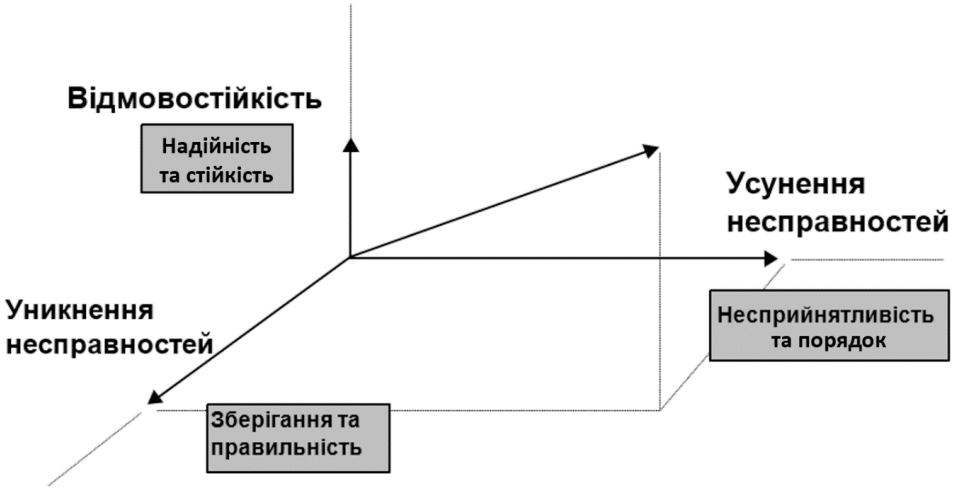


Рис. 2.7. Взаємозвязок надійності яка залежить від уникнення та вирішення несправностей та відмовостійкості.

Водночас у деяких випадках можливий зворотний перехід із активного до відновленого режиму за умови мимовільного усунення несправності, що не потребує зовнішнього втручання. Цей механізм отримав назву спонтанного відновлення.

Вимоги до надійності системи залежать від рівня критичності функції та максимального часу впливу. Критичною для польоту є функція, втрата якої може призвести до втрати самого ПС та людей на борту. В останньому випадку система називається критичною з точки зору безпеки. Критично важлива функція — це функція, втрата якої призведе до компрометації або скасування відповідної місії. Таким чином, постійна критично важлива

система, як правило, висуватиме набагато більші вимоги до надійності, ніж критично важлива система на етапі польоту.

Таким чином, розуміння характеристик і взаємозв’язків між цими режимами є ключовим для розробки стратегій моніторингу, прогнозування стану системи та забезпечення її відмовостійкості, таблиця 2.7.

Таблиця 2.7

Підходи до забезпечення надійності за типами помилок

Сфера забезпечення надійності	**Фізичні вади**	**Помилки розвитку (проектні, програмні)**
Уникнення несправностей	Мінімізація через інженерний аналіз	Запобігання помилкам на етапі розробки
Усунення несправностей	Зменшення частоти появи типових збоїв	Виявлення та усунення помилок шляхом перевірки
Відмовостійкість	Забезпечення за рахунок апаратного резервування	Підвищення живучості за допомогою тестування

Далі вимоги до надійності системи в поєднанні з граничною надійністю компонентів системи визначають рівень резервування для забезпечення живучості від несправностей, тобто мінімальну кількість несправностей, які повинні вижити. Щоб забезпечити відповідність вимогам надійності, еволюція відмовостійкого проекту повинна базуватися на взаємодії між зобов’язаннями щодо конфігурації проекту та обґрунтовуючим аналізом.

Пропонуємо алгоритм, який ілюструє взаємозв'язок між надійністю, уникненням та вирішенням несправностей та відмовостійкістю:

1. Вхідні дані.

Визначення функціональних вимог та потенційних ризиків для системи авіоніки.

Крок 1: Аналіз ризиків. Виявлення потенційних ризиків, пов'язаних з недоліками та відмовами системи. Оцінка ймовірності виникнення цих ризиків та їхнього впливу на безпеку та надійність системи.

Крок 2: Уникнення несправностей.Розробка стратегій та процесів, спрямованих на уникнення можливих несправностей. Використання високоякісних матеріалів та технологій, що дозволяють зменшити ризик

виникнення несправностей.

Крок 3: Вирішення несправностей. Розробка систем виявлення та діагностики несправностей для швидкого виявлення та вирішення проблем.

Реалізація процедур обслуговування та підтримки, що дозволяють ефективно вирішувати виявлені несправності.

Крок 4: Відмовостійкість. Розробка систем резервування та резервних шляхів, що дозволяють системі продовжувати функціонування в умовах відмов. Проведення тестів та симуляцій, що дозволяють перевірити роботу системи в умовах відмов та визначити її відмовостійкість. Вихідні дані: оцінка надійності системи та розробка стратегій для підвищення її надійності та безпеки.

Цей алгоритм надає план дій для забезпечення надійності та безпеки системи авіоніки шляхом уникнення та вирішення несправностей та підвищення відмовостійкості.

Для цивільних, транспортних ПС в рівень резервування для такої критичної функції на етапі польоту, як автоматична посадка в при будь-яких погодних умов, як правило, є одноразовою відмовою, що означає, що система повинна залишатися працездатною після будь-якої потенційної єдиної елементарної відмови. Слід зазначити, що функція, яка сама по собі не є критичною для польоту, може мати режими відмови, які загрожують безпеці польоту. У разі активних засобів керування для зменшення структурних навантажень через пориви або маневрування функція не буде критичною, якщо метою є лише зменшення ефектів структурної втоми. Якщо пов’язані поверхні управління польотом мають право під час несправності хардоверу або розбігу викликати структурні пошкодження, то такий режим відмови є критичним для безпеки. У таких випадках режими відмови повинні бути розроблені так, щоб вони були пасивними, що виключає будь-який ефект відмови активного режиму, як-от контрольна поверхня жорсткого перекриття. Режим відмови, який демонструє активну поведінку, все ще може бути відмовостійким, однак, якщо швидкість і серйозність активних

ефектів знаходяться в межах можливостей льотного екіпажу для безпечного керування.

Практично неможливо спроектувати складну систему авіоніки, яка б витримувала всі можливі несправності. Несправності можуть включати як постійні, так і тимчасові несправності, несправності апаратного та програмного забезпечення, і вони можуть виникати окремо або одночасно. Помилки синхронізації безпосередньо пов’язані з вимогою відповіді в реальному часі протягом кількох мілісекунд і можуть бути пов’язані як з апаратним і програмним забезпеченням, що сприяє затримці даних, так і з неправильними даними.

Реалізація відмовостійкості супроводжується збільшенням накладних витрат на систему, ускладненням її архітектури та виникненням додаткових викликів щодо перевірки та валідації. Накладні витрати, які, по суті, зумовлені розширенням системних ресурсів і відповідними керуючими діями, включають потребу в додатковому обладнанні, засобах зв’язку та підвищених обчислювальних потужностях, рисунок 2.8.

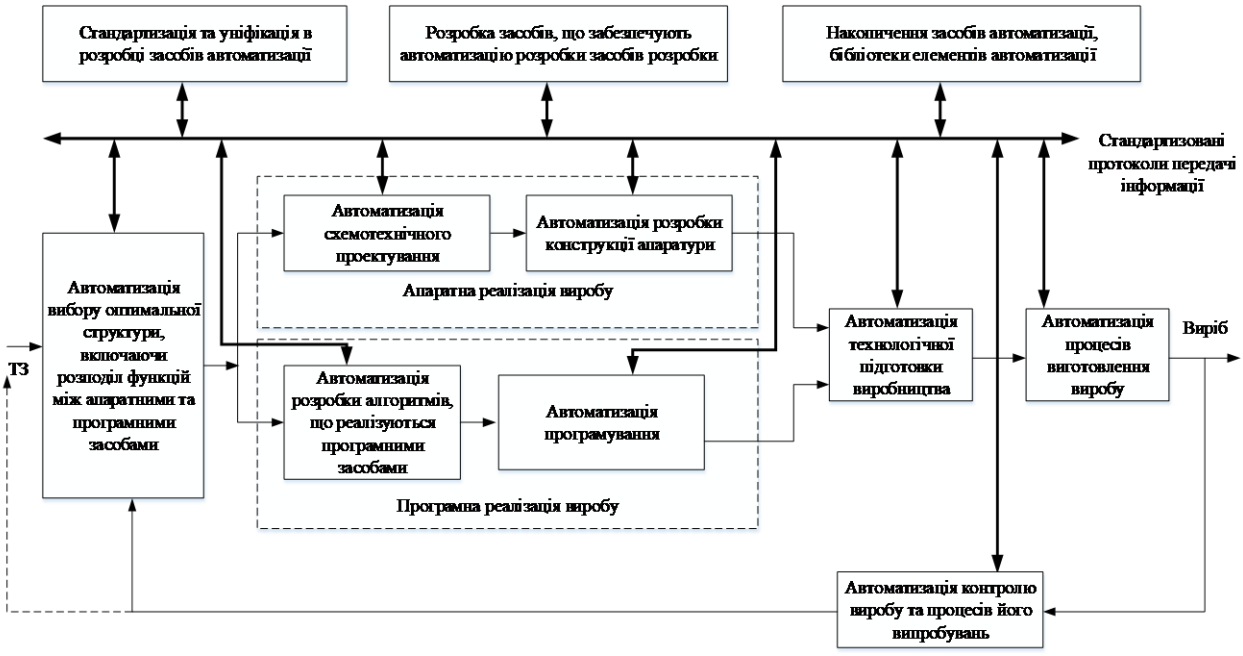


Рис.2.8 Структурна схема автоматизованого процесу розробки та реалізації відмовостійкого виробу авіоніки

Розширена складність походить від складніших зв’язків і залежностей між апаратними та програмними елементами; більша кількість станів системи, які можна припустити; і значно більше задіяної логіки для

керування системою. Проблеми валідації пов’язані з необхідністю виявити, оцінити та підтвердити здатність підтримувати функціональність системи в широкому діапазоні потенційних випадків несправностей.

Отже, підхід до розробки відмовостійкої авіоніки повинен досягати балансу між витратами, понесеними при реалізації відмовостійкості, та ступенем реалізованої надійності.

Підхід до проектування охоплює як системні концепції, методологію розробки, так і елементи відмовостійкості. Концепції системи в основному випливають із основних параметрів відмовостійкості, з наголосом на розумних комбінаціях функцій, адаптованих до заданих атрибутів програми. Методологія розробки зводиться до взаємно підтримуваних методів, керованих впевненістю, які поширюють узгодженість, забезпечують підзвітність і точні високі рівні впевненості щодо надійності системи. Елементи проектування стійкості до відмов мають тенденцію уніфікувати концепції та методи проектування в сенсі забезпечення впорядкованої моделі організації та еволюції системи, таблиця 2.8.

Таблиця 2.8

Компоненти забезпечення відмовостійкості та принципи їхнього функціонування

Елемент відмовостійкості	Виконання
Резервування	Використання дублювання компонентів або систем для забезпечення неперервної роботи в разі відмови.
Аварійні зупинки	Механізми, що автоматично вимикають систему або компонент в разі виявлення небезпечної ситуації.
Самодіагностика	Системи, які можуть виявити свої власні відмови та повідомити про них для подальшого вирішення.
Управління ризиками	Аналіз та управління потенційними ризиками та відмовами для зменшення їх впливу на систему.
Реакція на відмови	Плани та процедури для ефективного реагування на відмови та відновлення роботи системи.

Ці елементи відмовостійкості, які, з'являються в тій чи іншій формі в будь-якій відмовостійкій системі:

– Виявлення помилок — розпізнавання виникнення несправності.

– Оцінка збитків — діагностика місця несправності.
– Усунення несправностей — обмеження обсягу наслідків несправності.
– Відновлення помилок — відновлення безпомилкового стану з можливістю перезапуску.
– Продовження обслуговування — постійне надання системних послуг.
– Усунення несправності.

Основним параметром конструкції, який охоплює ці елементи та обмежує їхню механізацію, є деталізація обробки несправностей. По суті, виявлення, ізоляція та відновлення після несправності мають відбуватися на тому ж рівні модульності, щоб досягти збалансованого та узгодженого дизайну. Загалом, невигідно чи виправдано виділяти або містити помилку на рівні, нижчому за відповідну межу обробки помилок. Однак можуть бути винятки, особливо у випадку виявлення помилок, де може бути застосований вищий ступінь деталізації, щоб скористатися перевагами вбудованих функцій тестування або зменшити затримку помилки.

Залежно від основи для її ініціювання, стримування несправності може включати гальмування розповсюдження фізичної несправності або придушення помилкових обчислень. Фізичне обмеження несправностей має бути розроблено в апаратному забезпеченні, а програмне утримання станів помилок має бути розроблено в прикладному програмному забезпеченні. У більшості випадків помилковий стан програмного забезпечення необхідно виправити через розбіжності прикладної програми, внесені під час затримки виявлення несправності. Це усунення помилок може призвести до скидання певних значень об’єктів даних і зворотного відстеження шляху потоку керування в працездатному процесорі. На цьому етапі готовність основної архітектури, включаючи координацію робочих компонентів, має бути забезпечена інфраструктурою. Як правило, ця діяльність значною мірою залежить від програмного забезпечення для керування системою для відмовостійкості. Таким чином, відновлення починається зі встановлення відповідного стану програм для перезапуску програми. У системі авіоніки ця

послідовність дій щодо відмовостійкості має відбуватися досить швидко через функціональні вимоги в реальному часі. Відповідно, необхідно визначити абсолютний бюджет часу з допусками для найгіршого випадку продуктивності для швидкого продовження обслуговування.

2.3 Засоби реалізації систем підвищення відмовостійкості, контролю і діагностики бортового обладнання

Реалізація алгоритмів підвищення відмовостійкості в сучасних системах IMA базується на розпізнаванні та локалізації відмов авіоніки повітряного судна. Це, як правило, здійснюється за допомогою вбудованих реконфігурованих діагностичних систем, які працюють як у складі окремих модулів, так і на рівні загальної архітектури IMA.

Системи контролю і діагностики (СКД) реалізуються у взаємодії з базовою операційною системою (БОС) відповідно до вимог ARINC 653, та входять до складу програмного забезпечення бортового радіоелектронного обладнання (ПЗ БРЕО). Вони формують ознаки як апаратної, так і функціональної справності авіоніки, а також керують масивами вхідної та вихідної контрольно-вимірювальної інформації.

Апаратна частина діагностичної системи зазвичай реалізується на реконфігурованому обчислювальному полі (наприклад, FPGA або R-SoC). Обчислювальні ресурси такого поля поділені на логічні блоки, до яких належать модулі пам’яті для зберігання конфігурацій і моделей діагностичних сценаріїв.

Альтернативним підходом є використання спеціалізованого обчислювального комплексу (спецобчислювача) у складі IMA. Такий пристрій може працювати автономно або як частина розподіленої архітектури, вирішуючи широке коло обчислювальних і інформаційних задач, у тому числі в реальному часі. Його архітектура, як правило, є багатопроцесорною і включає: обчислювальний пристрій (ОчП), пристрій обміну (ПО), пристрій управління каналом (ПК), довготривалу (ДЗП) та

оперативну (ОЗП) пам’ять, а також внутрішній магістральний канал (ВМК), що забезпечує з’єднання між модулями.

Внутрішній магістральний канал реалізується у вигляді високошвидкісної шини обміну, яка включає магістралі запису (МЗп), читання (МЧт), адресну шину (МА) та сигнальні шини управління. Конфігурація елементів системи визначається програмно, зокрема шляхом зміни вмісту регістрів конфігурації та виконання спеціалізованих програм керування.

Для оцінювання обчислювальних ресурсів використовувався тестовий приклад, у якому програма виконувалась протягом 50 секунд (5001 крок) із частотою вибірки 0,01 с, що дало сумарно 224 744 224 обчислювальні цикли, або приблизно 44 940 циклів на кожен крок, таблиця 2.9.

Таблиця 2.9

Оцінювання ефективності обчислювальних алгоритмів корекції навігаційних даних за точністю та швидкодією

Обчислювальний алгоритм	**Точність корекції, %**	**Тривалість корекції, с**	**Інтервал формування вхідної вибірки, с**
ФК (Фільтр Калмана)	65	0.01	0.80
АФК (Адаптивний ФК)	70	0.01	0.80
НФК (Нелінійний ФК)	85	0.50	0.80
НАФК (Нелінійний адаптивний ФК)	88	0.75	1.50
МГУА	93	1.50	3.00
Нейромережева модель	94	3.00	3.00
Генетичний алгоритм (ГА)	95	27.50	3.00
ДЕС (Дескриптивна експертна система)	96	30.00	6.00

У межах реалізації математичних моделей для корекції інерціальної навігаційної системи (ІНС) застосовувались алгоритми фільтрації: фільтр Калмана (ФК), адаптивний фільтр Калмана (АФК), нелінійний фільтр Калмана (НФК), а також нелінійний адаптивний фільтр Калмана (НАФК). Ці фільтри забезпечують оцінку похибок та корекцію ІНС з оновленням даних GPS з періодичністю 1 с.

Інші реалізовані модулі включали математичні моделі на основі методу групового урахування аргументів (МГУА), нейронних мереж та генетичних алгоритмів (ГА). Для цих методів передбачалося оновлення вхідної вибірки кожні 3 секунди. Додатково, для системи діагностики й корекції було реалізовано модуль дескриптивної експертної системи (ДЕС), яка формує нову вибірку кожні 6 секунд. Усі зазначені модулі реалізовані на основі продукційної системи CLIPS, що дозволяє ефективно формалізувати правила діагностики та реагування.

Відмінною особливістю дескриптивної експертної системи (ДЕС) є те, що вона функціонує під управлінням операційної системи (ОС), і для її повної інсталяції необхідно не менше ніж 16 МБ оперативної пам’яті (ОЗП) персонального комп’ютера. Реалізація ДЕС та відповідних алгоритмів підвищення відмовостійкості передбачається безпосередньо на борту повітряного судна, з урахуванням вимог до обчислювальних ресурсів та режиму реального часу.

У межах цього дослідження також було розглянуто Марковський ланцюг як математичну модель, що описує процес технічної експлуатації авіаційної системи. Формалізація задачі марковського ланцюга починається з визначення множини станів SSS, яка репрезентує можливі стани технічного об'єкта. У даному випадку ми обмежимося розглядом кінцевої множини станів S, розмірність якої дорівнює n, де кожен стан відповідає певному технічному або функціональному рівню працездатності системи.

$$\mathbf{S} = \{S1, S2, \cdots, Si, \cdots, Sj, \cdots, Sn\}. \quad (2.1)$$

У початковий момент часу l = 0 має бути задано розподіл ймовірностей станів: $\mathbf{P}(0) = \{P1(0), P2(0), \cdots, Pi(0), \cdots, Pj(0), \cdots, Pn(0)\}$, (2.2)

при чому, $\sum_{i=1}^{n} P_i^{(0)} = 1$

Верхній індекс у дужках тут і надалі позначатиме момент часу.

У окремому випадку $P^{(0)}$ може являти собою вироджений одиничний вектор $P^{(0)} = \{0,0, \cdots, 1, \cdots, 0\}$. (2.3)

Якщо момент часу l ланцюг перебуває у стані $S_i \in \mathbf{S}$, то наступна еволюція СМЦ задається матрицею перехідних ймовірностей

$$P_1^S = \{p, (Sj / Si), Sj, Si, \in \mathbf{S}\}, \quad (2.4)$$

елементи якої залежать тільки від поточного стану S_i і не залежать від попередніх еволюцій і моменту часу l. Таким чином, ми визначили стаціонарний марківський ланцюг 1-го порядку, який являє собою сукупність

$$\xi_1^s = \{\mathbf{S}, \mathbf{P}^{(0)}, \mathbf{P}_1^s\} \quad (2.5)$$

Реалізація СМЦ $\xi^S{}_l$ має вигляд $h_{\xi_1^s}(m) = S_{i_0}^{(0)}, S_{i_1}^{(1)}, \cdots, S_{i_m}^{(m)}.$ (2.6)

Номер кроку марківського ланцюга є тимчасовим параметром. Запис $S_{im}^{(m)}$ означає стан ланцюга $S_{i0} \in \mathbf{S}$ у момент часу m. Реалізації h_ξ (m) відповідає ймовірність $P\{h_{\xi_1^s}(m)\} = P_{i_0}^{(0)} p\left(S_{i_1}^{(1)} / S_{i_0}^{(0)}\right) \cdots p\left(S_{i_m}^{(m)} / S_{i_{m-1}}^{(m-1)}\right)$ (2.7)

Позначимо через $p^{(m)}$ (Sj / Si) умовну ймовірність попадання СМЦ у стан Sj на m-му кроці, якщо початковим станом було S_i. Вона дорівнює сумі ймовірностей всіх реалізацій довжини m, що починаються в S_i і закінчуються в S_j.Зокрема, $p^{(1)}(S_j / S_i) = p(S_j / S_i)$, (2.8)

$$p^{(2)}(S_j / S_i) = \sum_{k=1}^{n} p(S_k / S_i) p(S_j / S_k).$$

За індукцією отримуємо загальну рекурентну формулу Колмогорова-Чепмена для СМЦ

$$p^{(m)}(S_j / S_i) = \sum_{k=1}^{n} p^{(1)}(S_k / S_i) p^{(m-1)}(S_j / S_k). \quad (2.9)$$

Рівняння (2.9) відображає той факт, що перші l кроків приводять СМЦ у стан S_k і що ймовірність наступного переходу зі S_k в S_j не залежить від того, яким чином було досягнуто S_k.

Отже безумовну ймовірність попадання у стан S_j на m-му кроці можна визначити за допомогою наступної рекурентної формули

$$p^{(m)}(S_j) = \sum_{k=1}^{n} p^{(0)}(S_k) p^{(m)}(S_j / S_k). \quad (2.10)$$

Стан S_j можна досягти зі стану S_i, якщо існує таке $m \geq 0$, що $p^{(m)}(S_j / S_k) > 0$, тобто. якщо є позитивна ймовірність потрапити зі стану Si

стан Sj. Марківський ланцюг є непривідним, коли кожен її стан досяжний з будь-якого іншого стану. Позначимо через $f^{(m)}(S_j / S_k)$ ймовірність того, що в процесі, що починається з S_i перше попадання в S_j відбудеться на m-му кроці. Покладемо $f^{(0)}(S_j/S_i) = 0$

$$f^{(\infty)}(S_j / S_i) = \sum_{1=1}^{\infty} f^{(1)}(S_j / S_i). \tag{2.11}$$

Імовірність $f^{(\infty)}(S_j / S_i)$ – є ймовірність того, що, виходячи з S_i процес коли-небудь пройде через S_j. Для обчислення $f^{(m)}(S_j/S_i)$ скористаємося наступними міркуваннями. Якщо перше досягнення S_j здійснюється при l - му переході ($1 \le l \le m - 1$), то умовна ймовірність попадання в S_j при m -му переході дорівнюватиме $p^{(m-1)}(S_j / S_i)$. Беручи до уваги, що $p^{(0)}(Sj / Si) = 1$, отримаємо

$$p^{(m)}(S_j / S_i) = \sum_{1=1}^{n} f^{(1)}(S_j / S_i) p^{(m-1)}(S_j / S_i). \tag{2.12}$$

Послідовно вважаючи $l = 1, 2, \cdots, n$, ми за формулою (2.12) отримаємо значення $p^{(l)}(S_j / S_i)$, які підставимо в (144) і послідовно отримаємо $f^{(l)}(S_j / S_i)$, $l = 1, m$. Тоді середній час повернення процесу $\mu(S_j)$ у стан S_j можна визначити наступним чином

$$\mu(S_j) = \sum_{1=1}^{\infty} 1 f^{(1)}(S_j / S_j). \tag{2.13}$$

Стан S_j зворотний, якщо $f^{(\infty)}(S_j / S_i) = 1$, і незворотний, якщо $f^{(\infty)}(S_j / S_i) < 1$. Стан S_j має період $t > 1$, якщо $p^{(m)}(S_j / S_i) = 0$, коли m не є кратним t і t на t. Стан S_j є неперіодичним, якщо такого не існує. Неперіодичний зворотний стан S_j, у якого $\mu(S_j) < \infty$, є ергодичним. Дуже важливим є наступний результат. У ергодичному ланцюгу, що не приводиться, існують не залежать від початкового стану S_i межі

$$\pi(Sj) = \lim p^{(m)}(Sj / Si),\ Sj \in \mathbf{S} \tag{2.14}$$

причому $\pi(S_j) > 0$

$$\pi(S_j) = \sum_{i=1}^{n} \pi(S_i) p(S_j / S_i),\ S_j \in \mathbf{S}, \tag{2.15}$$

де $\sum_{i=1}^{n} \pi(S_i) = 1,$ $\mu(Sj) = 1/\pi(Sj).$

Розподіл ймовірностей $\{\pi(S_j)\}$ є стаціонарним розподілом ймовірностей станів МЦ. Для нестаціонарного марківського ланцюга матриця перехідних ймовірностей набуває наступного вигляду : $\mathfrak{R}_1^{NS} = \{P_1^{NS}, 1 = \overline{0, m}\},$ (2.16)

де $\mathbf{P}_l^{NS} = \{p(S_j / S_i, l), S_i, S_j \in \mathbf{S}\}$ - матриці, елементи яких залежать від моментів часу l. Таким чином, НСМЦ 1-го порядку є сукупністю

$\xi_1^{NS} = \{\mathbf{S}, \mathbf{P}^{(0)}, \mathfrak{R}_1^{NS}\}.$ (2.17)

Реалізації $h_{\xi_1^{NS}}(m)$ відповідає ймовірність:

$$P\left\{h_{\xi_1^{NS}}(m)\right\} = P_{i_0}^{(0)} p\left(S_{i_1}^{(1)}/S_{i_0}^{(0)}, 0\right) \dots p\left(S_{i_m}^{(m)}/S_{i_{m-1}}^{(m-1)}, m-1\right). \quad (2.18)$$

Умовна ймовірність $p^{(m)}(S_j / S_i, m)$ потрапляння НСМЦ у стан S_j на m-му кроці, якщо початковим станом було S_i, дорівнює сумі ймовірностей всіх реалізацій довжини m, що починаються в S_i і закінчуються в S_j:

$p^{(1)}(S_j / S_i, 1) = p(S_j / S_i, 1);$ (2.19)

$$p^{(2)}(S_j / S_i, 2) = \sum_{k=1}^{m} p^{(1)}(S_k / S_i, 1) p(S_j / S_k, 2) \cdot$$

Далі по індукції отримуємо загальну рекурентну формулу Колмогорова-Чепмена для НСМЦ

$$p^{(m)}(S_j / S_i, m) = \sum_{k=1}^{m} p^{(1)}(S_k / S_i, 1) p^{m-1}(S_j / S_k, m-1) \cdot \quad (2.20)$$

Безумовна ймовірність попадання НСМЦ у стан S_j на m-му кроці визначається наступним чином

$$p^{(m)}(S_j, m) = \sum_{k=1}^{m} p^{(0)}(S_k, 1) p^{m}(S_j / S_k, m) \cdot \quad (2.21)$$

Позначимо через $f^{(m)}(S_j / S_i, m)$ ймовірність того, що в нестаціонарному процесі, що починається з S_i, перше потрапляння в S_j відбудеться на m-му кроці. Для обчислення цієї ймовірності скористаємося за аналогією з СМЦ виразом:

$$p^{(m)}(S_j / S_i, m) = \sum_{k=1}^{m} f^{(1)}(S_j/S_i, 1)p^{m-1}(S_j/S_k, m-1) \cdot \quad (2.22)$$

Тоді середній час повернення нестаціонарного процесу $\mu_{\xi_1^{NS}}(S_j)$ в стані S_j визначається наступним чином :

$$\mu_{\xi_1^{NS}}(S_j) = \sum_{1=1}^{\infty} 1f^{(1)}(S_j / S_i, 1) \cdot \quad (2.23)$$

Розглянемо тепер марківський стаціонарний ланцюг k-го порядку. Вона задається сукупністю

$$\xi_k^s = \{\mathbf{S}, \mathbf{P}^{(0)}, \mathfrak{R}_k^s\}, \quad (2.24)$$

де $\mathfrak{R}_k^s = \{P_1^s, l = 1, k\}$ – сукупність матриць перехідних ймовірностей від 1-го до k-го порядку, елементи яких залежать від попередніх станів.

$$P_1^s = \{p(S_{i_1}/S_{i_{1-1}}, \dots, S_{i_0}), s_{i_1}, S_{i_{1-1}}, \dots, S_{i_0} \in \mathbf{S}\} \quad (2.25)$$

Реалізації $h_{\xi_k^s}$ відповідає ймовірність:

$$P\{h_{\xi_1^s}(m)\} = P_{i_0}^{(0)} p\left(S_{i_1}^{(1)} / S_{i_0}^{(0)}\right) \dots p\left(S_{i_k}^{(k)}/S_{i_{k-1}}^{(k-1)}, \dots, S_{i_0}^{(0)}\right) \dots p\left(S_{i_k}^{(m)}/S_{i_{m-1}}^{(m-1)}, \dots, S_{i_{m-k}}^{(m-k)}\right). \quad (2.26)$$

Позначимо через $p^{(m)}$ $(S_{i_m} / S_{i_{m-1}}, \dots, S_{i_{m-k}})$ умовну ймовірність попадання СМЦВ у стан S_{i_m} на m-му кроці, якщо початковим станом у момент часу $(m - k)$ було $S_{i_{m-k}}$:

$$p^{(1)}(S_j / S_i) = p(S_{i_1} / S_{i_0});$$

$$p^{(2)}(S_{i_2} / S_{i_1}, S_{i_0}) = \sum_{i_1=1}^{n} p^{(1)}(S_{i_1} / S_{i_0}) p(S_{i_2}, S_{i_1}, S_{i_0});$$

$$p^{(k)}(S_{i_k}/S_{i_{k-1}}, \dots, S_{i_0}) = \sum_{i_{k-1}=1}^{n} p^{(k-1)}(S_{i_k} / S_{i_{k-1}}, \dots, S_{i_0}) p(S_{i_k}, S_{i_{k-1}}, \dots, S_{i_0}); \quad (2.27)$$

$$p^{(m)}(S_{i_m}/S_{i_{m-1}}, \dots, S_{i_{m-k}}) = \sum_{i_{m-1}=1}^{n} p^{(m-1)}(S_{i_{m-1}} / S_{i_{m-1}}, \dots, S_{i_{m-k}}) p(S_{i_m}, S_{i_{m-1}}, \dots, S_{i_{m-k}}).$$

Якщо k = m, то отримуємо стаціонарний немарківський ланцюг, переходи якого залежать від усіх попередніх еволюцій.

З (160) випливає, що безумовна ймовірність потрапляння у стан S_{i_m} на m-му кроці можна визначити за допомогою наступної рекурентної формули

$$P^{(m)}(S_{i_m}) = \sum_{i_0=1}^{n} P^{(0)}(S_{i_0}) p^{(m)}(S_{i_m}, S_{i_{m-1}}, \dots, S_{i_0}). \quad (2.28)$$

Позначимо через $f^{(m)}(S_{i_m}, S_{i_{m-1}}, \dots, S_{i_0})$ ймовірність того, що в процесі, що починається з S_{i_0}, перше потрапляння в S_{i_m} відбудеться на m-му кроці. Для цієї ймовірності справедливий вираз (52)

$$f^{(m)}(S_{i_m}/S_{i_{m-1}}, \dots, S_{i_0}) = \sum_{1=1}^{n} f^{(1)}(S_{i_m} / S_{i_{m-1}}, \dots, S_{i_0}) p^{(m-1)}(S_{i_m}, S_{i_{m-1}}, \dots, S_{i_0}).$$

Тоді середній час повернення $\mu_{\xi_1^{NS}}(S_i)$ СМЦВ у стан S_i визначається наступним чином

$$\mu_{\xi_1^{NS}}(S_i) = \sum_{1=1}^{\infty} 1 f^{(1)}(S_{i_1} / S_{i_{1-1}}, \dots, S_{i_{1-k}}) \cdot \quad (2.29)$$

На закінчення розглянемо нестаціонарний марківський ланцюг k-го порядку. Вона задається сукупністю $\xi_k^{NS} = \{\mathbf{S}, \mathbf{P}^{(0)}, \mathfrak{R}_k^{NS}(1)\}$, (2.30)

де , $\mathfrak{R}_k^{NS} = \{P_r^S (l), l = 1, m; r = 1, k\}$ - сукупність матриць перехідних ймовірностей від 1-го до k-го порядку, елементи яких залежать від попередніх станів і від моментів часу l

$$P_1^{NS}(1) = \{p(S_{i_k}/S_{i_{k-1}}, 1_{k-1}, \dots, S_{i_0}, 1_0), S_{i_k}, S_{i_{k-1}}, \dots, S_{i_0} \in \mathbf{S}\} \quad (2.31)$$

Реалізації НСМЦВ $h_{\xi_k^{NS}}(m)$ відповідає ймовірність (2.32)

$$P\{h_{\xi_1^{NS}}(m)\} = P_{i_0}^{(0)} p(S_{i_1}^{(1)}/S_{i_0}^{(0)}, 0) \dots p(S_{i_k}^{(k)}/S_{i_{k-1}}^{(k-1)}, (k-1), \dots, S_{i_0}^{(0)}, 0) \dots p(S_{i_k}^{(m)}/S_{i_{m-1}}^{(m-1)}, (m-1) \dots, S_{i_{m-k}}^{(m-k)}, (m-k)).$$

Таким чином, ми розглянули основні різновиди марківських ланцюгів як можливих моделей процесів технічної експлуатації авіоніки. Стаціонарна марківська модель процесу технічної експлуатації Визначення аналітичних функцій інтенсивності відмов та ймовірності безвідмовної роботи авіоніки за статистичними даними наведено. Нехай об'єкт експлуатації може бути в 2-х станах: S_1 – працездатний стан використання за призначенням; S_2 – непрацездатний стан, у якому проводиться відновлення працездатності.

У час l = 0 об'єкт перебуває у S_2 , тобто,

$P^{(0)} = (0,1)$. Матриця перехідних імовірностей має вигляд $P_1^S = \begin{bmatrix} 0,99 & 0,01 \\ 1 & 0 \end{bmatrix}$.

Визначимо ймовірність попадання об'єкта у стани S_1 та S_2 за 4 кроки:

$p^{(1)}(S_1 / S_2) = 1,$

$p^{(1)}(S_2 / S_2) = 0,$

$p^{(2)}(S_1 / S_2) = p^{(1)}(S_1 / S_2)\, p(S_1 / S_2) + p^{(1)}(S_2 / S_2)\, p(S_1 / S_2) = 0,99;$

$p^{(2)}(S_2 / S_2) = p^{(1)}(S_1 / S_2)\, p(S_1 / S_2) + p^{(1)}(S_2 / S_2)\, p(S_2 / S_2) = 0,01;$

$p^{(3)}(S_1 / S_2) = p^{(2)}(S_1 / S_2)\, p(S_1 / S_1) + p^{(2)}(S_2 / S_2)\, p(S_1 / S_2) = 0,9901;$

$p^{(3)}(S_2 / S_2) = p^{(2)}(S_1 / S_2)\, p(S_1 / S_2) + p^{(2)}(S_2 / S_2)(S_2 / S_2) = 0,0099;$

$p^{(4)}(S_1 / S_2) = p^{(3)}(S_1 / S_2)\, p(S_1 / S_1) + p^{(3)}(S_2 / S_2)\, p(S_1 / S_2) = 0,990099;$

$p^{(4)}(S_2 / S_2) = p^{(3)}(S_1 / S_2)\, p(S_1 / S_2) + p^{(3)}(S_2 / S_2)(S_2 / S_2) = 0,009901;$

Визначимо безумовні ймовірності попадання об'єкта у стани S_1 та S_2 за 4 кроки:

$p^{(1)}(S_1) = p^{(0)}(S_1)\, p(S_1 / S_1) + p^{(0)}(S_2)\, p(S_1 / S_2) = 1;$

$p^{(1)}(S_2) = 0;$

$p^{(2)}(S_1) = p^{(1)}(S_1)\, p(S_1 / S_1) + p^{(1)}(S_2)\, p(S_1 / S_2) = 0,99;$

$p^{(2)}(S_2) = 0,01;$

$p^{(3)}(S_1) = p^{(2)}(S_1)\, p(S_1 / S_1) + p^{(2)}(S_2)\, p(S_1 / S_2) = 0,9901;$

$p^{(3)}(S_2) = 0,0099;$

$p^{(4)}(S_1) = p^{(3)}(S_1)\, p(S_1 / S_1) + p^{(3)}(S_2)\, p(S_1 / S_2) = 0,990099;$

$p^{(4)}(S_2) = 0,009901.$

Обчислимо ймовірності того, що в процесі, що почався в стані S_2, перше попадання в S_2 відбудеться на $l = 1,4$ кроці:

$$f^{(1)}(S_2 / S_2) = \frac{p^{(1)}(S_2 / S_2)}{p^{(0)}(S_2 / S_2)} = 0;$$

$$f^{(2)}(S_2 / S_2) = \frac{p^{(2)}(S_2 / S_2) - f^{(1)}(S_2 / S_2)\, p^{(1)}(S_2 / S_2)}{p^{(0)}(S_2 / S_2)} = 0,01;$$

$$f^{(3)}(S_2 / S_2) = \frac{\{p^{(3)}(S_2 / S_2) - f^{(1)}(S_2 / S_2)\, p^{(2)}(S_2 / S_2) - f^{(2)}(S_2 / S_2)\, p^{(1)}(S_1 / S_2)\}}{p^{(0)}(S_2 / S_2)} =$$

0,0099;

$f^{(4)}(S_2 / S_2) =$

$$\frac{\{p^{(4)}(S_2/S_2) - f^{(1)}(S_2/S_2)p^{(3)}(S_2/S_2) - f^{(2)}(S_2/S_2)p^{(2)}(S_1/S_2) - f^{(3)}(S_2/S_2)p^{(1)}(S_2/S_2)\}}{p^{(0)}(S_2/S_2)} =$$

0,009801;

Середній час повернення об'єкта у стан S_2 визначимо за формулою

$$\mu(S_2) = 1 * f^{(1)}(S_2/S_2) + 2 * f^{(2)}(S_2/S_2) + 3 * f^{(3)}(S_2/S_2) + 4 * f^{(4)}(S_2/S_2)$$
$$= 1 * 2 + 2 * 0{,}01 + 3 * 0{,}0099 + 4 * 0{,}009801$$
$$= 2 * 0{,}01 + \sum_{1=3}^{\infty} (0{,}99)^{1-1}\, 0{,}01 * 1$$

Стаціонарні ймовірності станів S_1 і S_2 визначимо з (2.14) та (2.15):

$\pi(S_1) = \pi(S_1)\, p\,(S_1 / S_1) + \pi(S_2)\, p\,(S_1 / S_2)$,

$\pi(S_1) + \pi(S_2) = 1$,

$\pi(S_1) = \pi(S_1)\, p\,(S_1 / S_1) + (1 - \pi(S_1)\, p\,(S_1 / S_2)$,

$\pi(S_1) = \frac{p\,(S_1/S_2)}{1 - p\,(S_1/S_2) + p\,(S_1/S_2)} = \frac{1}{1 - 0.99 + 1} = \frac{1}{1.01} = 0{,}99(0099)$;

$\pi(S_2) = 0.00(9901)$.

Середній час $\mu(S_2)$ можна визначити так

$\mu(S_2) = \frac{1}{\pi(S_2)} = \frac{1}{0.00(9901)} \approx 100{,}91$.

Нестаціонарна марківська модель процесу технічної експлуатації

Припустимо, що матриці перехідних ймовірностей мають вигляд:

$p_1^{NS} = \begin{bmatrix} 0{,}99 & 0{,}01 \\ 1 & 0 \end{bmatrix}$; $p_2^{NS} = \begin{bmatrix} 0{,}98 & 0{,}01 \\ 1 & 0 \end{bmatrix}$;

$p_3^{NS} = \begin{bmatrix} 0{,}97 & 0{,}03 \\ 1 & 0 \end{bmatrix}$; $p_4^{NS} = \begin{bmatrix} 0{,}96 & 0{,}04 \\ 1 & 0 \end{bmatrix}$;

Тобто поведінка об'єкта описується НСМЦ. Визначимо ймовірність попадання об'єкта у стани S_1 та S_2 за 4 кроки:

$p^{(1)}(S_1 / S_2, 1) = 1$,

$p^{(1)}(S_2 / S_2, 1) = 0$,

$p^{(2)}(S_1 / S_2, 2) = p^{(1)}(S_1 / S_2, 1)\, p\,(S_1 / S_2, 2) + p^{(1)}(S_2 / S_2, 1)\, p\,(S_1 / S_2, 2) = 1 * 0{,}98 = 0{,}98$;

$p^{(2)}(S_2 / S_2, 2) = 1 - 0{,}98 = 0{,}02$;

$p^{(3)}(S_2 / S_2, 3) = p^{(2)}(S_2 / S_2, 2)\, p(S_1 / S_1, 3) + p^{(2)}(S_2 / S_2, 2)\, p(S_1 / S_2, 3) = 0{,}98 * 0{,}97 + 0{,}02 * 1 = 0{,}9706;$

$p^{(3)}(S_2 / S_2, 3) = 1 - 0{,}9706 = 0{,}0294;$

$p^{(4)}(S_1 / S_2, 4) = p^{(3)}(S_1 / S_2, 3)\, p(S_1 / S_1, 4) + p^{(3)}(S_2 / S_2, 3)\, p(S_1 / S_2, 4) == 0{,}9706 * 0{,}96 + 0{,}294 * 1 = 0{,}961176;$

$p^{(4)}(S_2 / S_2, 4) = 1 - p^{(4)}(S_1 / S_2, 4).$

Обчислимо ймовірності того, що в процесі, що почався в стані S_1, перше попадання в S_2 відбудеться на $l = 1{,}4$ кроці:

$$f^{(1)}(S_2 / S_2, 1) = \frac{p^{(1)}(S_2 / S_2,1)}{p^{(0)}(S_2 / S_2,0)} = 0;$$

$$f^{(2)}(S_2 / S_2, 2) = \frac{p^{(2)}(S_2 / S_2,2) - f^{(1)}(S_2 / S_2,1)p^{(1)}(S_2 / S_2,1)}{p^{(0)}(S_2 / S_2,0)} = 0{,}02;$$

$$f^{(3)}(S_2 / S_2, 3) = \frac{\{p^{(3)}(S_2 / S_2,3) - f^{(1)}(S_2 / S_2,1)p^{(2)}(S_2 / S_2,2) - f^{(2)}(S_2 / S_2,2)p^{(1)}(S_2 / S_2,1)\}}{p^{(0)}(S_2 / S_2,0)} = 0{,}0294;$$

$$f^{(4)}(S_2 / S_2, 4) = \frac{\{p^{(4)}(S_2 / S_2,4) - f^{(1)}(S_2 / S_2,1)p^{(3)}(S_2 / S_2,3) - f^{(2)}(S_2 / S_2,2)p^{(2)}(S_2 / S_2,2) - f^{(3)}(S_2 / S_2,3)p^{(1)}(S_2 / S_2,1)\}}{p^{(0)}(S_2 / S_2,0)} = 0{,}0306;$$

Отже, було проведено моделювання процесу технічної експлуатації бортового радіоелектронного обладнання (БРЕО) із застосуванням апарату стаціонарних та нестаціонарних марковських ланцюгів (МЛ), що дозволило формалізувати ймовірнісну динаміку переходів між робочими станами авіонної системи.

На основі стаціонарної моделі з двома дискретними станами S1 (працездатний стан) та S2 (відмовний стан) визначено стаціонарні ймовірності перебування системи в кожному з цих станів. Розв’язання системи рівнянь показало, що значення $\pi(S1) = 0.9900$ і $\pi(S2) = 0.0100$, що свідчить про високий ступінь надійності системи у тривалому часовому масштабі. Окрім того, було обчислено середній час перебування системи у стані S2, який склав приблизно $\mu(S2) \approx 100.91$ умовних кроків, що також підтверджує незначну частку часу простою внаслідок відмов.

Нестаціонарну модель реалізовано у вигляді неперервного ланцюга з часово-залежними матрицями перехідних ймовірностей. На основі покрокового аналізу ймовірностей досягнення станів S1 та S2 протягом чотирьох кроків було продемонстровано зміну поведінки об’єкта експлуатації за умов змінної інтенсивності переходів. У процесі розрахунків було отримано значення ймовірностей досягнення стану S2 на різних кроках: f(2)(S2|S2) = 0.02, f(3)(S2|S2) = 0.0294, f(4)(S2|S2) = 0.0306, що дозволяє оцінити розподіл часу до першого входження у стан відмови з моменту запуску системи. Такий підхід є особливо цінним для задач прогнозування технічного стану системи в реальному експлуатаційному середовищі.

Загалом, проведене дослідження підтвердило доцільність використання марківських процесів для моделювання процесів технічної експлуатації авіоніки. Стаціонарна модель дозволяє виконати узагальнене оцінювання надійності у сталих умовах, у той час як нестаціонарна модель забезпечує можливість аналізу системи в умовах змінних факторів навколишнього середовища або режимів функціонування.Таким чином, сформована основоположна математична база для подальшого синтезу адаптивних методів технічної діагностики та оцінки надійності в інтелектуальних системах підтримки прийняття рішень у складі ІМА.

2.4 Розробка діагностичної моделі КБО для етапів проектування та експлуатації.

Потреба в новій базовій моделі обумовлена необхідністю поліпшення показників і характеристик технічного діагностування електронного модуля бортового обчислювача. Класичні базові моделі ґрунтуються на блочній структурі досліджуваного об'єкта, а в разі неможливості виділення такої структури - на графі причинно-наслідкових зв'язків [34, 52, 101].

Бортові обчислювальні системи, що розглядаються, складаються з декількох конструктивно закінчених електронних модулів. У цьому випадку вони теж можуть бути представлені блочною моделлю. Однак, така модель

більше підходить для етапу експлуатації.

Для етапів розробки, налагодження і виготовлення існує необхідність в інший базової моделі. Перед тестуванням системи обов'язково тестується кожен модуль окремо. Кожен модуль створюється, починаючи зі схеми електричної принципової. Більшість сучасних схем створюються в тій чи іншій САПР. Потім конструкторам-розробникам друкованої плати передається ця схема, перелік елементів і, найголовніше, список з'єднань або список ланцюгів.

Список ланцюгів формується САПР після закінчення розробки схеми і є однозначним описом цієї схеми. Різні САПР створюють списки в різних форматах. Список з'єднань в форматі за стандартом Protel99 1.1 представлений на рис. 29. Спочатку в квадратних дужках перераховуються компоненти: мікросхеми, резистори, конденсатори, діоди та ін. Вказується їх позначення в схемі (DD1, R5, C14, VD2...), типи корпусів, найменування, процентні відхилення та ін. Потім в круглих дужках перераховуються ланцюги. Перший рядок після круглої дужки - найменування ланцюга (NET00009, NET00003, GND, VCC, ADDR1, DATA7...). Наступні рядки аж до круглої дужки, що закриває - контактні площадки компонентів, з'єднаних даним ланцюгом.

```
[
DD1
4244.256-3
1986VE8T
]
…
(
NET00009
R36-2
S10-3
)
(
NET00003
DA2-4
C9-1
)
…
(
…
)
```

Рис. 2.9. Фрагмент списку з'єднань за стандартом Protel99 1.1

Фактично, цей список є моделлю електронного модуля з точністю до

контактного випадку (майданчику).

Вивчивши структуру в формі списку з'єднань, можна відзначити, що цей список встановлює відповідність U між двома множинами: множиною ланцюгів N в схемі і множиною контактних майданчиків P електронних компонентів схеми. Тоді цю модель умовно можна зобразити у вигляді двудольного графа $G = (N, P, U)$, як показано на рисунок 2.10

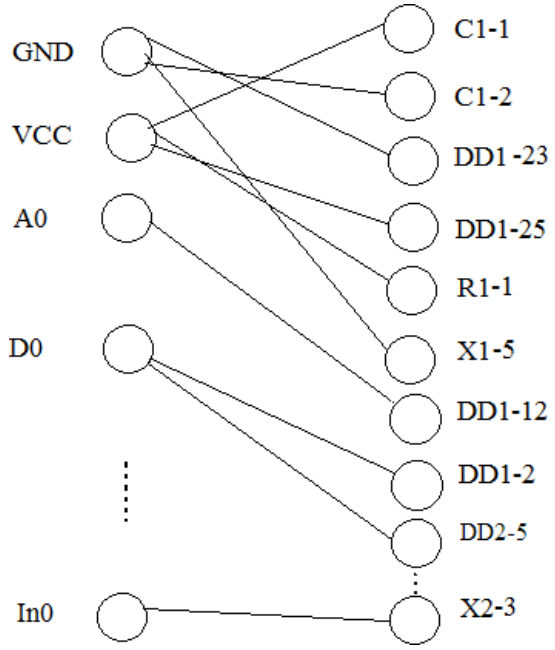


Рис. 2.10. Двочастковий граф ланцюгів і контактних майданчиків

Необхідні відомості з теорії графів наводяться згідно [104].

Граф G - це впорядкована пара множин $G(V, E)$, де V - це множина вершин, а E - безліч ребер.

Дводольний граф визначається як звичайний граф з хроматичним числом не більше двох і заданим правильним розфарбуванням вершин в два кольори. Іншими словами, граф є дводольним, якщо існує таке розбиття множини його вершин V, що $N \cup P = V$ і $N \cap P = \emptyset$, і ребра U можуть з'єднувати тільки вершини N з вершинами P.

Відносини між елементами графа можуть бути виражені в матричному вигляді.

Матрицею суміжності називається матриця (4)

$$A = \|a_{ij}\|, i, j = \overline{1, |P|}, \tag{2.33}$$

елементи якої будуть визначатися відповідно до виразу:

$$a_{ij} = \begin{cases} 0, \text{ якщо ці вершини не суміжні} \\ 1, \text{ якщо вершини } p_i \text{ та } p_j \text{ суміжні (з'єднані ребром),} \end{cases}$$

Отже, базова модель у вигляді двудольного графа, згідно рисунку 2.9, вершини в лівій долі графа відповідають елементам множинності ланцюгів електронного модуля $\boldsymbol{n_i} \in \boldsymbol{N}, \boldsymbol{i} = \overline{\boldsymbol{1, |N|}}$, вершини справа - елементам множини контактних майданчиків $\boldsymbol{p_j} \in \boldsymbol{P}, \boldsymbol{j} = \overline{\boldsymbol{1, |P|}}$. Ребра графу $\boldsymbol{u_k} \in \boldsymbol{U}, \boldsymbol{k} = \overline{\boldsymbol{1, |U|}}$ відображають відповідність ланцюгів контактних площадок згідно зі схемою електричної принципової.

Відповідність справа наліво є однозначною, тобто кожен елемент множини справа (контактна площадка) має тільки один зв'язок з елементом множини зліва (ланцюгом).

Іншими словами, одна контактна площадка не може бути з'єднана відразу з двома ланцюгами - це буде означати коротке замикання ланцюгів.

При обриві ланцюга в розглянутому графі буде спостерігатися ізольована від інших вершина, що відповідає певному контактному майданчику.

Заплутування ланцюгів буде еквівалентно новому графу, відмінному від того, який був заданий схемою електричної принципової.

Таким чином, будь-який структурний дефект призведе до порушень інцидентності або навіть зв'язності вихідного графа.

Базова модель у вигляді графа є наочною і представляється зручною для аналізу, але це справедливо лише для списків ланцюгів дуже малих розмірів. У реальності, навіть прості пристрої, що представляються невеликими схемами електричними принциповими, мають сотні контактних майданчиків і десятки ланцюгів. При цьому очевидно, що побудова моделі у вигляді графа і подальший аналіз стають дуже утрудненими.

За списком з'єднань можна також побудувати базову модель у вигляді матриці суміжності, елементи якої будуть визначатися в такий спосіб

$$a_{ij} = \begin{cases} 1, \text{ якщо контактні майданчики } p_i \text{ і } p_j \text{ з'єднані ланцюгом} \\ 0, \text{ якщо з'єднання між майданчиками відсутнє} \end{cases}$$

Матриця суміжностей є симетричною з нулями на головній діагоналі. Елементи матриці згруповані за належністю до ланцюгів схеми. Більшість ланцюгів в схемі з'єднує між собою дві або три контактні площадки. На цьому фоні часто виділяються два ланцюги, що охоплюють відносно велику кількість контактів, - це ланцюги напруги живлення і «землі».

Побудована на основі інформації зі списку з'єднань базова модель електронного модуля в графовому або матричному представленні дає потенційну можливість локалізації структурних дефектів модуля з точністю до контактної площадки елемента. Класичні базові моделі, засновані на блочно-функціональних схемах або причинно-наслідкові зв'язки, не дозволяють забезпечити такий ступінь деталізації.

Діагностична модель розробляється на основі базової та дозволяє за рахунок інформації зі списку з'єднань (що лежить в основі базової моделі) досягти високої точності локалізації.

Ключовим моментом для побудови діагностичної моделі є наявність у схемі електронного модуля хоча б однієї мікросхеми, що підтримує технологію граничного сканування за стандартом IEEE 1149.1 [60].

Технологія граничного сканування також широко відома по найменуванню об'єднаної групи розробки методів тестування, що створила її.

В мікросхему з підтримкою стандарту IEEE 1149.1 вбудовується додаткова тестова логіка, основою якої є послідовний зсувний регістр (boundary scan register), що розташовується між функціональним ядром і контактами введення-виведення мікросхеми (на кордоні).

Тестова логіка обов'язково включає TAP-контроллер (Test Access Port - порт тестового доступу), регістр команд (Instruction Register) і набір регістрів даних (Data Registers). Обов'язковими за стандартом регістрами даних є регістр граничного сканування (Boundary-Scan Register) і регістр обходу (Bypass Register). Також в тестову логіку можуть бути включені додаткові

регістри даних: регістр ідентифікації (Identification Register), допоміжні тестові регістри (при наявності вбудованих в системну логіку тестових засобів, наприклад, для забезпечення покрокового налагодження).

Особливості технології граничного сканування на рисинку 2.11.

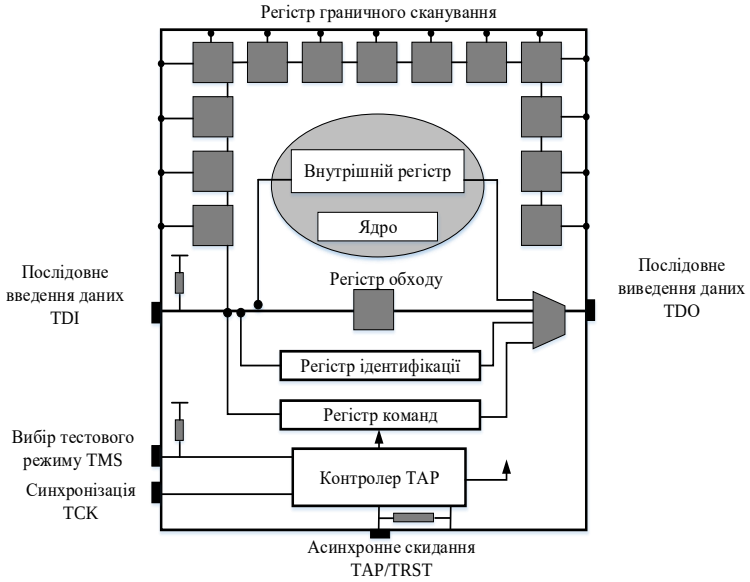


Рис. 2.11 Структура підтримки технології граничного сканування

TAP-контролер представляє собою кінцевий автомат з 16 станами: Test-Logic-Reset, Run-Test-Idle і дві ідентичних групи станів (Select, Capture, Shift, Exit1, Pause, Exit2, Update), що відрізняються суфіксом: -IR (для операцій з регістром команд) або -DR (для операцій з регістрами даних). Схема кінцевого автомата показана на рисинку 2.12.

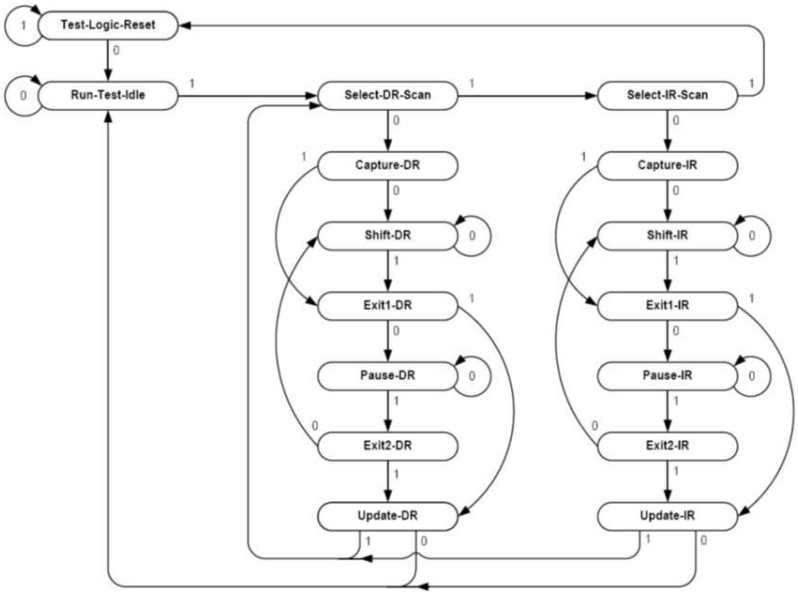


Рис. 2.12. Кінцевий автомат TAP-контролера

Зміна станів автомата здійснюється в залежності від значення сигналу TMS по кожному фронту сигналу TCK. Стандарт IEEE 1149.1 визначає наступні сигнали тестового доступу:

– TDI (Test Data Input),
– TDO (Test Data Output),
– TMS (Test Mode Select input),
– TCK (Test ClocK input),
– TRST (Test ReSeT input).

Через контакти TDI, TDO здійснюється послідовне введення-виведення даних в перераховані вище тестові регістри. Через контакт TMS здійснюється управління TAP-контролером. За сигналом TCK здійснюється зміна станів TAP-контролера і зрушення даних від TDI до TDO через будь-який тестовий регістр. Необов'язковий сигнал / TRST надає можливість асинхронного скидання тестової логіки.

Технологія IEEE 1149.1 дозволяє виставляти на контакти мікросхеми, завантажені через регістр граничного сканування тестові вектори і через цей же регістр зчитувати з контактів реакцію на впливи. Функціональне ядро мікросхеми при цьому залишається (повинно залишатися) незадіяним.

Пропонована діагностична модель полягає в порівнянні (знаходженні деякого показника близькості) інформації, отриманої в результаті діагностичного експерименту з використанням регістра граничного сканування, з базовою моделлю, заздалегідь створеної за списком ланцюгів даного модуля.

З точки зору підтвердження справності шляхом перевірки відсутності в електронному модулі структурних дефектів типу замикань, обривів, змішування ланцюгів доцільно в якості базової моделі розглянути зв'язки між двома множинами: множина ланцюгів N (nets) і множина контактних майданчиків P (pads). У множині контактних майданчиків можна виділити дві (що представляють інтерес) підмножини: керованих D (drived) і спостережуваних S (sensed) за допомогою регістра граничного сканування. За

допомогою майданчиків підмножини D можна виставляти тестові впливи, за допомогою майданчиків підмножини S можна зчитувати реакцію. Співвідношення множин ілюструється у відповідності з рисунок 2.13

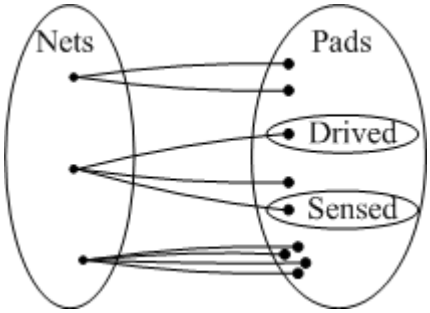


Рис. 2.13 Множина тестованих ланцюгів і множини керованих і спостережуваних контактних майданчиків

Як випливає з рис. 2.13, далеко не всі ланцюги можуть бути протестовані. В цьому відношенні, чим більше мікросхем з підтримкою технології граничного сканування містить схема, тим краще.

У справної схеми кола не перетинаються, кожному контактному майданчику ставиться у відповідність один ланцюг. Кожному ланцюгу може відповідати кілька контактних майданчиків. Еталонна матриця A (без несправностей) складається за схемою. Обмеженням даної моделі є те, що елементами еталонної матриці можуть бути тільки майданчики, які керуються і спостерігаються за допомогою технології граничного сканування. Наприклад, на рис. 28 показаний граф, що відповідає справній схемі, в якій майданчики p_1 і p_2, а також p_3 і p_4 з'єднані двома різними ланцюгами.

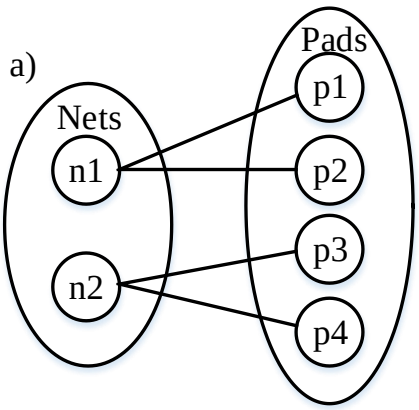


Рис. 28. Граф справної схеми

При цьому матриця суміжності графа буде відображатися у вигляді:

$$A = \begin{Vmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{Vmatrix}$$

Інформація, отримана в результаті діагностичного експерименту, може бути представлена у вигляді матриці B.

Наприклад, на рисинку 2.14 показана діагностична модель у вигляді графа, коли в схемі є обрив ланцюгів.

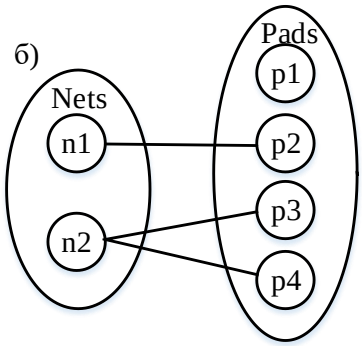


Рис. 2.14 Діагностична модель у вигляді графа для обриву ланцюга

Матриця суміжності для цього випадку буде відображатися у вигляді:

$$B_o = \begin{Vmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{Vmatrix}$$

Таким чином, при обриві ланцюга в матриці суміжності будуть 0 замість 1.

На рисунку 2.15 показаний граф для змішування ланцюгів.

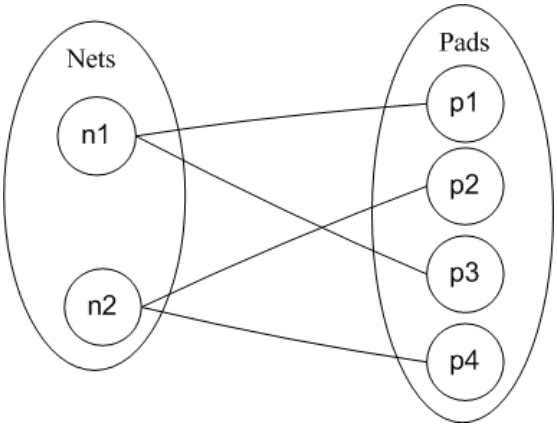


Рис. 2.15 Діагностична модель змішування ланцюгів

Матриця суміжності для змішування ланцюгів буде представлена у вигляді:

$$B_c = \begin{Vmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{Vmatrix}$$

На рисунку 2.16 проілюстрована діагностична модель для короткого замикання ланцюгів. При цьому матриця суміжності графа буде відображатися у вигляді:

$$B_s = \begin{Vmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{Vmatrix}$$

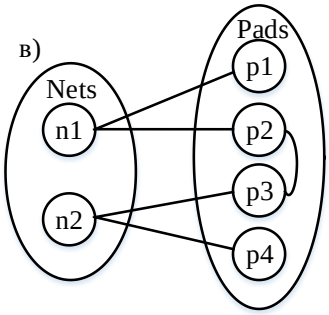


Рис. 2.16 Діагностична модель у вигляді графа для короткого замикання ланцюгів

При замиканні або переплутуванні ланцюгів в матриці суміжності будуть або зайві 1, або 1 не на своєму місці.

Пошук структурного дефекту зводиться до обчислення виразу $C = A \oplus B$, де $\oplus$ - операція «що виключає АБО».

При $C = 0$ виходить збіг інформації, отриманої в результаті діагностичного експерименту, з еталонною. При $C \neq 0$ в схемі міститься структурний дефект, стан якого відображається ненульовими елементами матриці C. Наприклад, для обриву ланцюга буде справедливо:

$$C = A \oplus B = \begin{Vmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{Vmatrix}$$

Представлені діагностичні моделі можуть застосовуватися для електронних модулів як МБВС-М, так і МБВС-П.

Для аналізу на етапі експлуатації в якості вихідної обрана структура бортової обчислювальної системи, спрощено зображеної на рисинку 2.17

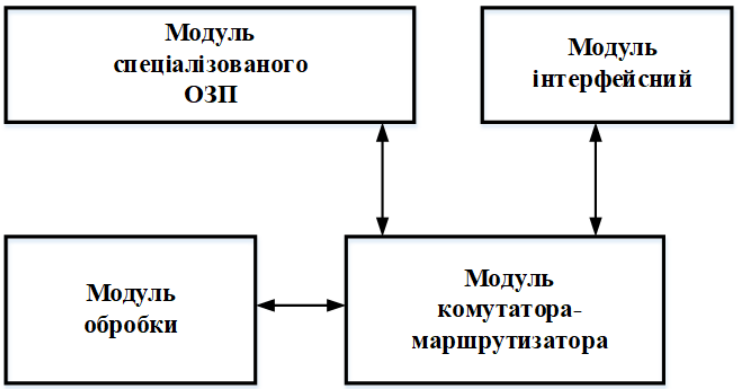


Рис. 2.17. Вихідна структура бортової обчислювальної системи

Система будується на основі високошвидкісної комутаційної мережі SpaceWire. Детальна структурна схема і елементна база такої системи розглянуті в розділі 2.1.

Модуль комутатора-маршрутизатора має вільні порти, які можуть бути використані для масштабування або резервування. При нарощуванні модулів обробки структуру можна віднести до класу багатомашинних, коли кожен процесор, що обробляється, має власну пам'ять (як постійну, так і оперативну) [75]. Багатомашинні структури дозволяють забезпечити одночасне підвищення продуктивності та відмовостійкості. При цьому, зрозуміло, алгоритми управління повинні допускати можливість розпаралелювання.

В даному прикладі надмірність вводиться тільки для модулів обробки в припущенні, що вони з більшою ймовірністю схильні до відмов в порівнянні з іншими модулями. Припущення базується на традиційному уявленні, що найменш надійними елементами обчислювальної техніки є мікросхеми

оперативних запам'ятовуючих пристроїв [105]. Мікросхеми, що застосовуються в модулі спеціалізованого ОЗУ для зберігання проміжних результатів, мають підвищений рівень безперебійної роботи, проте їх інформаційна ємність недостатня для застосування в модулях обробки. Модулі інтерфейсний і комутатора-маршрутизатора побудовані на основі спеціалізованих НВІС, які (в порівнянні з ОЗУ) також менш схильні до збоїв або відмов. Тому надмірність далі розглядається тільки відносно модулів обробки.

Також слід зазначити, що в даній роботі розглядається мінімальна надмірність, а саме та, яка дозволяє зберегти працездатність при відмові тільки одного модуля. Відповідно, в структурній схемі додається тільки один резервний модуль обробки, як показано на рисунку 2.18

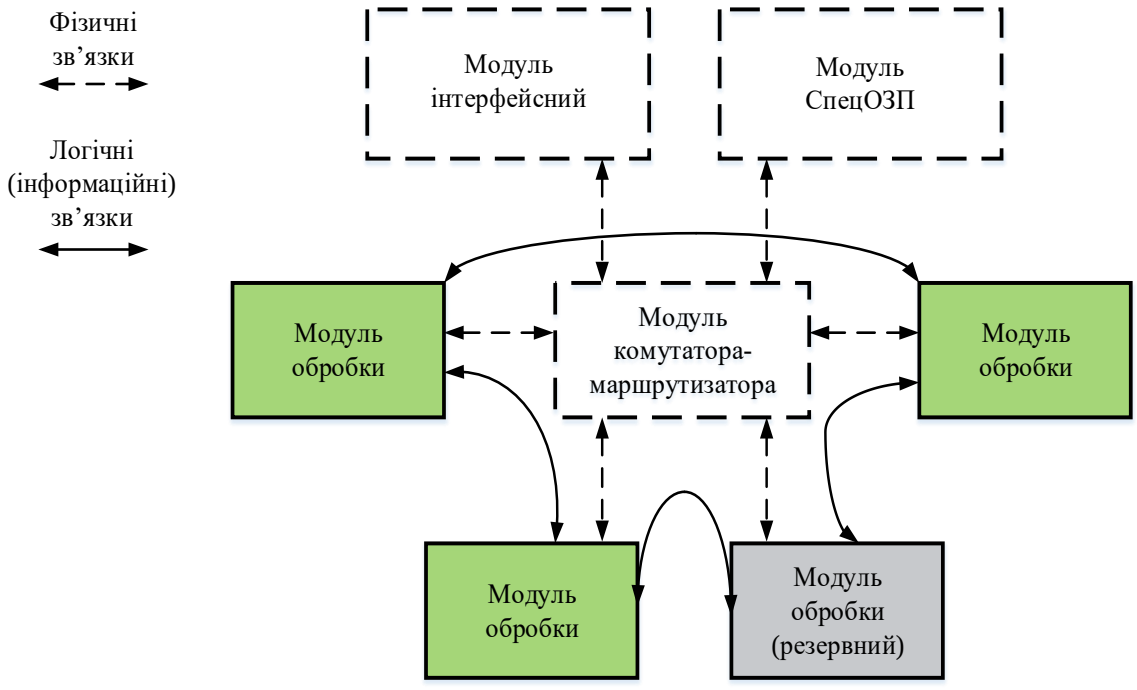


Рис. 2.18 Багатомашинна структура з резервним модулем

Відмовостійкість в більш загальному випадку, тобто коли відмовляють k модулів, розглядається в [106].

Згідно рисунку 2.18 апаратна частина бортової обчислювальної системи має повнозв'язну структуру. Організація логічних зв'язків, тобто розподіл потоків даних і керуючої інформації при паралельній обробці, менш

очевидна і може бути реалізована різними способами при розробці програмного забезпечення системи. Пошук спільних рекомендацій з проектування системи на логічному рівні, що дозволяють забезпечити розглянутий рівень відмовостійкості, і становить предмет подальшого розгляду.

Структура моделюється у вигляді графа, вершини якого відповідають модулям обробки, а ребра - логічним (інформаційним) зв'язкам між ними. Для визначеності резервний модуль обробки відповідає вершині $V0$. Застосовувана модель передбачає, що відмовам схильні тільки вершини графа.

Абстрактною групою G називається множина, кінцева або нескінченна, для якої виконуються умови [107]:

1. $g_i * g_j = g_s \in G$, де * асоціативна операція
2. $\exists g_k = e \in G$ такий, що $e * g_k \in G$
3. $\exists g_i \in G, \exists g_i^{-1} * g_i = e$

Група симетрії деякого об'єкта (багатогранника або множини точок з метричного простору) - це група всіх рухів, для яких даний об'єкт є інваріантом, з композицією в якості групової операції.

Далі будуть розглянуті циклічна, діедральна та тетраедральна групи симетрії [108].

Циклічна група симетрії (C_n) являє собою групу циклічних поворотів на кут $\alpha = \frac{360^o}{N}$ навколо осі, що проходить через центр O системи, як показано на рисунку 2.19.

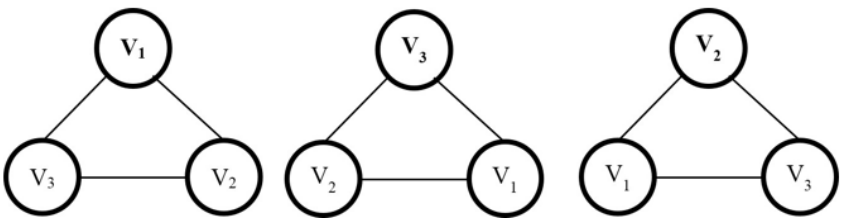


Рис. 2.19 Група автоморфізмів графа G_3 циклічної групи симетрії

Дана група має порядок $|C_n| = N = n$ отже група містить автоморфізмів [108].

Діедральна група симетрії (D_n) складається з власних обертань в просторі на кути $\alpha = \frac{360^o}{2N}$, навколо осі, що проходить через центр O і на 180° навколо $N/2$ осей другого порядку. Порядок групи $|D_n| = N = 2n$ отже, група містить $2n$ автоморфізмів, як показано на рисунку 2.20: тотожний автоморфизм; – 1 поворотів на кути $2\pi k/n$, де = 1 . . . −1; відображень щодо осей, що проходять через центр O і утворюючі між собою кути, кратні $2\pi/n$ [108].

Тетраедральна група симетрії (T) тісно пов'язана з представленням плоского графа у вигляді просторового. Тетраедр має 4 осі симетрії, що проходять через його вершини і центри протилежних граней. Навколо кожної осі крім тотожного можливі ще два обертання. Таким чином, група обертань тетраедра має 12 перестановок, включаючи тотожне [108].

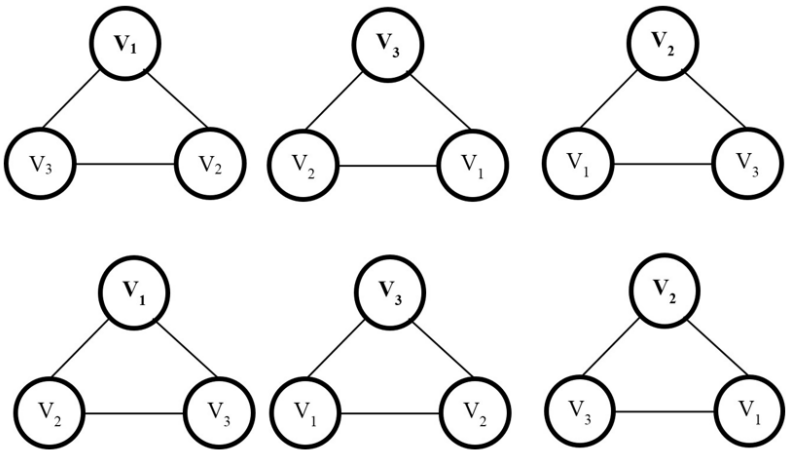


Рис. 2.20. Група автоморфізмів графа G_3 діедральної групи симетрії

Структура МБВС-П моделюється у вигляді графа, вершини якого відповідають модулям обробки, а дуги - логічним (інформаційним) зв'язкам між ними. Для визначеності резервний модуль обробки відповідає вершині …$V0$. Застосовувана модель передбачає, що до відмов схильні тільки вершини графа. Комутаційна мережа до відмов не схильна, тобто при відмові модуля обробки можна здійснити реконфігурацію системи відповідно до рисунку 2.21 за рахунок перебудови роботи модуля комутатора -

маршрутизатора і замінити модуль обробки, що відмовив на резервний.

У найпростішому випадку в системі два модуля, при цьому резервний модуль заміщає той, що відмовив, як показано на рис. 2.21 а.

Такий граф відповідає формату двоканального обчислювача з «холодним» резервом.

При організації логічних зв'язків в структурі з трьома модулями слід врахувати, що в ланцюжках на рис. 2.21 б і рис. 2.21 в резервний модуль може замістити лише один з основних. Більшою відмовостійкістю будуть володіти структур и з організацією логічних зв'язків, показані на рис. 2.21 г і рис. 2.21 д. У цих випадках резервний модуль здатний замінити будь-який з основних.

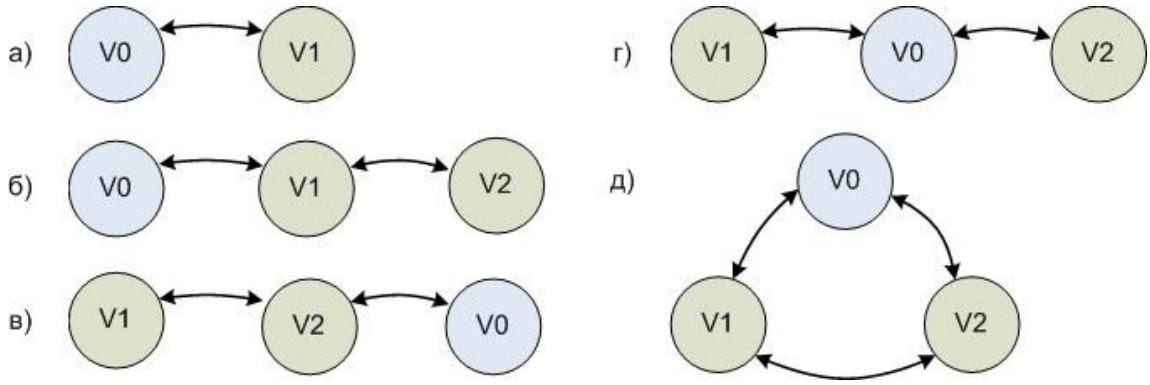


Рис. 2.21: Варіанти надлишкових структур з двома і трьома модулями обробки

При збільшенні розмірності системи відповідно зростає кількість варіантів з'єднань між модулями, при цьому організація логічних зв'язків стає менш очевидною. Як приклад на рисунку 2.20 показані структури з чотирма модулями обробки.

На рис. 2.21а – 2.21г вершина $V0$ є суміжною лише з однією з вершин графа, тобто резервний модуль здатний замінити тільки один з основних. Кращі можливості щодо забезпечення відмовостійкості відповідно до рис. 2.21ж є у повнозв'язного графа. При подальшому зростанні розмірності системи організація такої структури буде ставати складніше і дорожче. Аналізуючи два приклади, що залишились, слід зазначити, що замінити будь-який з основних модулів на резервний, не порушивши відносин

інцидентності, можна лише для випадку на рис. 2.21д.

У загальному випадку показано, що відмовостійкість з мінімально можливою надмірністю можна реалізувати тільки в структурах, що володіють певного роду математичною симетрією. Заміна модуля, що відмовив, на резервний в рамках застосовуваної моделі означає перейменування логічних імен вершин графа таким чином, що надлишкова вершина заміщає ту, що відмовила, без зміни інцидентності між новими логічними іменами[109]. Таке перетворення графа є інваріантним і також називається автоморфизмом. Сукупність автоморфізмів утворює групу симетрії. Знаючи групу симетрії, якою володіє система, можна скласти таблицю групових перетворень. При цьому число перетворень дорівнює порядку групи симетрії. Зазначена таблиця буде однозначно визначати процес реконфігурації після відмови. Для системи, показаної на рис. 2.21, діагностику і реконфігурацію повинен здійснювати модуль комутатора-маршрутизатора, відповідно таблиці реконфігурації повинні зберігатися у нього в пам'яті.

2.5 Висновки до другого розділу

Розроблено метод підтримки проектування інтегрованої модульної авіоніки який дозволяє вдосконалити процес проектування комплексів бортового обладнання нового покоління, а саме підвищенню якості та ефективності процесу проектування, скороченню часу розробки та впровадженню нових технологій у виробництво авіаційної техніки, сприяючи покращенню конкурентоспроможності та інноваційності авіаційної промисловості.

А також розроблена графова модель структури функцій комплексу бортового обладнання інтегрованої модульної авіоніки для реалізації обчислювальної потужності та навантаження на мережу передачі даних є важливим інструментом для аналізу та оптимізації роботи авіонічних систем. Ця модель дозволяє систематизувати та візуалізувати взаємозв'язки між

різними функціями та параметрами системи, що допомагає зрозуміти їх вплив на обчислювальні та комунікаційні процеси.

Результати застосування цієї моделі можуть бути корисними для пла планування та оптимізації роботи бортових систем з урахуванням обмежень обчислювальної потужності та мережі передачі даних.

Виділення та відокремлення процесу проєктування структури функцій в загальному процесі проєктування КБО дозволяє вирішувати це завдання як оптимізаційне, сформувати вимоги щодо застосування методів підвищення надійності та відмовобезпечності виконання функцій на ранніх стадіях проєктування і подальшу розробку КБО здійснювати з урахуванням даних вимог. Виділення і відокремлення процесу проєктування структури функцій не призводить до погіршення адаптивності процесу проєктування щодо федеративних КБО. Функціональні засади розробки та експлуатації програмного забезпечення у сфері авіації мають власну специфіку, яка базується на відповідних стандартах.

Для розробки, налагодження і виготовлення важливо забезпечити пошук дефекту з глибиною пошуку до ланцюга або до контактної площадки. Тому елементами графа є ланцюги і відповідні їм контакти. При цьому побудована діагностичної моделі на основі списків з'єднань і з застосуванням технології граничного сканування. Це дозволяє збільшити глибину пошуку структурних дефектів (таких як обриви, короткі замикання або змішування ланцюгів), що знижує ризики втрат різного виду. Для етапу експлуатації бортових систем важливим є забезпечення відмовостійкості при впливі несприятливих факторів. В цьому випадку діагностична модель не вимагає високої деталізації. Необхідно встановити факт правильного функціонування, а в разі порушення роботи виконати реконфігурацію системи. Тому елементами графа в цьому випадку є модулі обробки. При цьому як витрати на апаратурну надмірність, так і витрати часу на відновлення повинні бути мінімальні. Це можна досягти для систем, структура яких описується певною алгебраїчної моделлю.

РОЗДІЛ 3. РОЗРОБКА МАТЕМАТИЧНОГО Й АЛГОРИТМІЧНОГО ЗАБЕЗПЕЧЕННЯ ІНФОРМАЦІЙНОГО ПРОЦЕСУ АВТОМАТИЗОВАНОГО ПРОЄКТУВАННЯ КОМПЛЕКСУ БОРТОВОГО ОБЛАДНАННЯ ІНТЕГРАЛЬНОЇ МОДУЛЬНОЇ АВІОНІКИ

Розробити модель процесу проектування комплексу бортового обладнання інтегрованої модульної авіоніки для побудови математичних методів підвищення надійності та відмовостійкості на етапі проектування Запропоновано підхід до проектування додатків з використанням модельно-орієнтованого проектування в Simulink та їх перенесення на ARINC 653-сумісну виконавчу платформу на основі XtratuM. Цей підхід базується на правильному відображенні абстракцій Simulink на абстракції та сутності ARINC 653, що дозволяє досягти інтеграції різних компонентів і платформ.

Для вирішення цієї задачі використовуються інструменти Simulink для автоматизації процесу відображення, а також спеціальні інструменти, розроблені для цієї задачі. Використання архітектур Model-Based Design і Integrated Modular Avionics (IMA) з Simulink і XtratuM підтверджує, що XtratuM може бути використана як платформа для додатків авіоніки.

Для досягнення поставленої мети були вирішені такі задачі: досліджено взаємодію основних функціональних компонентів, що автоматизують проектні процедури створення апаратури комплексів бортового обладнання; визначено відмовостійкість комплексу бортового обладнання інтегральної модульної авіоніки для процесів проектування та експлуатації.

3.1. Системно-технологічний процес розробки, виробництва і випробувань КБО на основі галузевої САПР

Сучасні тенденції розвитку приладобудівної галузі в області проектування бортового радіоелектронного обладнання нерозривно пов'язані із застосуванням систем автоматизованого проектування (САПР). Розробники САПР пропонують різні універсальні програмні інструменти,

призначені для автоматизації окремих етапів проектування БРЕО. Відомі, наприклад, пакети прикладних програм:

– тривимірного проектування конструкцій апаратури БРЕО – AutoCAD, Solidworks, UniGraphics;

– інженерних розрахунків і моделювання теплових полів, створюваних апаратурою БРЕО: BETAsoft, Sauna, АСОНІКА-Т;

– моделювання електромагнітних полів, що створюються апаратурою БРЕО –OrCAD Family Release, GENESYS;

– спектрального аналізу радіочастотних сигналів, що генеруються і приймаються апаратурою БРЕО – TESLA;

– моделювання міцності і резонансних характеристик конструкцій авіаційних виробів - Samcef, Mecano, Boss Quattro;

– автоматичного трасування друкованих плат з урахуванням тривимірного компонування елементів монтажу – ACCEL EDA, PCAD;

– конструювання та розводки джгутів в тривимірному просторі корпусів апаратури БРЕО - UG / Wiring;

– моделювання гідродинамічних процесів в системах охолодження авіаційних виробів з урахуванням тепловиділення електрорадіоелементів і фізичних процесів теплопереносу – Fine / Turbo;

– електронного моделювання радіотехнічних сигналів і ланцюгів - Cadence Design, Mentor Graphics, XILINX Foundation, ALTERA, MicroCap;

– проектування програмного забезпечення C / C ++ / C #, ADA і ін.

Дані програмні інструменти складають основу автоматизації проектних технологій і є закінченими програмними рішеннями. Специфічність форматів зберігання і представлення даних результатів проектування в таких пакетах програм істотно впливає на рівень автоматизації процесу проектування БРЕО в цілому. Це пов'язано в першу чергу з тим, що формат представлення даних результатів проектування будь-якої складової частини виробу, отриманого із застосуванням САПР одного виду, виявляється неузгодженим з форматом зберігання і представлення даних в САПР іншого виду, що застосовується на

наступному, більш пізньому, етапі проектування.

У таких випадках виникає потреба введення в технологічні маршрути проектування виробів «ручних» процедур підготовки (переформатування) окремих електронних документів, що істотно знижує ефект застосування САПР і підвищує сумарну трудомісткість розробки конструкторської, технологічної та програмної документації на виріб в цілому.

Для виключення або мінімізації обсягів «ручних» робіт в процесі проектування всі застосовувані при розробці БРЕО кошти САПР повинні інтегруватися в єдину галузеву САПР [66], що володіє сумісними форматами уявлення, зберігання і передачі даних і підтримує всі етапи життєвого циклу БРЕО - від маркетингових досліджень доцільності розробки нових видів виробів до їх подальшої утилізації. Відповідна функціональна схема взаємодії різнорідних засобів САПР в складі єдиної галузевої САПР підтримки життєвого циклу БРЕО приведена на рисунку 3.1.

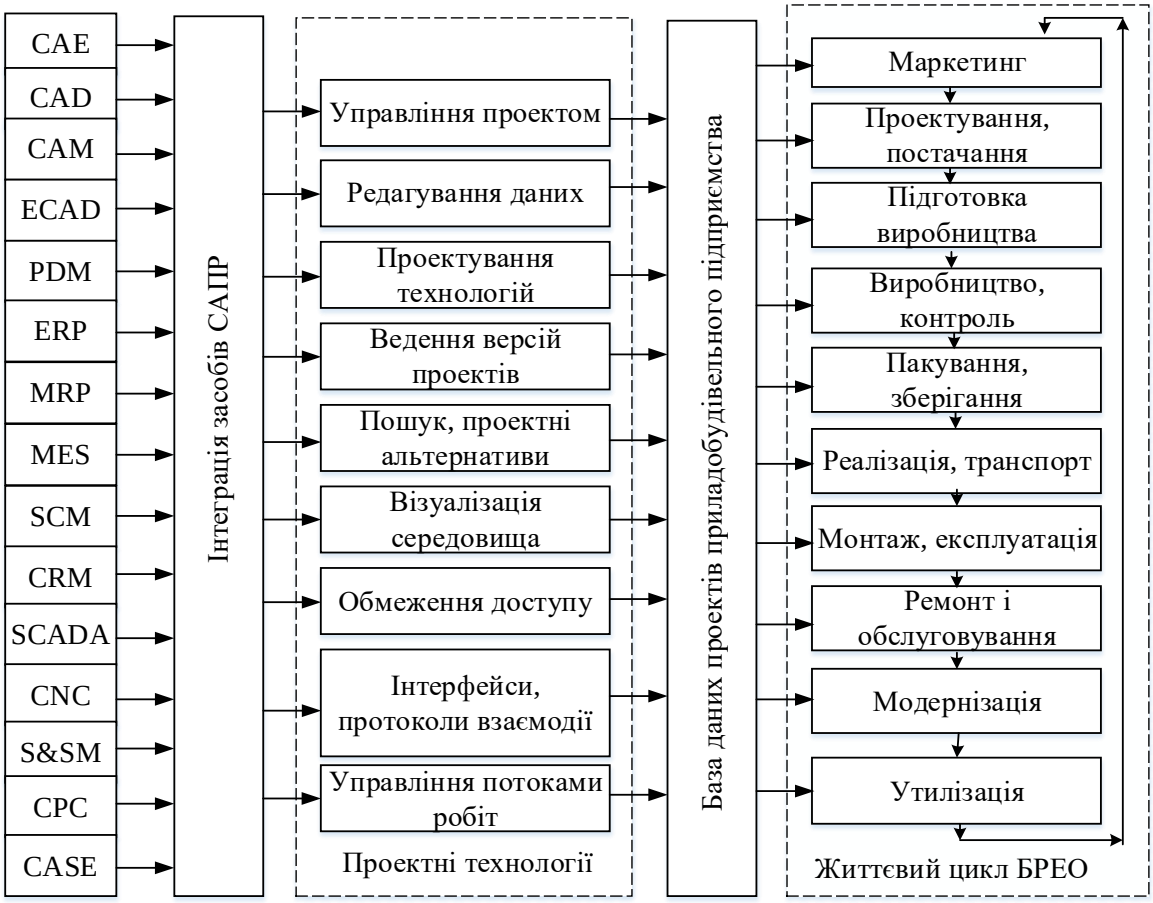


Рис. 3.1. Функціональна схема взаємодії засобів САПР в складі єдиної галузевої САПР підтримки життєвого циклу БРЕО

Інтеграція різнорідних засобів САПР передбачає об'єднання технологічних операцій, що реалізуються кожною з них окремо, в єдину проектну середу «проектування-виробництво-експлуатація». Єдине проектне середовище забезпечить автоматизовану підтримку етапів життєвого циклу БРЕО і дозволить дослідити і розробити «наскрізні» проектні технології створення виробів, що входять в БРЕО, до їх реалізації у вигляді готових фізичних об'єктів; розробити технологічний цикл розробки, виробництва і випробувань апаратури БРЕО;

Технологічний цикл розробки, виробництва і випробувань апаратури БРЕО на основі галузевої САПР авіаційного приладобудування представлений на рисунку 3.2

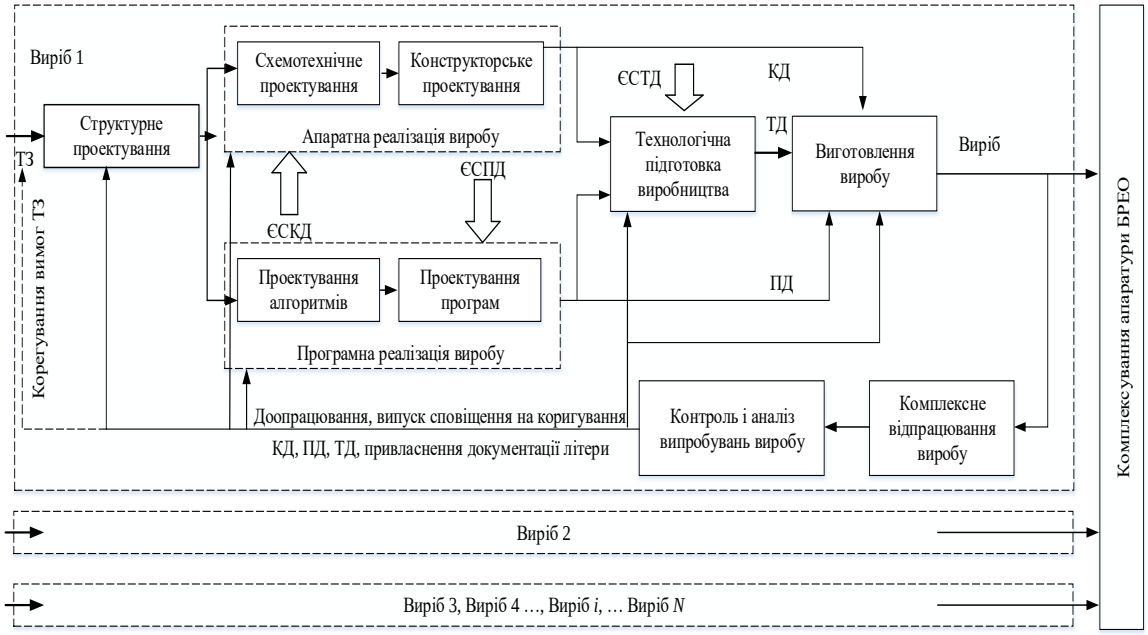


Рис. 3.2 Технологічний цикл розробки, виробництва і випробувань апаратури БРЕО

Де, КД – конструкторська документація; ТД – технологічна документація; ПД – програмна документація; ЄСКД – єдина система конструкторської документації; ЄСТД – єдина система технологічної документації; ЄСПД – єдина система програмної документації; ТЗ - технічне завдання.

Отже, основу технологічного циклу розробки, виробництва і випробувань апаратури БРЕО складають проектні процедури структурного

проектування окремих виробів, реалізації їх апаратних і програмних компонентів, підтримки завдань технологічної підготовки виробництва, виготовлення і випробувань виробу. Суттєвою складовою проектування БРЕО є також етап комплексування апаратури [3].

Основне завдання комплексування апаратури полягає в об'єднанні методами структуризації різнорідних приладів і пристроїв (виріб 1, виріб 2 і т.д.) в єдиний комплекс бортового обладнання. Об'єднання здійснюється на апаратному та програмному рівні через аналіз інформаційних потоків аеродинамічних, пілотажно-навігаційних та ін. параметрів руху повітряного судна і синтез структур БРЕО. Результатом етапу комплексування апаратури є комплект документації на комплекс БРЕО. Відповідно до прийнятої термінології комплекс БРЕО являє собою систему більш високого порядку по відношенню до окремих її компонентів.

Таким чином, завдання автоматизації етапів «проектування-виробництво-експлуатація» БРЕО на основі єдиного проектного середовища галузевої САПР авіаційного приладобудування слід розглядати як завдання створення такої інтегрованої САПР, в складі якої:

– автоматизувалися би всі основні проектні процедури проектування БРЕО;

– взаємодія (обмін даними між додатками) різнорідних засобів САПР здійснювалася б на основі стандартизованих протоколів передачі інформації, наприклад, за стандартами групи STEP (Standard for the Exchange of Product).

Схема взаємодії основних функціональних компонентів, що автоматизують проектні процедури створення БРЕО на основі галузевої САПР, представлена на рис. 3.3.

Автоматизація етапу структурного проектування виробу здійснюється засобами автоматизації процедур вибору оптимальної (по заздалегідь введеному критерію) структури виробу і процедур розподілу функцій між апаратними та програмними компонентами виробу. Методи вибору оптимальної структури виробу в класі бортового приладового обладнання

розглянуті в роботі [4], рішення задачі «про призначення» функціональних завдань на доступні обчислювальні ресурси БРЕО розглянуто в [5]. Автоматизація етапу розробки апаратної реалізації виробу здійснюється послідовно на основі автоматизації процедур схемотехнічного проектування та процедур розробки конструкції виробу. Методи автоматизації процедур схемотехнічного проектування і генерації проектних рішень розглянуті в роботах [6] і [7] відповідно.

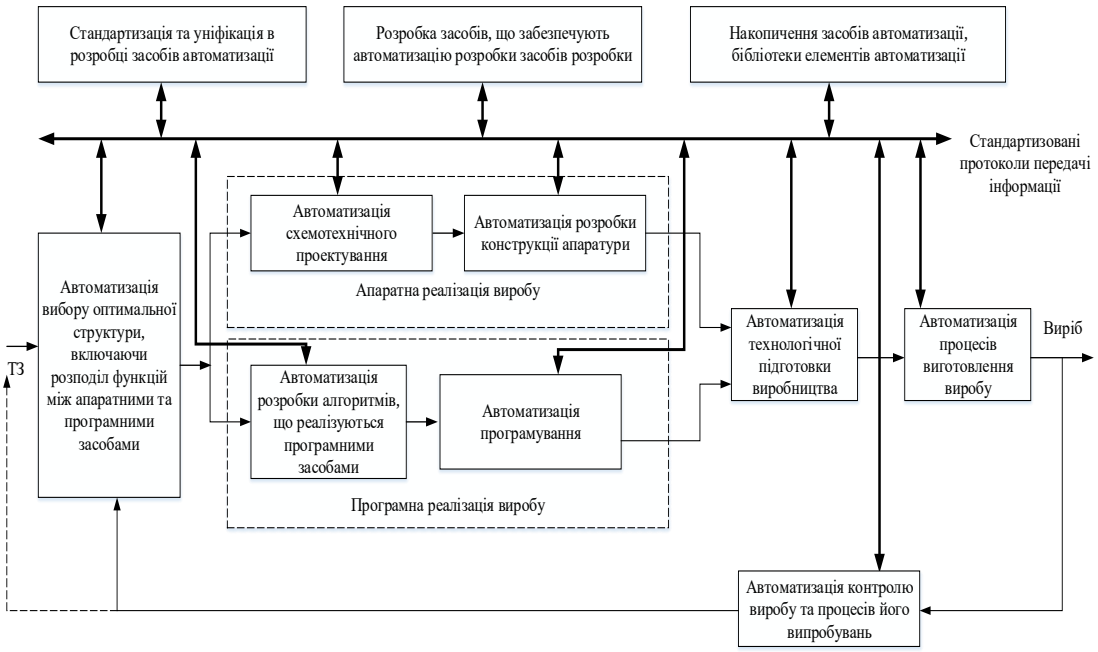


Рис. 3.3 Схема взаємодії основних функціональних компонентів, що автоматизують проектні процедури створення апаратури комплексів БРЕО

Автоматизація етапу розробки програмної реалізації виробу здійснюється послідовно на основі автоматизації процедур проектування алгоритмічного забезпечення і процедур автоматизації програмування. Методи автоматизації процедур проектування алгоритмів і програм розглянуті в роботі [8].

Результатом автоматизації етапів розробки апаратної і програмної реалізації виробу є комплекти програмної та конструкторської документації, що представляють собою основу (вхідні дані) для етапу технологічної підготовки виробництва. Автоматизація технологічної підготовки виробництва реалізується на основі принципів стандартизації і уніфікації технологічних операцій і технологічних процесів із застосуванням

універсального верстатного обладнання. При цьому комплект КД на виріб трансформується в формат 3-Dimension (3D) -моделей.

3D-моделі є основою для виготовлення механічних деталей виробу автоматизованим способом на верстатах з числовим програмним управлінням. Складальне виробництво (поверхневий монтаж електрорадіоелементів, виготовлення багатошарових друкованих плат) також реалізується автоматизованим способом на основі маршрутних карт, що перетворюють конструкторські документи в керуючі програми для теплових печей і автоматизованих установників компонентів.

Автоматизація процедур контролю і випробувань виробу здійснюється обчислювальними засобами діагностики, розробляються автоматизованим способом. До складу засобів діагностики входять програмні компоненти діагностичних тестів справності апаратури, що входять до складу комплекту програмного забезпечення (ПЗ) виробу, апаратні компоненти автоматизованих робочих місць (АРМ) з перевірки, налаштування і регулювання апаратури та випробувальне обладнання підприємства.

Таким чином, автоматизація основних процедур проектування, виготовлення і випробувань апаратури може бути здійснена на основі інтеграції спеціалізованих компонентів САПР, взаємодіючих між собою за допомогою стандартизованих протоколів передачі інформації.

Очевидною перевагою створення галузевої САПР є можливість її навчання в процесі експлуатації. В результаті одноразового застосування формуються галузеві бібліотеки елементів автоматизації, придатні для повторного використання на підприємствах приладобудівного профілю, уточнюються принципи стандартизації і уніфікації в розробці, виробництві і експлуатації авіаційного приладового обладнання, розробляються засоби, що забезпечують автоматизацію розробки самих засобів автоматизації проектування.Функціональна схема ПЗ типової САПР АРМ розробника БРЕО з різними формами взаємодії «САПР$_i$ - файл результату проектування» приведена на рисунок 3.4.

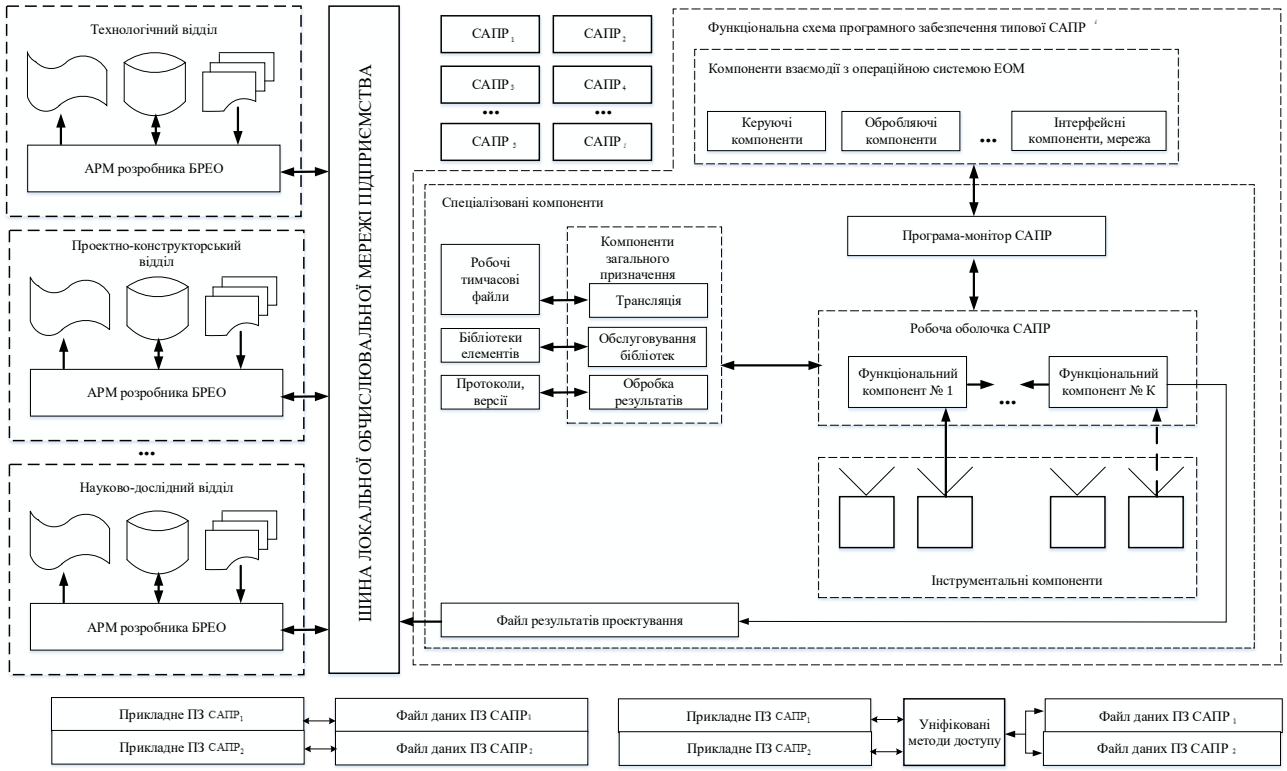


Рис. 3.4. Функціональна схема ПЗ типової САПР автоматизованого робочого місця розробника БРЕО з різними формами взаємодії «САПР i – файл результату проектування».

До складу ПЗ галузевої САПР входять компоненти взаємодії ПЗ САПР з операційною системою інструментальної ЕОМ, на основі якої організовано АРМ розробника апаратури БРЕО, і спеціалізовані компоненти. Спеціалізовані компоненти представлені сукупністю:

– інструментальної складової, що здійснює математичну підтримку вирішення основних проектних завдань і реалізації основних проектних операцій;

– робочою складовою, що здійснює діалогову взаємодію розробника апаратури і робочого середовища проектування;

– компонентів загального призначення, що реалізують невидимі для користувача операції;

– програми-монітора, яка взаємодіє з компонентами операційної системи інструментальної ЕОМ для функціонування універсальних процедур і функцій ПЗ САПР.

Файл результатів проектування галузевої САПР формується в робочій

оболонці САПР і доступний для передачі по локальній мережі підприємства всім спеціалізованим САПР$_i$. Як випливає з рис. 3.5, можливі два різних режими роботи ПЗ САПР з файлами результатів проектування: безпосередній, коли кожне прикладне ПЗ САПР$_i$ може обробляти тільки файли, розроблені в даному середовищі проектування, і універсальний, коли файли результатів проектування, підготовлені в різних САПР, доступні для обробки прикладного ПЗ будь-якої САПР на основі уніфікованих методів доступу. Для побудови галузевої САПР БРЕО, очевидно, придатний другий (універсальний) варіант.

Схема взаємодії різнорідних САПР$_i$ в складі єдиної галузевої САПР авіаційного приладобудування приведена на рисунку 3.6.

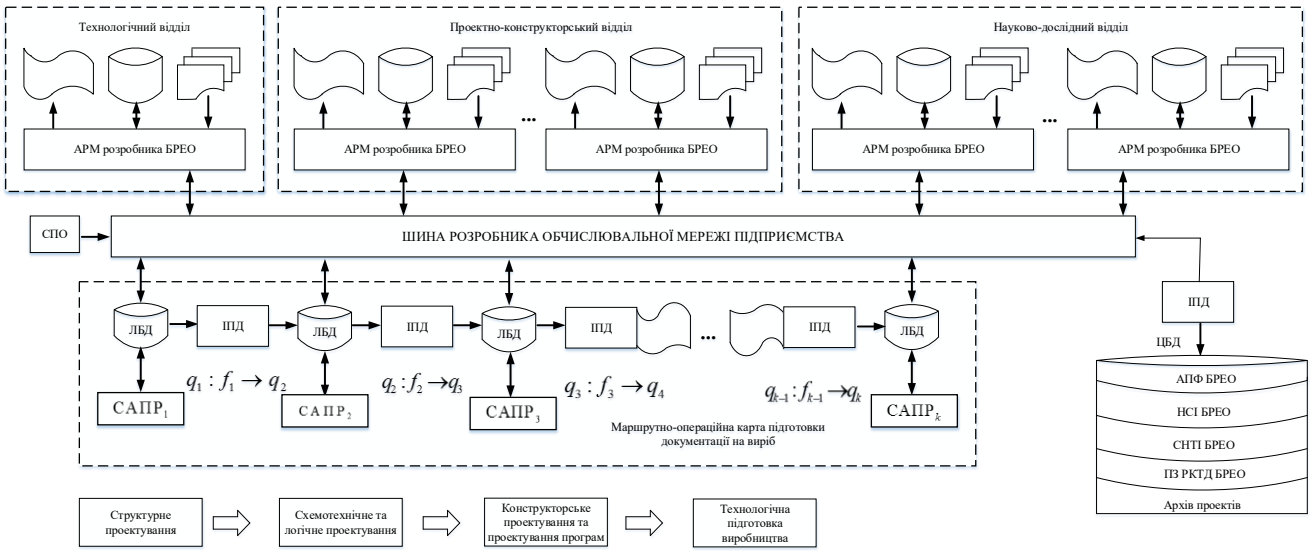


Рис. 3.6. Схема організації процесу наскрізного проектування виробів БРЕО на базі локальної обчислювальної мережі підприємства і центрального банку даних проектів підприємства

Взаємодія різнорідних САПР$_i$ в складі єдиної галузевої САПР БРЕО здійснюється програмним способом на основі інтерфейсів перетворення даних (ІПД). Функція інтерфейсу полягає в нестандартизованих перетвореннях результатів проектування САПР$_i$, що зберігаються в локальному банку даних (ЛБД) цієї САПР, у вид, придатний для введення цих даних в САПР$_{i+1}$ і для подальшої обробки. Функцію перетворення (технологічний перехід) даних q_i в q_{i+1} здійснює оператор f_i (оператор

перетворення форматів). Фізичний обмін даними між різнорідними САПР здійснюється через технічні засоби шини локальної обчислювальної мережі підприємства і спеціалізоване мережеве програмне забезпечення (МПЗ).

Отже, всі технічні відділи, які беруть участь в розробці конструкторської, технологічної та програмної документації, мають доступ до локальної обчислювальної мережі підприємства. Доступ здійснюється через підключені уніфіковані АРМ розробників БРЕО. Результати проектування бортового авіаційного обладнання з використанням галузевої САПР зберігаються в центральному банку даних (ЦБД) підприємства, доступ до якого також здійснюється з АРМ розробників через ПС. В ЦБД зберігаються алгоритми і програми функціонування (АПФ) БРЕО, що розробляються підприємством; документація на вироби; нормативно-довідкова інформація (НДІ) - опис виробів, опис матеріалів, опис технологічних процесів; довідкова науково-технічна інформація (ДНТІ) - опис авторських винаходів, патентів, результатів виконання науково-дослідних і дослідно-конструкторських робіт (НДДКР); програмне забезпечення розробки конструкторської та технологічної документації (ПЗ РКТД).

Дослідження проблеми побудови галузевих САПР авіаційного приладобудування проводяться сьогодні як у нашій країні, так і за кордоном. Зокрема, відомі роботи [9] в області розробки технологій наскрізного проектування авіаційного обладнання в США, які отримали назву ATIP (Avionics Technology Integrated and Prototyping). Технологія ATIP була вперше використана в якості елемента САПР дослідних стадій проектування компаніями Boeing і Lockheed Mar- tin в процесі виконання НДДКР AVSEP (Avionics Virtual Systems Engineering and Prototyping) при розробці архітектури БРЕО багатоцільового ударного винищувача-бомбардувальника F35 за програмою Joint Strike Fighter [10] . Проведення НДДКР AVSEP дозволило американським розробникам БРЕО визначити основні процедури ітераційного процесу проектування, за допомогою якого проектні організації повинні здійснювати перехід від затвердженої концепції побудови БРЕО до

конкретного складу блоків (модулів), що є матеріалізацією технічного вигляду комплексу.

У нашій країні, в сучасних ринкових умовах проблема автоматизації рішення проектних завдань має ключове значення для успішного функціонування підприємства приладобудівного профілю. Створення галузевої САПР авіаційного приладобудування має стратегічну спрямованість і сприяє технічному і програмному переоснащення проектного та виробничого обладнання промислових підприємств. Застосування галузевої САПР авіаційного приладобудування жорстко не пов'язано зі структурою підприємства і можливо практично на будь-якому підприємстві авіаційної галузі.

Розглянуті елементи вітчизняної галузевої САПР були апробовані на практиці. Експериментально отримані оцінки тривалості проведення НДДКР з використанням елементів галузевої САПР і без них представлені на рис. 3.6. Не важко бачити, що тривалість етапів проектування комплексу БРЕО з використанням засобів САПР становить 7 років (крива 1), без використання засобів САПР - 9 років (крива 2), із застосуванням САПР і принципів наступності, уніфікації та стандартизації розробок - 4,5 року (крива 3). Таким чином, ефект від автоматизації проектних процедур виражається в скороченні тривалості етапів проектування БРЕО на 2-5 років.

3.2. Відмовостійкість комплексу бортового обладнання інтегральної модульної авіоніки в процесі проектування та експлуатації.

Відмовостійкість комплексу бортового обладнання інтегральної модульної авіоніки в процесі проектування та експлуатації є критичною для безпеки польотів. Для забезпечення відмовостійкості важливо використовувати надійні компоненти, захисні технології, системи дублювання та відновлення, а також виконувати систематичне технічне обслуговування та діагностику. Крім того, важливо проводити аналіз вимог до безпеки, оцінювати потенційні ризики та враховувати їх у процесі

проектування та експлуатації.

Як обговорювалося в другому розділі, збій системи — це втрата системних служб або очікуваної функціональності. За відсутності відмовостійкості система може вийти з ладу лише після однієї критичної помилки. Така система, яка фактично є безвідмовною працездатністю, була б допустимою для некритичних функцій, крім того, характеризує цей тип системи, оскільки безперервне обслуговування залежить від спонтанної ремісії активної несправності або несправності, наслідки якої не є настільки серйозними, щоб призвести до відмови системи.

Якщо ймовірність безперервного обслуговування має бути високою, можна включити резервування, щоб забезпечити працездатність системи за наявності будь-яких постійних збоїв. Такі відмовостійкі системи містять додатковий стан несправності, а саме стан відновлення, як показано на рисунку 3.7.

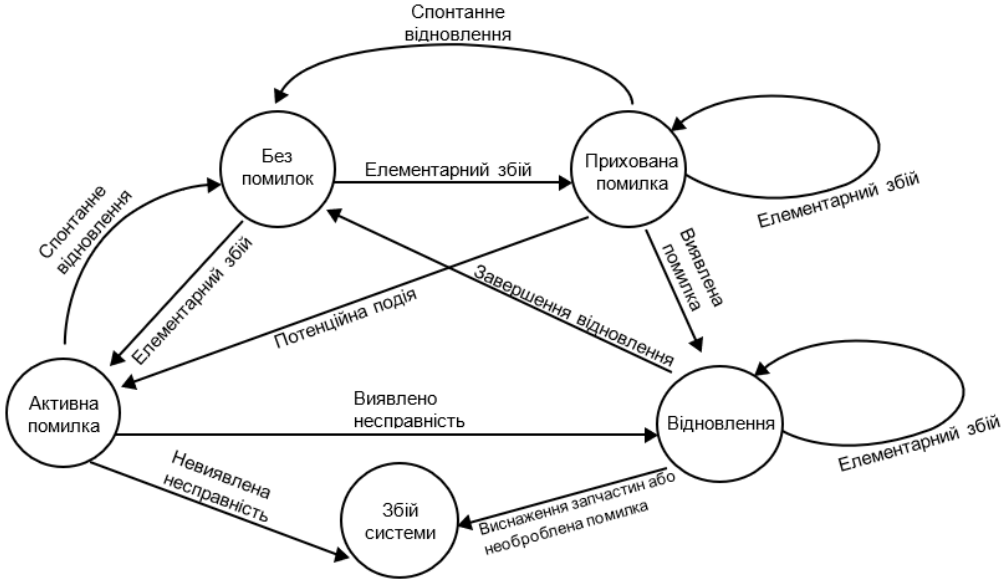


Рис. 3.7. Схема стану обладнання з коригуючою дією.

Збій системи виникає лише після вичерпання запасних частин або невирішеної серйозної несправності. Вищезазначений рівень резервування може зробити вкрай малоймовірним, що запасні частини будуть вичерпані лише в результаті несправності обладнання. Необроблена помилка може виникнути лише як наслідок помилки проектування, як-от вчинення загальної помилки, коли наявність помилки навіть не буде виявлено.

У цьому розділі розглядається перспектива системного рівня та досліджуються відмовостійкі системні архітектури та їх приклади. Тим не менш, ці приклади втілюють і висвітлюють загальні принципи відмовостійкості системного рівня. Особлива увага приділяється системам керування польотом, оскільки вони мотивували та запровадили багато технологій відмовостійкості у застосуванні до цифрових систем польоту. У минулому такі системи були функціонально виділеними, таким чином забезпечуючи значний захист від несправностей через сторонні причини. Проте зі збільшенням поширення інтегрованої авіоніки ступінь розподілу функцій безповоротно зменшується. Насправді це не зовсім доцільно, більше функцій авіоніки, ніж будь-коли, є критично важливими, і ряд переваг отримується від інтегрованої обробки. Крім того, розробник системи може використовувати прерогативи, які забезпечують захист від сторонніх помилок.

Відмовостійкість зазвичай базується на певній формі резервування для підвищення надійності системи за допомогою виклику альтернативних ресурсів. Надмірність може полягати в апаратному забезпеченні, програмному забезпеченні, часі або їх комбінаціях. Існує три основних типи резервування апаратного та програмного забезпечення: статичне, динамічне та гібридне.

Статичне резервування маскує помилки, беручи більшість результатів від повторюваних завдань. Динамічне резервування передбачає двоетапну процедуру для виявлення та відновлення збоїв. Гібридна надлишковість — це комбінація статичної та динамічної надлишковості.

Загалом, велика частина цієї надмірності міститься в додаткових апаратних компонентах. Додавання компонентів скорочує середній час між діями з технічного обслуговування, оскільки є більше електроніки, яка може, і в якийсь момент, відмовить. Оскільки у відмовостійких системах можуть виникати численні різні несправності, існує багато додаткових режимів відмов, які необхідно оцінити при встановленні льотної придатності всієї

системи. Вага, енергоспоживання, охолодження тощо є іншими покараннями за надмірність компонентів. Інші форми резервування також викликають накладні витрати на керування системою, наприклад, обчислювальну потужність для виконання програмно реалізованих завдань відмовостійкості.

Отже, проектування, реалізація відмовостійкості передбачає компроміси та необхідність оптимізації дизайну. Зрештою, слід шукати збалансовану, мінімальну та придатну для валідації конструкцію, яка б наочно забезпечувала гарантії та межі стійкості до несправностей, що відповідають даному застосуванню.

Існує широкий діапазон варіантів реалізації резервування для механізації бажаних рівнів і типів відмовостійкості. На рисунку 3.8 представлено класифікацію варіантів резервування, які можуть бути використані у відповідних комбінаціях, щоб отримати всеохоплюючу відмовостійку архітектуру.

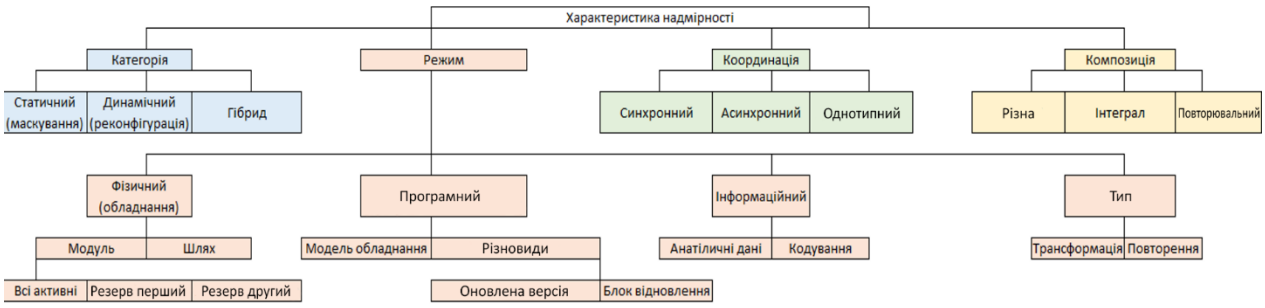


Рис. 3.8.Структура класифікації резервування.

Ця таксономія вказує на широкий спектр можливостей резервування, які можуть бути використані при проектуванні системи. Хоча більшість із цих варіантів наведено в якості прикладів або описано в наступних параграфах, можна коротко зазначити, що резервування характеризується його категорією, режимом, координацією та аспектами складу. Архітектурні зобов’язання мають бути зроблені в кожному аспекті. Таким чином, класична система може, наприклад, використовувати маскування несправностей за допомогою дубльованих апаратних модулів, які працюють синхронно. На нижчому рівні пов’язані шини даних можуть використовувати надлишкове кодування для виявлення та виправлення помилок. Для захисту від загальної

помилки проектування можна додати можливість резервного копіювання за допомогою іншої схеми резервування.

Зокрема, розглянуті елементи маскування проти реконфігурації, активних і резервних запасних частин, а також реплікованого та різнорідного резервування.

Найбільш економічно ефективна фаза загального процесу проектування та розробки для зменшення ймовірності CMFs є найранішою частиною програми.

У таблиці 3.1 представлені методи уникнення помилок та інструменти, які використовуються. Методи та інструменти усунення несправностей загального режиму включають перевірку дизайну, моделювання, тестування, впровадження несправностей і програму контролю якості.

Таблиця 3.1

Методи та інструменти уникнення несправностей

Техніка
Використання зрілих і офіційно перевірених компонентів
Відповідність стандартам
Формальні методи
Автоматизація проектування
Інтегровані формальні методи та методологія проектування VHDL
Спрощення абстракцій
Уникнення збоїв загального режиму продуктивності
Практика розробки програмного та апаратного забезпечення
Різноманітність дизайну

Відмовостійкість загального режиму вимагає виявлення та відновлення помилок. Необхідно підтвердити інформацію про помилку через резервні канали, щоб визначити, який механізм відновлення (тобто фізичне відновлення або відновлення збоїв у загальному режимі) використовувати. Відновлення з CMF у реальному часі вимагає відновлення системи до попередньо відомої правильної точки, з якої обчислювальна діяльність може

відновитися.

Як показано на рис. 3.8, існує три категорії відмовостійких архітектур: маскування, реконфігурація та гібрид.

Підхід маскування є класичним за концепцією потрійного модульного резервування (TMR) фон Неймана, яка була узагальнена для довільних рівнів резервування. Концепція TMR зосереджується на виборці, який у межах обмеження виснаження запасів запобігає проходженню сигналу з помилкою по шляху сигналу. Цей підхід є пасивним, оскільки не потрібно змінювати конфігурацію, щоб запобігти поширенню помилкового стану або ізолювати несправність.

Модульні системи авіоніки, що складаються з кількох ідентичних модулів і вибірковий елемент, вимагають компромісу між надійністю та безпекою. «Модуль» не може бути апаратним модулем; модуль представляє сутність, здатну виробляти результат. Коли безпека розглядається разом з надійністю, конструкція модуля впливає як на безпеку, так і на надійність. Зазвичай очікується, що надійність і безпека підвищаться з додаванням резервування. Якщо модуль має вбудовану функцію виявлення помилок, можна підвищити як надійність, так і безпеку за допомогою додавання одного модуля, який надає дані виборцеві. Якщо на рівні модуля немає можливості виявлення помилок, для підвищення надійності та безпеки потрібні принаймні два додаткові модулі. Арбітражна стратегія контролю помилок — це функція, яку реалізує виборець, щоб вирішити, що є правильним виходом, і коли помилки у виходах модуля є надмірними, тому правильний вихід не може бути визначений, виборець може подати небезпечний сигнал. Надійність і безпека n-module safe modular redundant (nSMR) залежать від надійності окремого модуля та від конкретної стратегії арбітражу, що використовується. Жодна стратегія арбітражу не є оптимальною для підвищення як надійності, так і безпеки. Надійність визначається як ймовірність того, що вихідні дані вибірковий елемент правильні, і виборець не стверджує про небезпечний сигнал. Безпека

надійність плюс ймовірність того, що виборець стверджує про небезпечний сигнал. Оскільки надійність і безпека системи взаємопов'язані, підвищення надійності системи може призвести до зниження безпеки системи, і навпаки.

Вибіркові елементи, які використовують порівняння біт за бітами, використовуються, коли помилки складаються з довільної поведінки з боку несправних компонентів, навіть до екстремальних проявів, здавалося б, розумної зловмисної поведінки. Такі розломи отримали назву візантійських. Вимоги, що висуваються до архітектури, стійкої до візантійських помилок (іменованої стійкою до візантійської архітектури [BR]), включають нижню межу кількості областей локалізації помилок, їх підключення, синхронності та використання певних простих протоколів обміну інформацією. Жодних апріорних припущень щодо поведінки компонентів не потрібно при використанні порівняння біт за біт. Домінуючим фактором відмови правильно розробленої архітектури системи BR є збій синфазного режиму.

Наслідки несправності необхідно маскувати, доки не можна буде вжити заходів щодо відновлення. Необхідно керувати резервною системою, щоб продовжувати правильну роботу за наявності несправності. Одним з підходів є розділення надлишкових елементів на окремі зони локалізації несправності (FCR). FCR — це набір компонентів, який працює належним чином незалежно від будь-якої логічної чи електричної несправності за межами регіону. Межа локалізації несправностей вимагає, щоб апаратні компоненти були забезпечені незалежними джерелами живлення та синхронізації. Інтерфейси між FCR повинні бути електрично ізольовані. Стійкість до такого фізичного пошкодження, як попадання зброї, вимагає фізичного розділення FCR, наприклад, різних відсіків авіоніки. У системах керування польотом канал може бути природним FCR. Наслідки несправності, що проявляються як помилкові дані, можуть поширюватися через межі FCR. Таким чином, система також повинна забезпечувати стримування помилок. Це робиться за допомогою виборців у різних точках обробки, включаючи голосування за надлишковими входами, голосування за

результат обчислень закону керування та голосування на вході приводу. Маскування несправностей і помилок забезпечує коректну роботу системи з необхідністю негайної оцінки пошкоджень, ізоляції несправностей і реконфігурації системи.

Апаратні блокування забезпечують перший рівень захисту перед реконфігурацією або використанням решти несправних каналів. У триплексній або вищій системі резервування більшість каналів можуть вимкнути вихід несправного каналу. Перш ніж виконати цю дію, система визначить, чи є збій постійним чи тимчасовим.

Після того, як система визначить, що збій є постійним або постійним, наступним кроком є визначення того, які функції потрібні для решти місії та чи потрібно системі викликати оцінку збитків, ізоляцію збоїв і реконфігурацію решти системних активів. Розробник системи, необхідної для довготривалих місій, може взяти на себе реалізацію проекту, що має можливість реконфігурації.

Гібридна відмовостійкість використовує гібридне резервування, яке є комбінацією статичної та динамічної надмірності, тобто маскування, виявлення та відновлення, що може передбачати реконфігурацію. Система, яка використовує гібридне резервування, матиме N-активні резервні модулі, а також запасні (S) модулі. Детектор розбіжностей визначає, чи вихідні дані будь-якого з активних модулів відрізняються від вихідних даних вибірковий елемент. Якщо вихідний сигнал модуля не відповідає виборцю, схема комутації замінює несправний модуль на запасний. Гібридна (N, S) система не може мати більше (N-1)/2 несправних модулів одночасно в ядрі, інакше система неправильно вимкне справний модуль, якщо два з трьох вийшли з ладу.

Гібридна відмовостійкість використовує комбінацію маскування та реконфігурації, як зазначено в другому розділі. Намір полягає в тому, щоб використати сильні сторони обох підходів для досягнення чудової відмовостійкості. Маскування запобігає впливу помилкового стану на роботу

системи, таким чином усуваючи потребу у виправленні помилок. Реконфігурація видаляє помилкові входи для вибірковий елемент, щоб численні помилки не могли перемогти вибірковий елемент. Дії маскування та реконфігурації зазвичай реалізуються в механізмі порівняння виборців, який обговорюється в розділі

На рисунку 3.9 зображено гібридне розташування TMR із резервним каналом, що забезпечує подвійну можливість роботи при збоях. При виході з ладу першого активного каналу, він вимикається з конфігурації введення вибірковий елемент, і вмикається резервний канал. При виході з ладу другого каналу, невідповідний вхід для вибірковий елемент вимикається. Тоді залишаються лише два входи, тому наступний (третій) збій каналу може бути виявлений, але не ідентифікований виборцем як таким. З виборцем, який вибирає нижчий із двох сигналів, що залишилися, і, отже, виключає вихідний сигнал, постійне неправильне порівняння призводить до втрати функції системи при відмові.

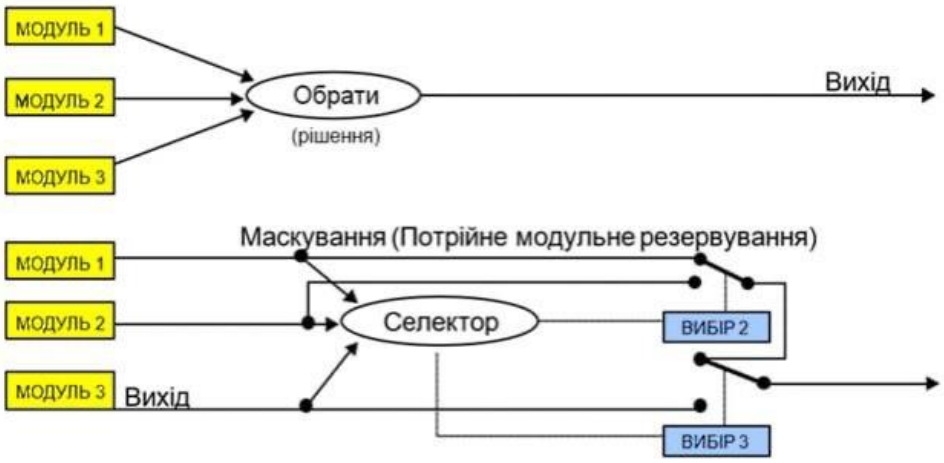


Рис. 3.9. Маскування проти реконфігурації.

Альтернативна робоча конфігурація з подвійною невдачею відмовилася б від перемикання резервних каналів і просто використовувала б квадруплексний виборець. Ця архітектура насправді досить поширена в спеціальних критично важливих системах, таких як системи управління польотом Fly-by-wire (FBW). Ця архітектура все ще використовує реконфігурацію для видалення помилкових вхідних даних для вибіркового

елемента.

Елементи проектування відмовостійкості, описані в розділі нижче, відображені у відмовостійкій архітектурі на рис. 45. Наприклад, виявлення помилок забезпечується компараторами; потім оцінка пошкоджень виконується логікою реконфігурації з використанням різних станів компаратора. усунення несправностей і продовження служби реалізуються через вибір, що також усуває потребу у виправленні помилок. І, нарешті, усунення несправності виконується шляхом перемикання несправного шляху, запропонованого логікою реконфігурації. Таким чином, цей простий приклад ілюструє на високому рівні, як різні аспекти відмовостійкості можуть бути включені в інтегрований дизайн.

Отже, розробники повинні враховувати ці вразливості при проектуванні відмовостійкої системи.

Аналіз надійності системи авіоніки залежить від припущень про готовність системи до руху. Для систем з меншою критичністю певне зменшення надлишковості іноді може бути прийнятним під час руху. Однак для постійно важливих для польоту систем, як правило, є повністю працездатна система з усім резервуванням.

Передбачається в прогнозі надійності системи. Це припущення висуває значні вимоги до самоперевірки системи перед польотом щодо охоплення та достовірних значень. Такий тест, як правило, є наскрізним тестом, який виконує всі елементи поетапно, що було б неможливо під час польоту. Положення про відмовостійкість вимагають особливої уваги. Наприклад, таке тестування змушує рідко використовувані відключення компаратора, щоб переконатися у відсутності прихованих несправностей, таких як пасивні збої обладнання. Аналіз пов’язаних схем тестування та їхнього охоплення обов’язково є поточним завданням аналізу проекту під час розробки. Ці схеми також повинні включати відповідні логічні блокування для забезпечення безпечного виконання попередніх випробування, наприклад, блокування ваги на колесах, щоб виключити тестування, окрім як на землі.

Оскільки, програмування системних самоперевірок може бути виконано відносно повним і високоточним способом.

Через характер дискретного часу цифрових систем не вся ємність використовується для прикладних функцій. Таким чином, під час польоту можливі періодичні самоперевірки цифрових компонентів, таких як процесори. Також процесори можуть періодично перевіряти стан справності інших компонентів системи. Такі тести забезпечують самоконтроль, який може виявити наявність невідповідності до появи станів помилки або перевищення порогів виявлення. Час виконання самотестування може бути значним, оскільки постійний політ може не імітувати поїздки компаратора через сигнали низької амплітуди. Крім того, чим довше несправність залишається латентною, тим більша ймовірність появи другої несправності. Таким чином, періодичні самоперевірки можуть значно підвищити надійність і безпеку системи шляхом зменшення впливу множинних одночасних проявів несправностей.

Самоконтроль може використовуватися на ще нижчих рівнях, але існує компроміс щодо деталізації виявлення несправностей. Цей компроміс залежить від відповіді на виявлення/відновлення помилок і вибраного рівня локалізації помилок. Загалом стримування несправностей визначає деталізацію виявлення несправностей, якщо тільки час реакції відновлення не вимагає швидшого виявлення несправностей, що найкраще досягається на нижчих рівнях.

Компаратори обрання дуже широко використовуються в стійких до відмов системах авіоніки, і вони, як правило, життєво важливі для цілісності та безпеки пов’язаних систем. Через вирішальну роль компараторів для виборців необхідно приділяти особливу увагу їх розвитку. Ці динамічні елементи системи, які можуть бути реалізовані як програмним, так і апаратним забезпеченням, не такі прості, як може здатися. Зокрема, цілісність пристрою та налаштування порогових параметрів можуть бути проблематичними.

Певні основні елементи та проблеми застосовуються до діапазону варіантів порівняння виборців. Концептуальний вигляд триплексного вибірковий елемент-компаратора зображено на рисунку 3.10.

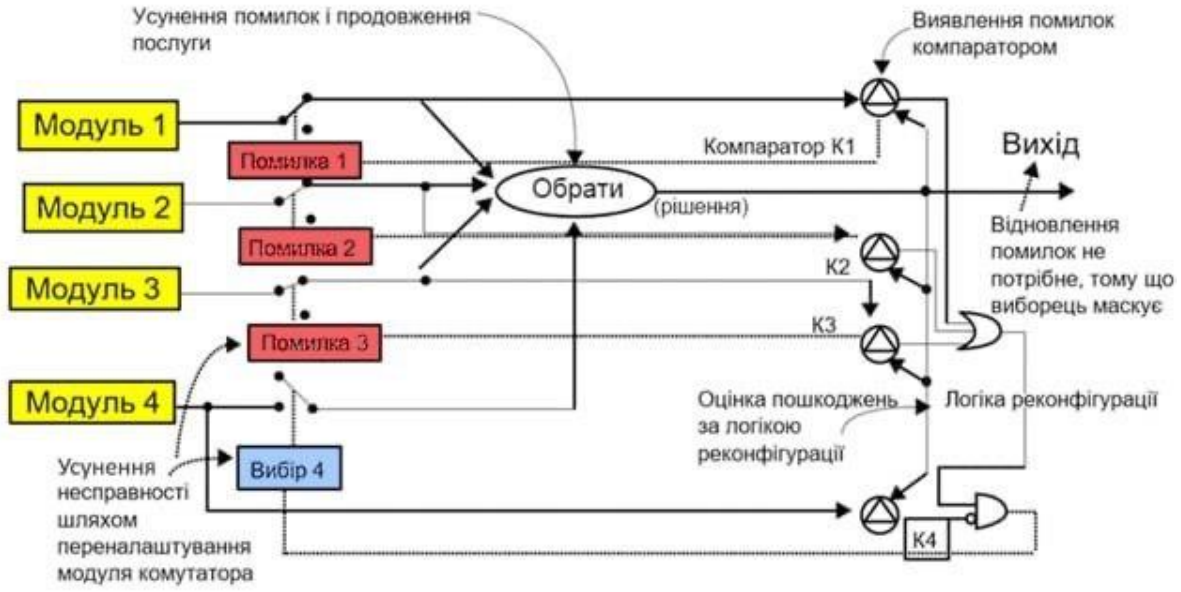


Рис. 3.10. Гібридна схема TMR.

Вибірковий елемент у цьому випадку реалізований як середній селектор сигналу, тобто як вихід обирається середнє (проміжне) значення з трьох вхідних сигналів. Блок вибіркового елементу розташований перед компараторами, оскільки саме його вихід подається на вхід кожного компаратора. Вихід вибіркового елемента розглядається як еталонний (опорний) сигнал, а будь-який вхідний сигнал, що суттєво відхиляється від цього еталона, вважається помилковим.

На рис. 3.10 показано, що відповідні входи кожного з сигналових каналів є амплітудно-модульованими серіями імпульсів, як це типово для цифрової обробки сигналів. Кожна ітерація роботи вибірковий елемент є незалежним актом вибору, тому вихідний сигнал може бути сформований на основі будь-якого з трьох вхідних каналів.

Це видно на рисунку 3.11, де компоненти вихідної серії імпульсів пронумеровані для вхідного шляху, вибраного в кожен момент часу. З кожним кроком часу вихід вибірковий елемент подається на кожен із компараторів, а різниця з кожним вхідним сигналом подається на

відповідний пороговий детектор амплітуди. Поріг амплітуди встановлюється таким чином, щоб накопичені допуски не спровокували спрацьовування детектора. Як показано, детектор амплітуди видає встановлений вихід, коли вперше спостерігається надмірна різниця. Коли різниця знову падає в межах порогу, видається вихід скидання.

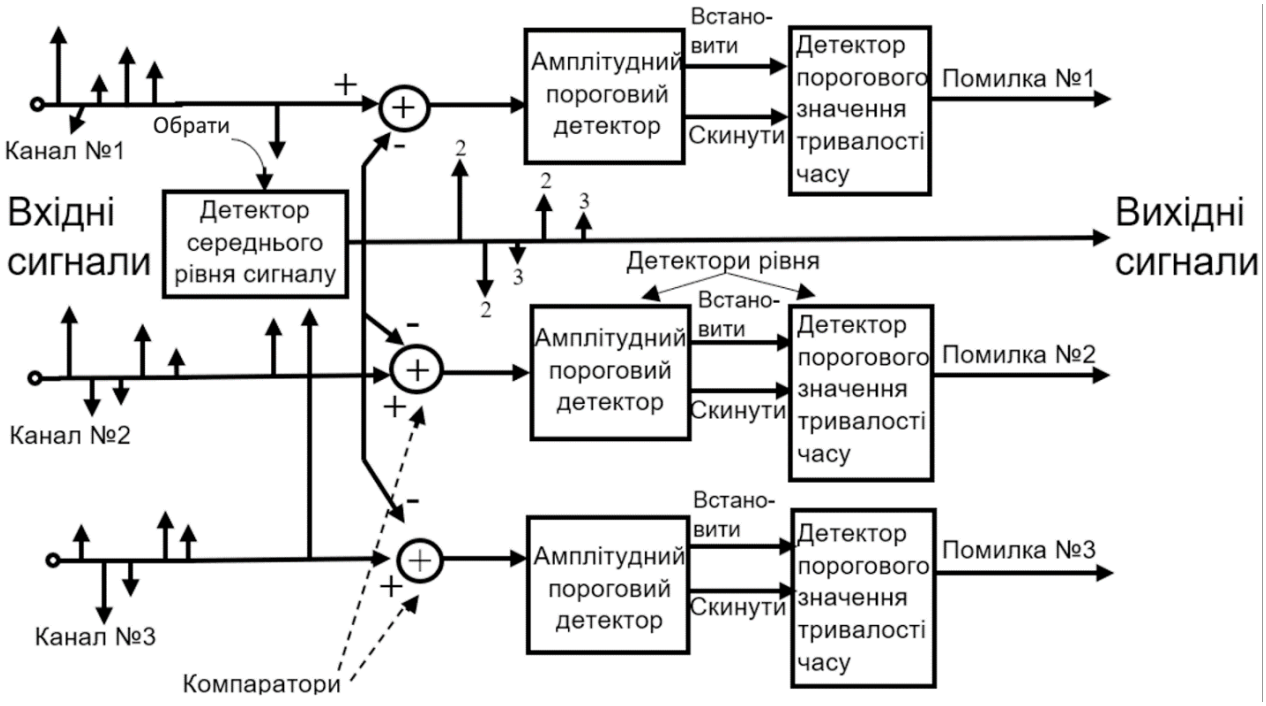


Рис. 3.11 Триплексний вибірковий елемент-компаратора.

Оскільки перехідні процеси можуть призвести до короткочасних відключень амплітудного детектора, до виходу кожного амплітудного детектора застосовується поріг синхронізації. Як правило, необхідна певна кількість послідовних порогових відключень амплітуди за межами допуску, щоб оголосити несправний сигнал. Таким чином, детектор порогового значення тривалості часу починає підрахунок щоразу, коли надходить встановлений сигнал, і за відсутності подальших вхідних даних збільшує відлік для кожного інтервалу вибірки після цього. Якщо задана кількість циклів перевищена, оголошується помилковий стан і для уражених каналів встановлюється логічний сигнал несправності. В іншому випадку підрахунок повертається до нуля, коли надходить сигнал скидання.

Встановлення порогів часу та амплітуди має вирішальне значення

через компроміс між логічними відключеннями неприємної несправності та повільною реакцією на фактичні несправності. Неприємні відключення підривають довіру користувачів до системи, їх невиправдані відключення потенційно можуть призвести до виснаження ресурсів. З іншого боку, запізніла реакція на помилку може призвести до виникнення небезпечних умов або катастрофічної події. Дозволений час для відновлення після даного типу несправності, який залежить від програми, є ключем до правильного встановлення порогових значень. Ступінь синхронізації каналів і спотворення даних також впливають на порогові налаштування, оскільки вони повинні враховувати будь-яку похибку. Компроміс може стати досить проблематичним, якщо потрібне швидке відновлення несправності.

Оскільки цілісність і функціональність системи поставлені під загрозу, детальний проект компаратора виборців має бути предметом ретельної оцінки на всіх етапах розробки. У випадку з апаратно реалізованим пристроєм необхідно ретельно перевірити його аспекти виявлення несправностей. Пасивні збої в схемах, які зазвичай не використовуються, викликають основне занепокоєння. Вбудований тест, самоконтроль або симптоми збоїв є звичайними підходами до цілісності пристрою. У випадку компараторів виборців, реалізованих програмним забезпеченням, їхню надійність можна посилити за допомогою формальних методів перевірки та коду, перевіреного під час експлуатації.

Сторожові таймери можна використовувати, щоб уловлювати як апаратне, так і програмне забезпечення, що переходять у небажані стани. Перевірки часу є формою перевірки тверджень. Цей вид перевірки корисний, оскільки багато помилок програмного та апаратного забезпечення проявляються у надмірному часу, який витрачається на певну операцію. У синхронних архітектурах потоку даних дані мають надходити в певний час. Помилки передачі даних такого типу можна виявити за допомогою таймера.

Програмне забезпечення відіграє вирішальну роль у цифрових системах. Термін «програмно реалізована відмовостійкість», який

використовується в цьму розділі, використовується в ширшому сенсі, вказуючи на роль програмного забезпечення в реалізації відмовостійкості.

Програмне забезпечення відіграє важливу роль у виявленні помилок. Виявлення помилок на системному рівні має базуватися на специфікації поведінки системи. Виходи системи слід перевіряти, щоб переконатися, що виходи відповідають специфікаціям. Ці перевірки мають бути незалежними від системи. Оскільки вони реалізовані в програмному забезпеченні, перевірки вимагають доступу до інформації, що перевіряється, і тому можуть потенційно зіпсувати цю інформацію. Отже, незалежність між системою та її перевіркою не може бути абсолютною. Надання ідеальних перевірок для виявлення помилок рідко є практичним, і більшість систем використовують перевірки на прийнятність .

Вирішити, де використовувати виявлення системних помилок, непросто. Перевірки на ранніх стадіях не слід розглядати як заміну перевіркам в останній момент. Перевірки на ранніх стадіях обов’язково базуватиметься на знанні внутрішньої роботи системи і, отже, не буде незалежною від системи. Перевірки на ранніх стадіях може виявити помилку на самих ранніх стадіях і таким чином мінімізувати поширення пошкоджень.

Перевірка в останній момент гарантує, що жоден вихід системи не залишиться неперевіреним. Тому в системі повинні бути передбачені як перевірки в останній момент, так і перевірки на ранніх стадіях.

Щоб виявити помилки програмного забезпечення, необхідно, щоб резервні версії програмного забезпечення були незалежними одна від одної, тобто мали різний проект.

Якщо очікуються помилки проекту, необхідно забезпечити реплікацію з використанням версій системи з різними проектами. Перевірки реплікації порівнюють два набори результатів, отриманих як вихідні дані реплікованих модулів. Перевірка реплікації піднімає позначку помилки та ініціює запуск інших процесів, щоб визначити, який компонент або канал є помилковим.

Перевірки часу використовуються для виявлення наявності

несправностей у системі, але не їх відсутності. У синхронних системах жорсткого реального часу повідомлення, що містять дані, передаються через шини даних за певним розкладом. Неотримання повідомлення в запланований час є помилкою. Помилка може бути спричинена несправністю датчика, шини даних тощо. У цьому випадку, якщо дані були критичними, методом допущення несправності може бути використання прямої екстраполяції стану.

Скасована перевірка бере виходи з системи та обчислює, якими мали бути вхідні дані, щоб отримати цей вихід. Потім розраховані вхідні дані можна порівняти з фактичними вхідними даними, щоб перевірити, чи є помилка. Системи, що забезпечують математичні функції, часто піддаються сторнованій перевірці.

Аналітична надлишковість з використанням будь-якого з двох загальних методів виявлення помилок, множинної моделі (ММ) або узагальненого відношення правдоподібності, є формою зворотної перевірки. Обидва методи використовують модель системи, представлену фільтрами Калмана. ММ намагається обчислити міру того, наскільки добре відстежує кожен із фільтрів Калмана, дивлячись на помилки передбачення. Реальні системи мають нелінійність, і модель передбачає лінійну систему. Питання полягає в тому, чи відповідає помилка відстеження від розширеного фільтра Калмана лінеаризованій моделі, «найближчій до» справжньої нелінійної системи, і чи є вона помітно меншою, ніж помилки від фільтрів, заснованих на «більш віддалених» моделях. Відмови приводів і датчиків можна моделювати різними способами за допомогою цієї методології.

Узагальнене співвідношення правдоподібності (GLR) використовує формулювання, подібне до формулювання для ММ, але настільки інше, що структура рішення є зовсім іншою. Відправною точкою GLR є модель, що описує нормальну роботу спостережуваних сигналів або системи, з якої вони надходять. Оскільки GLR безпосередньо спрямований на виявлення різких змін, його застосування обмежене проблемами, пов’язаними з такими

змінами, такими як виявлення збоїв. GLR, на відміну від MM, вимагає одного фільтра Калмана. Будь-який виявлений збій демонструватиме систематичне відхилення між тим, що спостерігається, і тим, що передбачається спостерігати. Якщо ефект параметричного збою «достатньо близький» до ефекту адитивного, система працюватиме.

В основі обох методів GLR і MM лежить проблема використання надмірності системи для генерації сигналів порівняння, які можна використовувати для виявлення збоїв. Фундаментальна ідея пошуку сигналів порівняння полягає у використанні надмірності системи, тобто відомих взаємозв'язків між виміряними змінними, щоб генерувати сигнали, які є малими за нормальної роботи та які відображають передбачувані моделі, коли розвиваються певні аномалії. Усі виявлення несправностей базуються на аналітичних взаємозв'язках між вимірюваними змінними, включаючи методи голосування, які припускають, що датчики вимірюють ту саму змінну. Компроміс із використанням аналітичних зв'язків полягає в тому, що ми можемо зменшити надмірність апаратного забезпечення та підтримувати той самий рівень працездатності від збоїв. Крім того, аналітична надлишковість дозволяє отримувати більше інформації з даних, дозволяючи виявити тонкі зміни в характеристиках компонентів системи. З іншого боку, використання цієї інформації може спричинити проблеми, якщо існують великі невизначеності в параметрах, що визначають аналітичні зв'язки.

Другою частиною алгоритму виявлення відмов є правило прийняття рішень, яке використовує доступні сигнали порівняння для прийняття рішень щодо переривання нормальної роботи шляхом оголошення відмов. Однією з переваг цих методів є те, що правило прийняття рішень — максимізація та порівняння з порогом — є простим, тоді як основним недоліком є те, що правило явно не відображає бажаних компромісів. Підхід байєсівського послідовного рішення, в якому алгоритм для розрахунку приблизного рішення Байєса, має прямо протилежні властивості, тобто дозволяє пряме включення компромісів продуктивності, але є надзвичайно складним.

Проблема послідовного прийняття рішень Байєса полягає у виборі правила зупинки та правила кінцевого рішення, щоб мінімізувати загальну очікувану вартість та очікувану вартість, яка накопичується до зупинки.

Перевірки кодування базуються на надмірності у представленні об’єкта, що використовується в системі. Всередині об’єкта надлишкові дані підтримуються в певному фіксованому зв’язку з (не надлишковими) даними, що представляють значення об’єкта. Перевірка на парність є добре відомим прикладом перевірки кодування, як і коди виявлення та виправлення помилок, такі як код Хеммінга, циклічна перевірка надмірності та арифметичні коди.

Ці перевірки базуються на знанні мінімальних і максимальних значень вхідних даних, а також обмежень швидкості зміни вхідних даних. Ці перевірки базуються на знанні фізичної роботи датчиків і використовують моделі цієї роботи.

До структур даних в обчислювальній системі можна застосовувати дві форми перевірок. Перевірки семантичної цілісності даних стосуватимуться узгодженості інформації, що міститься в структурі даних. Перевірки структурної цілісності стосуватимуться того, чи є сама структура узгодженою. Наприклад, зовнішні дані з підсистем передаються з цифрових шин даних, таких як MIL-STD-1553, ARINC 429 або ARINC 629. Вміст повідомлення (кількість слів у повідомленні, вміст кожного слова) з підсистеми зберігається а вхідні дані перевіряються на відповідність.

Діагностичні перевірки створюють вхідні дані для апаратних елементів системи, які мають видавати відомий вихід. Ці перевірки рідко використовуються як основний засіб виявлення помилок. Зазвичай вони запускаються під час запуску та можуть бути ініційовані оператором як частина вбудованого тесту. Вони також можуть працювати безперервно у фоновому режимі, коли процесор може простоювати. Також проводяться діагностичні перевірки, щоб виявити певні несправності.

При виявленні помилки, необхідно визначити ступінь пошкодження,

завданої нею, перш ніж можна буде виконати відновлення помилки. Оцінка ступеня збитку зазвичай пов'язана зі структурою системи. За умови своєчасного виявлення помилок оцінка збитку зазвичай обмежується поточним обчисленням або процесом. Припускається, що стан є відповідним при вході. Перед виходом із поточного обчислення виконується перевірка виявлення помилок. Припускається, що будь-які виявлені помилки викликані помилками в поточному обчисленні.

Після визначення масштабу пошкодження важливо відновити систему до нормального стану. Існує два основні підходи — зворотне та пряме відновлення помилок. При зворотному відновленні помилок система повертається до попереднього узгодженого стану. Потім поточне обчислення можна повторити з наявними компонентами, з альтернативними компонентами, або його можна проігнорувати. Використання зворотного відновлення передбачає можливість збереження та відновлення стану. Зворотне відновлення помилок не залежить від оцінки збитку.

Упереджене відновлення помилок намагається продовжити поточне обчислення, відновлюючи систему до узгодженого стану, компенсуючи невідповідності, виявлені в поточному стані. Попереднє відновлення помилок передбачає детальне знання масштабу завданої шкоди та стратегію усунення невідповідностей. Пряме відновлення помилок важче реалізувати, ніж зворотне відновлення помилок.

Після відновлення системи після помилки може бути бажаним ізолювати та/або виправити компонент, який спричинив помилку. Усунення несправностей не завжди є необхідним через тимчасовий характер деяких несправностей або через те, що процедур виявлення та відновлення достатньо для вирішення інших повторюваних помилок. Для постійних несправностей усунення несправностей стає важливим, оскільки маскування постійних несправностей знижує здатність системи справлятися з наступними несправностями. Деякі технології відмовостійкого програмного забезпечення намагаються виділити помилки в поточних обчисленнях

шляхом своєчасного виявлення помилок. Після виділення несправності лікування несправності можна виконати шляхом переналаштування обчислень для використання альтернативних форм обчислень, щоб забезпечити безперервне обслуговування. (Це можна робити послідовно, як у блоках відновлення, або паралельно, як у програмуванні NVersion.) Припущення полягає в тому, що пошкодження через помилки належним чином інкапсульовано до поточного обчислення, а саме виявлення помилок є бездоганним (тобто виявляє всі помилок і не викликає власних причин).

Багатопроцесорні архітектури, що складаються з обчислювальних ресурсів, пов'язаних між собою зовнішніми шинами даних, повинні проектуватися як розподілена відмовостійка система. Обчислювальні ресурси можуть бути встановлені в корпусі за допомогою паралельної об’єднавчої шини для реалізації багатопроцесорної обробки всередині корпусу. Кожен корпус можна вважати віртуальним вузлом у загальній мережі. Мережева операційна система в поєднанні з шинами даних і їх протоколом завершує відмовостійку розподілену систему. Архітектура може бути асинхронною, слабко синхронною або жорстко синхронною. Підтримувати узгодженість даних через резервні канали асинхронних систем важко.

Помилки програмного забезпечення, які вважаються помилками проектування, можуть виникати під час розробки вимог, створення специфікацій, проектування архітектури програмного забезпечення, створення коду та інтеграції коду. Хоча під час системної інтеграції та тестування можна знайти й усунути багато несправностей, практично неможливо усунути всі можливі помилки розробки програмного забезпечення.

Отже, використовується відмовостійкість програмного забезпечення. Двома основними методами, які використовуються для забезпечення відмовостійкості програмного забезпечення, є програмне забезпечення N-версії та блоки відновлення.

Багатоверсійне програмне забезпечення — це будь-яка відмовостійка

програмна техніка, у якій реалізовано, виконано дві або більше альтернативних версій, а результати порівнюються за допомогою певної форми алгоритму прийняття рішень. Мета полягає в тому, щоб розробити ці альтернативні версії таким чином, щоб помилки програмного забезпечення, які можуть існувати в одній версії, були відсутні. Не повна відмовостійка техніка програмного забезпечення, виявляє лише помилки міститься в іншій версії, і алгоритм прийняття рішень визначає правильне значення серед альтернативних версій.

Незалежно від того, які засоби використовуються для створення альтернативних версій, спільною метою є наявність різних версій програмного забезпечення, щоб імовірність одночасного виникнення помилок була невеликою і щоб помилки можна було розрізнити, коли результати виконання мультиверсій порівнюються один з одним.

Функція порівняння виконується як алгоритм прийняття рішень після отримання результатів від кожної версії. Алгоритм прийняття рішень вибирає відповідь або сигналізує, що він не може визначити відповідь. Цей алгоритм прийняття рішень і розробка альтернативних версій є основним методом виявлення помилок. Оцінка збитку передбачає, що збиток обмежується інкапсуляцією окремих версій програмного забезпечення. Несправні компоненти програмного забезпечення маскуються так, що несправності обмежуються модулем, у якому вони виникають. Відновлення несправного компонента може бути зроблено або не зроблено.

N-версій програми незалежно створюють N-групи інженерів програмного забезпечення, які працюють на основі загальної специфікації. Кожна версія виконується незалежно від інших версій. Кожна версія повинна мати доступ до ідентичного набору вхідних значень, а результати порівнюються виконавцем, який вибирає використаний результат. Вибір точного чи неточного алгоритму перевірки голосування залежить від критичності функції та часу, пов’язаного з голосуванням.

Друга основна техніка, наведена в таблиці — це блок відновлення та

його підкатегорії — механізм кінцевих термінів і різне програмне забезпечення резервного копіювання. Техніка блоку відновлення розпізнає ймовірність того, що в програмному забезпеченні можуть існувати залишкові помилки. Замість того, щоб розробляти незалежні надлишкові модулі, ця методика покладається на використання програмного модуля, який виконує приймальний тест на вихідних даних основного модуля. Приймальний тест викликає виключення, якщо стан системи неприйнятний. Наступним кроком є оцінка збитку та усунення несправності. Враховуючи, що помилка конструкції в основному модулі могла спричинити довільну шкоду стану системи, і що точний час, коли були згенеровані помилки, неможливо визначити, найбільш підходящим попереднім станом для відновлення є стан, який існував безпосередньо перед основним модулем було введено.

Охоплення великої кількості несправностей має пов’язані накладні витрати з точки зору надмірності та обробки, пов’язаної з виявленням помилок. Розробник може використовувати моделювання та симуляцію, щоб визначити кількість резервування, необхідного для реалізації відмовостійкості, порівняно з ймовірністю та впливом різних типів несправностей. Якщо помилка має мінімальний вплив або взагалі не впливає на безпеку чи виконання місії, інвестиції в резервування для усунення цієї помилки можуть бути неефективними, навіть якщо ймовірність виникнення помилки є значною.

Відмовостійкі системи повинні використовуватися щоразу, коли збій може призвести до втрати життя або втрати цінного активу. Фізичні збої апаратного забезпечення зменшуються, тоді як збої в дизайні практично неможливо повністю усунути.

При застосуванні відмовостійкості до складної системи існує небезпека того, що нові механізми можуть створити додаткові джерела збоїв через помилки проектування та реалізації. Тому важливо, щоб нові механізми були впроваджені таким чином, щоб зберегти цілісність конструкції з мінімальною додатковою складністю. Розробник повинен використовувати

засоби моделювання та симуляції, щоб переконатися, що проект забезпечує необхідну відмовостійкість.

Для спрощення процесу прийняття проектних рішень розроблено певні принципи проектування. Інкапсуляція та ієрархія пропонують способи досягнення простоти та загальності в реалізації окремих функцій відмовостійкості. Інкапсуляція забезпечує:

– організація даних і програм як єдиних об’єктів із суворим контролем взаємодії об’єктів.

– Організація наборів альтернативних версій програм у відмовостійкі програмні модулі (наприклад, блоки відновлення та набори програм N-версій).

– Організація узгоджених наборів точок відновлення для кількох процесів.

– Організація зв'язків між розподіленими процесами як атомарних (неподільних) дій.

– Організація функцій операційної системи в відновлювані модулі.

– Приклади принципу ієрархії, який використовується для підвищення надійності функцій відмовостійкості, включають:

– Організація всього програмного забезпечення, як прикладного, так і системного типу, на рівні, з односпрямованими залежностями між шарами.

– Інтеграція сервісних функцій і функцій відмовостійкості на кожному рівні.

– Використання вкладених блоків відновлення для забезпечення можливості ієрархічного відновлення.

– Організація функцій операційної системи таким чином, що лише мінімальний набір на найнижчому рівні («ядро») повинен бути звільнений від відмовостійкості.

– Інтеграція глобальних і локальних даних і управління в розподілених процесорах.

Тій частині ядра операційної системи, яка забезпечує базові механізми,

які решта системи використовує для досягнення відмовостійкості, слід «довіряти». Це ядро має бути обмеженої складності, щоб можна було перевірити всі можливі шляхи для забезпечення правильної роботи за будь-якої логіки та умов даних. Це ядро не повинно бути відмовостійким, якщо можна гарантувати вищезазначене.

Безпека визначається з точки зору небезпек і ризиків. Небезпека – це стан або сукупність умов, які можуть призвести до нещасного випадку за відповідних обставин. Рівень ризику, пов’язаного з небезпекою, залежить від імовірності того, що небезпека станеться, ймовірності нещасного випадку, якщо небезпека все ж відбудеться, і потенційних наслідків аварії.

Валідація — це процес, за допомогою якого системи демонструють роботу, яка відповідає специфікаціям. Процес перевірки починається, коли специфікація завершена. Визнали складність розробки точної специфікації, яка ніколи не зміниться. Ця реальність призвела до ітеративного процесу розробки та перевірки. Перевірка вимагає розробки тестових випадків і виконання цих тестових випадків на апаратному та програмному забезпеченні, що входить до складу системи, яка буде доставлена. Випробування повинні охоплювати 100% помилок, які система розроблена для терпимості, і дуже високий відсоток можливих проектних помилок, будь то апаратне забезпечення, програмне забезпечення або взаємодія апаратного та програмного забезпечення під час виконання всіх можливих даних і логічних шляхів. Після перевірки системи її можна вводити в експлуатацію. Щоб мінімізувати потребу перевіряти повну оперативну польотну програму кожного разу, коли вона змінюється, методи розробки намагаються розкласти велику систему на модулі, які незалежно визначаються, реалізуються та перевіряються. Лише ті модулі та їхні інтерфейси, які були змінені, повинні бути повторно перевірені за допомогою цього підходу.

Швидке створення прототипів, моделювання та анімація – це всі методи, які допомагають перевірити систему. Формальні методи використовуються для розробки та перевірки цифрових систем авіоніки.

Існують аргументи як на користь, так і проти використання формальних методів.

3.3. Проектування та тестування системи керування авіоніки в реальному часі за допомогою застосунків інформаційної підтримки.

Проектування та тестування системи керування авіоніки в реальному часі за допомогою Simulink та XtratuM включає :

1.Системне моделювання: створення моделі системи керування в Simulink, в якій відображені всі необхідні компоненти, процеси та взаємодії.

2. Відображення реального часу: Використання функцій Simulink для відображення роботи системи в реальному часі, що дозволяє врахувати часові обмеження та підтримувати синхронізацію з зовнішніми подіями.

3. Генерація коду для виконання: генерація коду з моделі Simulink для виконання на виконавчій платформі, яка підтримує ARINC 653, з використанням XtratuM.

4.Тестування та валідація: проведення тестів для перевірки коректності та ефективності роботи системи в реальному часі. Валідація полягає в перевірці, чи система відповідає вимогам та специфікаціям.

5. Інтеграція та підтримка: інтеграція системи керування з іншими компонентами авіоніки та забезпечення підтримки системи після введення в експлуатацію.

Цей процес дозволяє ефективно розробляти та тестувати системи керування авіоніки в реальному часі з використанням Simulink та XtratuM, забезпечуючи високу надійність та ефективність системи.

Використання Simulink та XtratuM забезпечує високу надійність та ефективність систем управління. Simulink дозволило моделювати та верифікувати систему управління перед реалізацією на апаратному забезпеченні, що дозволяє виявити та виправити помилки ще до реального використання. Завдяки можливості автоматичної генерації коду за допомогою Embedded Coder, реалізували модель на цільовій платформі.

XtratuM, з іншого боку, є реальним часовим операційним середовищем (RTOS), яке забезпечує високу надійність та ефективність управління в реальному часі. Воно дає можливості, спрямовані на забезпечення безпеки та стійкості системи, що особливо важливо для авіаційних застосувань. Його сумісність з ARINC 653 RTOS дозволяє інтегрувати його з іншими системами, що може бути корисним у складних авіаційних системах.

Отже, Simulink та XtratuM можуть створюють високонадійну та ефективну систему управління, що відповідає вимогам авіаційної індустрії. програмного забезпечення. Отже, проектування та тестування системи керування в реальному часі з використанням Simulink, а також автоматичного створення коду для цільової системи під керуванням XtratuM RTOS [113], яка є сумісною з ARINC 653 RTOS. Цей процес проілюстровано на прикладі проектування відносно складного додатку, такого як система автопілота.

Процес проектування та тестування системи керування в реальному часі з використанням Simulink та автоматичного створення коду для XtratuM RTOS, сумісної з ARINC 653 RTOS, може бути реалізований через кілька основних кроків. Розглянемо цей процес на прикладі проектування системи автопілота:

– визначення вимог: визначені функціональні та технічні вимоги до системи автопілота, включаючи режими роботи, вхідні та вихідні сигнали, архітектуру системи;

– моделювання системи в Simulink: створіть модель системи автопілота в Simulink, в якій будуть відображені всі компоненти системи та їх взаємозв'язки.

– параметризація моделі: налаштування параметрів моделі відповідно до вимог XtratuM RTOS та ARINC 653 RTOS, такі як частота оновлення, пріоритети завдань;

– верифікація та тестування моделі: виконайте симуляцію моделі для перевірки її правильності та відповідності вимогам. Проведено

тестування на різних наборах вхідних даних для виявлення помилок;

– генерація вбудованого коду: Використовуйте Embedded Coder для генерації вбудованого коду з моделі Simulink для виконання на цільовій системі під керуванням XtratuM RTOS;

– тестування в реальному середовищі: виконано тестування системи в реальному середовищі з використанням XtratuM RTOS для перевірки правильності та ефективності роботи автопілота;

– оптимізація та вдосконалення: внесені зміни до моделі та коду на основі результатів тестування та вимог для покращення роботи системи.

Цей процес дозволяє ефективно розробляти та тестувати систему керування в реальному часі з використанням Simulink та XtratuM RTOS для реалізації складних додатків, таких як система автопілота, забезпечуючи високу надійність та ефективність роботи системи.

Simulink та Embedded Coder підтримують ключові етапи розробки на основі методу моделювання (Model-Based Design). Ця методологія проектування дозволяє легко моделювати складні системи (в тому числі нелінійні) і забезпечує автоматичну генерацію коду. Системи автоматичного кодування іноді вважаються ненадійними, оскільки вони схильні до надійними, оскільки вони схильні до створення "спагеті-коду". Однак, це не стосується Simulink, оскільки він має ряд супутніх інструментів для автоматизації перегляду вихідного коду згенерованого коду, для автоматизації раннього відстеження вимог для забезпечення перевірки відповідності стандартам моделювання та документування моделей. Перевагами цього процесу проектування на основі моделей є процесу на основі моделей є прискорення розробки тестових стендів та допомога досягненню цілей DO-178B, пов'язаних із завданнями верифікації завдяки використанню належних інструментів.

Іншим ключовим елементом на етапі розробки програмного забезпечення є операційна система RTOS, на якій будуть виконуватися додатки для авіоніки. Авіоніка - це, як правило, розподілені додатки, що

працюють в реальному часі і включають декілька обчислювальних модулів. RTOS повинна забезпечувати середовище виконання для виконання різних обчислювальних модулів програми, які повинні бути критично важливими для безпеки (як визначено авіаційними правилами), працювати в реальному часі та бути детермінованими [4]. Ці цілі можуть бути досягнуті за допомогою двох різних підходів: федеративної архітектури або інтегрованої архітектури.

Федеративні архітектури базуються на спеціальних обчислювальних модулях або процесорних блоках, розподілених по всьому транспортному засобу. Як правило, кожен модуль містить одну програму в системі авіоніки. Основна перевага такої архітектури полягає в тому, що вона забезпечує відмовостійкість: збої в одному модулі не впливають на інші модулі. В інтегрованих архітектурах кожен модуль містить безліч додатків різного рівня критичності. Така архітектура дозволяє зменшити кількість обчислювальних блоків, але вимагає надійного механізму розбиття на блоки та механізму моніторингу стану для ізоляції виконання різних модулів, як у федеративній архітектурі. У цьому підході обчислювальний блок розбивається на кілька віртуальних комп'ютерів, на кожному з яких розміщується додаток або модуль додатку . Інтегрована модульна авіоніка є найважливішим прикладом такої архітектури, яка стандартизована ARINC 653. Ця специфікація також стандартизує інтерфейс прикладного програмування (API). API, визначений ARINC 653, називається APEX – APplication Executive. Наш процес розробки програмного забезпечення базується на XtratuM, гіпервізорі реального часу, який забезпечує модель IMA за допомогою різних API. Одним з них є Lithos, який сумісний з ARINC 653. У статті описується, як Simulink проекти можуть бути націлені на виконання на XtratuM.

XtratuM - це гіпервізор для вбудованих систем реального часу, який значною мірою базується на концепції IMA . Архітектура IMA – це, по суті, інтегрована архітектура, заснована на концепції розділення. Ця концепція

виникла для захисту та розділення додатків з просторової та часової точок зору.

XtratuM віртуалізує основні апаратні пристрої (пам'ять, таймери та переривання) для одночасного виконання декількох ОС. Однією з таких гостьових ОС є Lithos, яка реалізує концепцію процесу, представлену в стандарті ARINC 653, яка відсутня в XtratuM. Іншим способом реалізації концепції процесу є використання Partikle, що реалізує POSIX-потоки.

Сервіси, які надає XtratuM через Lithos або Partikle, є тими, що визначені специфікацією ARINC 653, і коротко описані далі.

Керування розділами. Розділи – це середовища виконання з незалежними областями пам'яті та суворо захищеним часом виконання. Розділи планується відповідно до циклічного планувальника, який вказується у файлі конфігурації. Всі ресурси, що використовуються розділом (процеси, дошки, семафори, порти) повинні бути визначені під час конфігурування системи, створені та ініціалізовані під час фази ініціалізації розділу.

В основі розробки програмного забезпечення авіоніки лежить основоположний стандарт RTCA/DO-178B [18]. Незважаючи на його відособленість від безпосередньої картини створення програмного забезпечення, даний стандарт описує весь процес розробки і висуває вимоги до подібного ПЗ. Для розробки критично важливих систем авіоніки необхідно забезпечити мінімальну можливість помилки (для найвищого рівня в авіоніці встановлена ймовірність відмови [19, 3], а також мінімізувати витрати на розробку і виправлення коду. Через складність системи і взаємозв'язки її зі складними апаратними комплексами, ми розробили структурну модель інформаційної підтримки процесу проектування та експлуатації складних автоматизованих систем управління комплексу бортового, що покладено в основу при створені програмного забезпечення рисунку 3.12.

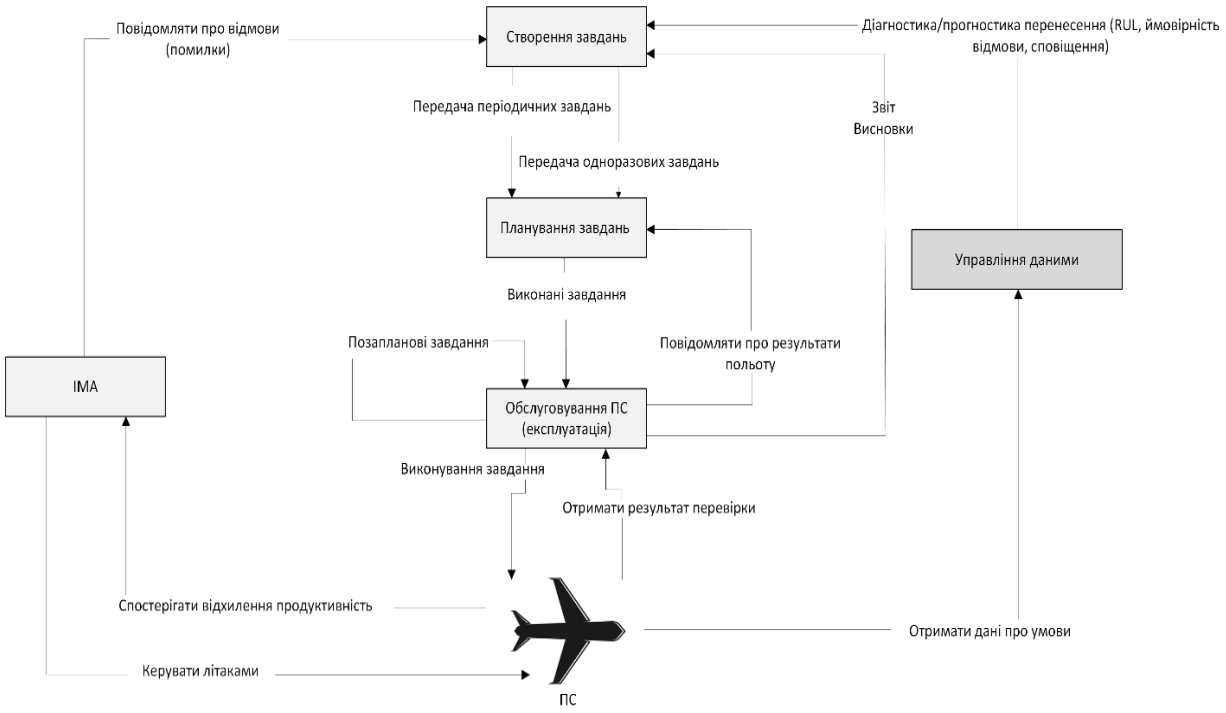


Рис.3.12. Структурна модель інформаційної підтримки процесу проектування та експлуатації складних автоматизованих систем управління комплексу бортового

Розроблена модель (рис. 3.12) дозволяє забезпечити ефективну синхронізацію дій усіх учасників проєкту на кожному етапі його життєвого циклу, особливо в умовах багатоітераційної розробки програмного забезпечення авіоніки. Вона створює основу для повторного використання вже апробованих даних, інструментів та методологічних підходів. Важливою особливістю є гнучкість цієї моделі, яка дозволяє адаптувати її до будь-якого типу проєкту або організаційної структури, незалежно від масштабу або специфіки завдання.

Такий підхід дає можливість розподілити процес розробки на окремі функціональні компоненти, кожен з яких має чітко визначені дії, супровідні інструкції, рекомендації та пояснення. Це є критично важливим для циклів розробки й тестування авіаційного ПЗ, оскільки дозволяє реалізовувати поетапне впровадження змін, контроль якості та валідацію.

Інформаційна система, що супроводжує модель, дозволяє оцінювати ризики на кожному етапі розробки, здійснювати контроль за навантаженням

фахівців (workload management) в умовах обмежених ресурсів і часу, а також підтримує прозору систему ухвалення рішень. Враховуючи складність сучасних Flight Control Systems, які є багаторівневими програмно-апаратними комплексами, створення подібної системи підтримки є критично необхідним. Жоден окремий фахівець не може охопити повну функціональність таких систем, тому акцент на структурованості та стандартизації процесу розробки є обґрунтованим.

Для забезпечення контролю якості та забезпечення подальшої сертифікації, процес розробки розділено на окремі рівні, кожному з яких відповідає набір документів. Для кожного документа формується контрольний журнал (беклог), що відстежує хід виконання, зафіксовані помилки, їх класифікацію та шляхи усунення. Повторення ітерацій з урахуванням зафіксованих невідповідностей дозволяє поступово знижувати ймовірність помилок та підвищувати рівень відповідності до вимог замовника та нормативів.

Усі документи створюються як на основі вимог замовника, так і відповідно до внутрішніх процедур компанії-розробника, з урахуванням стандартів сертифікації, таких як DO-178C.

Структура процесу розробки програмного забезпечення, орієнтованого на підтримку експлуатації повітряних суден, представлена на UML-діаграмі (рис. 14). Як видно з діаграми, замовник програмного забезпечення займає ключову позицію у формуванні вимог. Першим етапом процесу є аналіз цих вимог та розробка базової функціональності, що стає основою для створення концепції системи, технічної архітектури та специфікацій обладнання (Equipment Specification) і системних вимог (System Requirements).

Після узгодження основної технічної бази формується план розробки програмного забезпечення (Software Development Plan), а також план сертифікації програмного забезпечення (Qualification Plan / PSAC — Plan for Software Aspects of Certification). Слід зазначити, що розробка ПЗ в авіаційній галузі неможлива без тісної інтеграції з апаратною частиною, оскільки

більшість програмного забезпечення є вбудованим і функціонально залежним від компоновки бортових систем.

Таким чином, системна інтеграція, багаторівнева документація, повторювані ітерації розробки та використання адаптивної моделі управління процесом створюють надійне інформаційне середовище для забезпечення якості, безпеки та сертифікації програмного забезпечення авіоніки.

У цілому процес функціонування системи передбачає реалізацію сукупності прийомів, алгоритмів, дій та операцій, спрямованих на забезпечення відмовостійкості мікропроцесорного елемента. Враховуючи властивості інтегрованої модульної авіоніки (IMA), ймовірність безвідмовної роботи системи може бути подана як ймовірність безвідмовної роботи авіоніки у випадку застосування схеми ковзного резервування з навантаженим резервом рисунок 3.13.

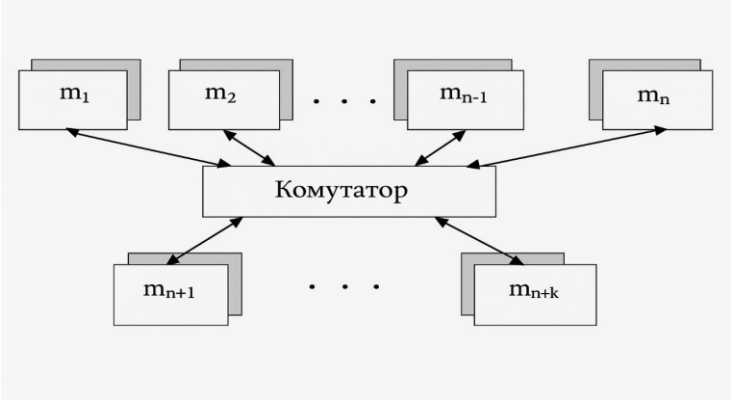


Рис.3.13 – Структурна схема надійності модулів IMA

По-перше, будь-який інформаційний тракт АЛП, що відмовив за основою $m_i(i=\overline{1,n})$ може бути замінений на будь-який справний контрольний тракт за основою $m_j(j=\overline{n+1,n-k})$; $m_i < m_j$, де n та k – відповідно кількість інформаційних та контрольних основ МА; по-друге, всі тракти модулів, як інформаційні, і контрольні, одночасно, незалежно й однаково беруть участь у обробці інформації.

Представлена на рис. 3.13 структурна схема, що відповідає надійності авіоніки в IMA з n інформаційними і k контрольними незалежними трактами. Комутатор здійснює контроль функціонування і перемикання трактів, що

відмовили на працездатні. Розглянемо, як, використовуючи запропоновану надійну модель, можна розрахувати ймовірність безвідмовної роботи.

Модуль керування використовує декілька режимів відповідно до різних фаз польоту: набір висоти, крейсерський політ або заходження на посадку. Під час набору висоти та крейсерського польоту VNAV перемикається між режимами VPATH HOLD і ALTITUDE HOLD в залежності від похибки висоти, в той час як LNAV висоти, в той час як LNAV може переходити в режим утримання висоти (HEADING HOLD) або розвороту (TURN). Залежно (ROLL HOLD) від помилки курсу. Під час заходження на посадку використовується інший режим керування: VNAV і LNAV контролюють вертикальну і бічну відстань до глісади схилу аналогічно до системи ILS.

Системи LNAV і VNAV вирішуються окремо. Вектори стану та керування векторами стану та керування для системи VNAV: нехай задана впорядкована (mi<mi+1) набором взаємно попарно простих чисел m1, m2,…, mn, mn+1,…, mn+k, де n та k відповідно кількість інформаційних та контрольних основ, а кратність визначиться величиною k/n. У цьому випадку формула для визначення ймовірності безвідмовної роботи набуде вигляду:

$$P_{MA}^{(k)}(t) = \sum_{i=0}^{k} C_{k+n}^{i} P_1^{k+n-i}(t) \sum_{j=0}^{i} (-1)^{j} C_i^j P_1^{j}(t), \qquad (3.1)$$

де $P_1(t) = e^{-\lambda_1 t}$ – ймовірність безвідмовної роботи з найбільшою (з найменш надійною) основою m_{n+k};

λ_1 – інтенсивність відмов обладнання з найбільшою основою m_{n+k}.

Співвідношення (3.1) може бути використане для розрахунку ймовірності безвідмовної роботи при наступних припущеннях:

- відмови трактів підпорядковані експоненційному розподілу;

- комутуючий пристрій ідеальний (тобто ймовірність безвідмовної роботи комутатора дорівнює одиниці);

- інформаційні та контрольні тракти обробки інформації рівнонадійні, тобто. ймовірність безвідмовної роботи всіх трактів приймається рівною

ймовірності безвідмовної роботи $P_1(t)$ тракту з найбільшою основою m_{n+k}, що має найменшу ймовірність безвідмовної роботи;

- не враховується можливість відновлення трактів, що відмовили.

Однією з основних проблем є розбиття програми на частини. У нашому контексті це означає розміщення блоків Simulink у різних розділах. Запропоноване рішення базується на використанні Simulink *Referenced models*. Посилання на моделі - це функція Simulink, яка дозволяє включати одну модель в іншу, посилаючись на неї. Модель, що посилається, має інтерфейс, який складається з портів вводу та виводу, а також аргументів-параметрів. Ці порти дозволяють включити модель, на яку є посилання, в блок-схему батьківської моделі.

Ми реалізуємо розділи, що відокремлюють моделі, які відповідають кожному розділу, в окремій моделі, що посилається. Ми вважаємо це рішення не надто обмежувальним, оскільки Simulink сам запитує про включення кожного блоку в моделі, на які посилаються, коли увімкнено паралельне виконання. Виходячи з цього, розбиття програми на розділи складається з наступних етапів :

Зазначимо, що реальна надійність буде вищою, ніж та, що визначається співвідношенням (3.1), оскільки дана формула не враховує можливість заміни одним контрольним трактом по основі m_j одного або одночасно кількох непрацездатних інформаційних трактів за умови

$m_j \geq \prod_{i=1}^{q} m_{k_i}$, де q – максимальна кількість робочих трактів, що одночасно замінюються, одним контрольним працездатним трактом за основою m_j [24].

Позначимо через λ_1 інтенсивність відмов обладнання, що віднесена до одного двійкового розряду. У цьому випадку ймовірність безвідмовної роботи обладнання відносно до одного двійковому розряду дорівнює $P_I(t) = e^{-\lambda_I t}$.

Для позиційного l – байтового ймовірність безвідмовної роботи дорівнює $P_0(t) = e^{-\lambda_0 t}$, де $\lambda_0 = 8l\lambda_1$.

Відомо, що ймовірність безвідмовної роботи для тройованої мажорної структури, що містить три модулі та ідеальний мажоритарний елемент, дорівнює [17]:

$$P_{MA}(t) = 3P_0^2(t) - 2P_0^3(t) = e^{-16\lambda_1 t}(3 - 2e^{-8\lambda_1 t}). \quad (3.2)$$

Ймовірність безвідмовної роботи з довільної основи $m_i (i = \overline{1,n+k})$ визначається, як $P_i(t) = e^{-\lambda_1 t}$ або $P_i(t) = e^{-\lambda_1 a_{n+k} t}$, де $a_{n+k} = [log_2(m_{n+k} - 1)] + 1$.

Імовірність безвідмовної роботи визначається відповідно до виразу (3.1).

Нехай $l = 1$ (однобайтовий) та $k = 1$. Тоді з урахуванням критерію мінімальності апаратурної надмірності можна подати у вигляді набору наступних основ (модулів) $m_1 = 3, m_2 = 4, m_3 = 5, m_4 = 7, m_5 = 11$. При цьому $\prod_{i=1}^{4} m_i = 420 > 2^8 = 256$ та НСД $(m_i, m_j) = 1$ для $i \neq j$. У цьому випадку співвідношення (3.1) запишемо у вигляді:

$$P_{MA}^{(1)}(t) = 5P_1^4(t) - 4P_1^5(t) = e^{-16\lambda_1 t}(5 - 4e^{-4\lambda_1 t}). \quad (3.3)$$

Позначимо $\lambda^* = 8\lambda_1$. При цьому вирази (3.2) та (3.3) можна записати у вигляді:

$$P_{MA}(t) = e^{-2\lambda^* t}(3 - 2e^{-\lambda^* t}). \quad (3.4)$$

$$P_{MA}^{(1)}(t) = e^{-2\lambda^* t}(5 - 4e^{-0,5\lambda^* t}). \quad (3.5)$$

У таблиці 3.1 представлено значення ймовірності безвідмовної роботи для нерезервованого однобайтового позиційного (I). Також відповідно до виразів (3.4) та (3.5) розраховані значення ймовірності безвідмовної роботи для тройованого позиційного (II), для модулів з однією контрольною основою (III) та для модулів з двома контрольними основами (IV).

Відповідно до даних таблиці 3.1 на рис. 3.2 представлені графіки залежностей для однобайтового: нерезервованого (I), триканального резервованого (II) , (III) з параметрами l = 1, n = 2, k = 1 та графік залежності для k = 2 (IV). З даного графіка видно, модулі з двома контрольними основами надійніше тройорованного позиційного (II) і надійніше з однією контрольною основою (III).

У таблиці 3.3 представлені, наведені до одного двійкового розряду, деякі дані про додаткову кількість обладнання ΔA, необхідного для реалізації розглянутих методів підвищення надійності, які показують, що застосування кодів, що забезпечує високе значення ймовірності безвідмовної роботи при меншій кількості додатково введеного обладнання.

Таблиця 3.3 – Розрахункові дані надійності модулів IMA

$\lambda^{*}t$				
	I	II	III	IV
0	1	1	1	1
0,2	0,819	0,912	0,926	0,986
0,4	0,670	0,746	0,775	0,922
0,6	0,549	0,573	0,613	0,816
0,8	0,449	0,424	0,468	0,687
1	0,368	0,306	0,328	0,558
1,2	0,301	0,225	0,254	0,439
1,4	0,247	0,151	0,187	0,337

Ми розділили систему на два етапи:

– робота у віртуальному середовищі Linux. Linux необхідний, оскільки для взаємодії з керованою системою (симулятор X-Plane) потрібен набір TCP/IP. Цей розділ взаємодіє з симулятором X-Plane через протокол UDP і обмінюється даними з іншими розділами через порти XtratuM;

– працює на віртуалізованому Partikle. Йому виділяється додаток керування та наведення.

Це здійснюється за допомогою портів вибірки:

– інтерфейс в основному складається з двох потоків. Один потік отримує дані від симулятора, витягує числові значення з послідовності байтів і передає ці значення в інший розділ. Інший потік отримує керуючі дії від іншого розділу, пакує повідомлення у форматі симулятора та надсилає це повідомлення через UDP. Складається з двох потоків: керівного та керуючого. Такий розподіл потоків було зроблено на основі того, що завдання стенду є досить незалежними, перш за все тому, що вони вимагають різної частоти виконання.

Проблеми в реальному часі в основному пов'язані з дуже мінливою частотою, з якою X-Plane надсилає дані. Середня частота трохи нижче запрограмованого значення. З точки зору роботи в реальному часі, система повинна гарантувати, що керуючі дії надсилаються вчасно, незалежно від частоти, з якою надходять вхідні дані. Прийняте рішення полягає в тому, щоб зробити два процеси (прийом і відправлення) незалежними, використовуючи окремі потоки. Перший потік реалізує прийом UDP від симулятора X-Plane. Це аперіодичний потік, який запускається при надходженні вхідних даних. Другий потік це періодичний потік, реалізований за допомогою POSIX-таймерів, який надсилає керуючі дії симулятору X-Plane через UDP. Модулі з однією контрольною основою (III) більш надійна, ніж тройована система (II).

Отже, МА розширює область значень (λ*t) при яких існує виграш порівняно з нерезервованим позиційним (I) у безвідмовності [9].

Нехай $k = 2$. У цьому випадку можна подати у вигляді набору наступних основ: $m_1 = 3, m_2 = 4, m_3 = 5, m_4 = 7, m_5 = 11, m_6 = 13$.

Для даного двобайтового (l=2) вираз (3.5) запишемо у такому вигляді:

$$P_{MA}^{(2)}(t) = P_1^4(t)\left\{P_1^2(t) + 6P_1(t)\left[1 - P_1(t)\right] + 15\left[1 - P_1(t)\right]^2\right\}, \tag{3.6}$$

або

$$P_{MA}^{(2)}(t) = e^{-2\lambda^* t}\left[e^{-\lambda^* t} + 6e^{-0,5\lambda^* t}(1 - e^{-0,5\lambda^* t}) + 15(1 - e^{-0,5\lambda^* t})^2\right]. \tag{3.7}$$

Таким чином, зі збільшенням кратності резервування надійність модулів підвищується, що відповідає загальній теорії надійності

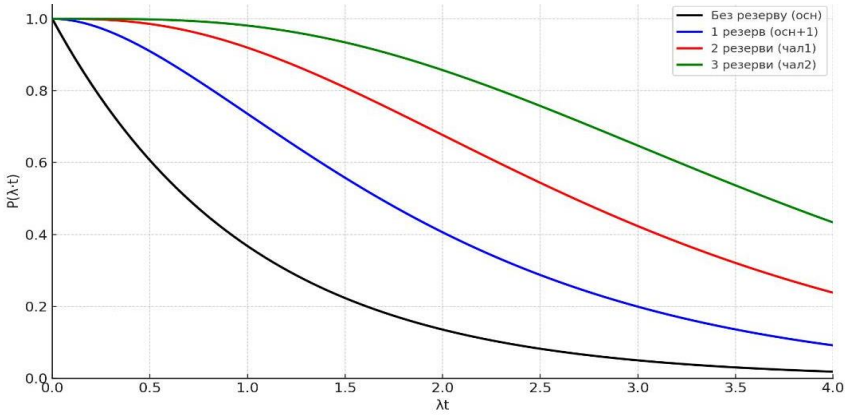


Рисунок 3.15 – Графіки залежностей ймовірності $P(\lambda * t)$ безвідмовної роботи від часу при різній кратності резервування

Згенерований код Simulink потребує адаптації до цільового середовища. Цей процес включає створення портів для вибірки, операцій читання та запису, а також генерацію допоміжних файлів із визначеннями констант. Окрему увагу приділяють реалізації механізмів використання потоків і семафорів згідно зі стандартом POSIX.

Адаптація здійснюється з використанням компілятора цільової мови програмування, що дозволяє налаштувати згенерований код відповідно до вимог цільової платформи. Такий підхід є надзвичайно ефективним під час розробки програмного забезпечення для систем інтегрованої модульної авіоніки (IMA), де критично важливими є сумісність, продуктивність і відмовостійкість.

3.4. Висновки до третього розділу

Розроблено загальний алгоритм побудови структури функцій КБО, який дозволяє автоматизувати даний процес при проєктуванні КБО в форматі IMA і вирішити оптимізаційну задачу структуризації та приватні алгоритми мінімізації кількості функцій в структурі і формування груп функцій.

Проведено математичний опис структури функцій, алгоритми її побудови та оптимізації є основою для побудови засобів САПР КБО IMA на

етапі проєктування даної структури.

Отже, сформовані різні підходи до побудови засобів тестування. Тести апаратури бортового обладнання 4-го покоління не підходять для контролю обчислювальних машин 5-го покоління у зв'язку з істотною відмінністю в організації архітектури обчислювальних систем і збільшення вимог по надійності і відмовостійкості.

Тестування апаратури бортового обладнання 5-го покоління слід організовувати з урахуванням нових підходів до архітектури обчислювальної системи. Конструктивно-функціональні модулі, об'єднані в єдину локальну мережу, слід тестувати шляхом зовнішнього функціонального контролю з введенням процедури мажорування. Причому, проведення тесту ініціюється стороннім модулем, що входять до складу обчислювальної системи.

Модульна побудови тестів конструктивно-функціональних модулів дозволяє скласти набір тестів, з яких можна скласти функціональний тест для будь-якого з типів обчислювальних модулів.

Для безпеки критично важливих систем необхідно використовувати відмовостійкість, щоб допускати помилки проектування, які переважно пов’язані з програмним забезпеченням і синхронізацією. Недостатньо усунути майже всі несправності, які виникли на пізніх стадіях життєвого циклу; потрібна впевненість, що вони були усунені або є вкрай малоймовірними.

РОЗДІЛ 4. ЕФЕКТИВНОСТЬ ТА ПІДВИЩЕННЯ НАДІЙНОСТІ ТЕХНІЧНОЇ ЕКСПЛУАТАЦІЇ ІНФОРМАЦІЙНОЇ ПІДТРИМКИ АВТОМАТИЗОВАНОГО КОНТРОЛЮ БРЕО.

Надійність технічної експлуатації інформаційної підтримки автоматичного контролю БРЕО є одним з визначальних чинників забезпечення безпеки та регулярності польотів у цивільній та військовій авіації. У процесі експлуатації радіоелектронного обладнання його надійність неминуче погіршуються з плином часу. Погіршення надійності пов’язане зі збільшення кількості відмов, а отже, зі зменшенням середнього напрацювання на відмову та коефіцієнта готовності [**Error! Reference source not found.**, **Error! Reference source not found.**]. Тому в цьому розділі проаналізовані моделі зміни показників надійності у випадку погіршення технічного стану обладнання на прикладі інтенсивності відмов.

Своєчасне виявлення змін у трендах реалізацій оцінок показників надійності є основою для визначення залишкового ресурсу **Error! Reference source not found.**. Тому метою цього розділу є синтез та аналіз методів та процедур виявлення погіршення технічного стану на основі статистичного оброблення даних щодо показників надійності, що є складовою частиною запропонованої методології оброблення даних.

Для досягнення поставленої мети в цьому розділі були вирішені такі задачі: розробка системи підвищення відмовостійкості БРЕО на основі теорії функціональних систем; виконано синтез та аналіз евристичної процедури виявлення погіршення технічного стану на основі оцінювання параметрів моделі змін показників надійності; виконано синтез та аналіз евристичної процедури виявлення погіршення технічного стану на основі апроксимації емпіричних даних для стрибкоподібної моделі; виконано синтез та аналіз послідовної процедури виявлення погіршення технічного стану; отримано оцінки параметрів погіршення технічного стану на основі методу максимальної правдоподібності для стрибкоподібної, лінійної та квадратичної моделей.

4.1. Розробка системи підвищення відмовостійкості БРЕО на основі теорії функціональних систем

В розділі описано синтез системи діагностики працездатності вимірювальних приладів ПС. Розглянуто технології розробки діагностичної системи на основі моделі похибок приладів, які використовуються для вимірювання параметрів ПС.

Вимоги по надійності точності, що пред'являються до вимірювальних систем ПС, постійно зростають. Для забезпечення необхідних характеристик вимірювальних систем і приладів необхідно мати інформацію про їх стан, точностних характеристиках і перспективах зміни цих параметрів в майбутньому. Зазначені завдання вирішуються за допомогою різних систем діагностики і контролю [2, 49, 43]. Синтез системи діагностики доцільно здійснювати з використанням підходу алгоритмічного конструювання, що дозволяє досягти необхідного результату за короткий час з мінімальними фінансовими витратами.

Реалізація систем контролю і діагностики ПС здійснюється в бортовий цифровий обчислювальної машині. Тобто, створення прототипу діагностичної системи визначення та прогнозування стану вимірювальної техніки ПС.

Життєздатність та розвиток інформаційних технологій пояснюється тим, що сучасний ПС вкрай чутливий до помилок в управлінні. Помилки і збої в роботі алгоритмічного забезпечення знижують якість функціонування систем управління і можуть привести до відмов і фатальним результатам.

Тому надзвичайно важливо зрозуміти, які вимірювальні прилади вийшли з ладу, рівень їх точності і перспективи зміни точностних характеристик. Ця інформація дозволяє прийняти рішення про способи використання сигналів вимірювальних приладів в системі управління ПС.

Система автоматичного діагностування являє собою комплекс програмних і апаратурних засобів і є складовою частиною ДЕС.

Розглянемо синтез систем діагностики на прикладі задачі оцінки і прогнозування похибок БРЕО на борту ПС.

Методи оцінки і прогнозування інструментальних похибок БРЕО.

Обурення викликаючі кутову швидкість догляду Ω БРЕО, яку можна розділити на швидкість догляду Ωн, не залежну від прискорення, і швидкість догляду Ωр, яка залежить від прискорення і пропорційна розбалансування. Ωр складається з швидкості догляду $\Omega_{рI}$ і $\Omega_{рII}$, пропорційні осьової і радіальної розбалансуванням відповідно [61]. Найбільший вплив на стан обладнання надає складова $\Omega_{рI}$.

Тому для оцінки та прогнозування інструментальних похибок БРЕО можна використовувати, наприклад, оцінку і прогнозування працездатності апаратного забезпечення, яка залежить від швидкості догляду $\Omega_{рI}$.

Даний метод полягає в визначенні швидкості догляду $\Omega_{рI}$ за інформацією, яка вимірювалася в попередніх випробуваннях і по вимірах при запуску приладів для введення поправок. Далі проводиться прогноз величини стабільної складової $\Omega_{рI}$. Прогноз здійснюється на наступний запуск приладів і обчислюється оцінка відповідності його допуску.

На інтервалі польоту проводиться визначення та прогноз точностних характеристик приладів. Розрахунок змінної складової $\Omega_{рI}$ проводиться в заданому допуску. Останні два пункти дозволяють прийняти рішення про можливість постановки приладів на борт ПС [78].

У практичних додатках проводиться рішення задачі оцінки і прогнозу інструментальних похибок приладів паралельно: визначається стабільна складова швидкості догляду $\Omega_{рI}$ при запуску, яка використовується в алгоритмах управління для формування поправок; обчислюється оцінка і прогноз характеристик приладів на інтервалі польоту ПС [99].

Модель швидкості догляду приладів $\Omega_{рI}$ має вигляд:

$$\Omega_{M_t} = \overset{з}{\ell}(\Omega_t^з) + \alpha_\Delta(\Delta\Omega_t^з) + \overset{с}{\ell}(\Omega_t^с), \qquad (4.1)$$

де Ω_{Mt} – швидкість догляду приладів $\Omega_р$; $\Omega^з_t$, $\Delta\Omega^з_t$ – стабільна і змінна

складові при конкретному запуску приладів; Ω^{c}_{t} – значення «стрибка» стабільної складової швидкості догляду приладів при дослідженні на різних запусках; $\overset{з}{\ell}, \alpha_{\Delta}, \overset{c}{\ell}$ – параметри моделі.

Введення поправок зазвичай здійснюється на $\Omega_{з}$, тому $\Omega_{M_t} = \overset{з}{\ell}(\Omega^{з}_{t})$, якість здійснення корекції за допомогою стабільної складової швидкості характеризується ступенем наближення розрахункової швидкості догляду до е фактичному значенню, тобто

$$/\Omega^{з}_{t} - \Omega_{M_t} / \le \varepsilon_{\Omega} \quad \text{при} \quad t_{\text{вим}} \le t_{\text{доп}}, \tag{4.2}$$

де ε_{Ω} – точність оцінки стабільної складової швидкості виходу на інтервалі е вимірювання $t_{вим}$, що не перевищує допустимого значення $t_{доп}$.

В умовах вдало сформованої похибка оцінки Ω_{Mt} залежить тільки від точності визначення її параметрів.

Важливою характеристикою при оцінці параметрів моделі є їх стабільність в сенсі наступних критеріїв:

$$K_{I} = \left| M\left[\ell^{з}\right] - \ell^{з}_{i} \right| \le \Delta; \quad K_{II} = \sqrt{\frac{\sum_{i=1}^{n} (\overset{з}{\ell}_{i} - \mathrm{M}\left[\overset{з}{\ell}_{i}\right])^{2}}{n-1}} - M \le \eta_{\ell}; \tag{4.3}$$

$$K_{III} = \sqrt{\frac{\sum_{i=1}^{n} (\overset{з}{\ell}_{i} - \overset{з}{\ell}_{i-1})^{2}}{n}} \le \varepsilon_{\ell},$$

де $M[\overset{з}{\ell}]$ – математичне очікування; Δ, η_{ℓ}, ε_{ℓ} – задані точності визначення параметрів моделі. Ці параметри визначають виходячи з умови забезпечення мінімального впливу інструментальних похибок БРЕО на точність переміщення ПС в задану область фазового простору; i = 1,2, ..., n - послідовність оцінок параметрів моделі.

Критерій K_{III} є чутливий до зміни параметрів моделі (4.1), яка використовується для оцінки і прогнозу працездатності приладів БОІС. Процес зміни «стрибка» характеризує член $\Omega_{M_t} = \overset{c}{\ell}(\Omega^{c}_{t})$.

Другий член (4.1) характеризує зміну змінної складової швидкості

догляду приладів. При цьому час знаходження $\Delta\Omega^{з}_{t}$ в межах допуску можна обчислити за допомогою змішаного розподілу Вейбулла [104].

$$f(t)=\left\{\frac{\beta_\Omega}{\alpha_\Omega}\left(e-\gamma_\Omega\right)^{\beta_\Omega-1}\exp\left[-\frac{(t-\gamma_\Omega)^{\beta_\Omega}}{\alpha_\Omega}\right]\right\};\ t\geq\gamma_\Omega,\beta_\Omega>0,\alpha_\Omega>0, \qquad (4.4)$$

0 – в інших випадках,

де α_Ω, β_Ω – параметри масштабу і форми, що характеризують, відповідно, час знаходження $\Delta\Omega^{з}_{t}$ в межах допуску і ступінь зносу приладів; γ_Ω – параметр положення, що визначає кінцевий час, до якого вихід $\Delta\Omega^{з}_{t}$ за допуск неможливий.

При $\gamma_\Omega\approx0$ ймовірність $P(t)$ і середній час μ_t знаходження $\Delta\Omega^{з}_{t}$ в заданому допуску обчислюються наступним чином

$$P(t)=\exp\left[-\frac{t^{\beta\alpha}}{\alpha_\Omega}\right];\ \mu t=(\alpha_\Omega)\frac{1}{\beta_\Omega}-\left(1+\frac{1}{\beta_\Omega}\right), \qquad (4.5)$$

де $\Gamma(o)$ – гамма функція.

Зміни швидкості догляду в запусках $\Omega_{M_t}=\overset{з}{\ell}(\Omega^{з}_{t})$ описаний авторегресійною моделлю:

$$\Omega^{з}_{k}=(1+\ell^{з}_{1})\Omega_{k-1}+\ell^{з}_{1}\Omega_{k-2}, \qquad (4.6)$$

а процес зміни «стрибків» стабільної складової швидкості догляду $\Omega_{M_t}=\ell^{c}(\Omega^{c}_{t})$ моделлю

$$\Omega^{c}_{k}=\ell^{c}_{1}\Omega_{k-1}, \qquad (4.7)$$

Параметри моделей визначаються в наступному вигляді

$$\ell^{з}_{t}=\frac{\sum_{i=2}^{k}\omega_i\omega_{i-1}}{\sum_{i=2}^{k}\omega^{2}_{i-1}};\ \ \ell^{c}_{1}=\frac{\sum_{i=2}^{k}\Omega_i\Omega_{i-1}}{\sum_{i=2}^{k}\Omega^{2}_{i-1}}, \qquad (4.8)$$

де $\omega_k=\Omega_k-\Omega_{k-1}$ – різниця першого порядку.

Параметри Вейбулла в виразі (4.8) визначимо методом максимальної правдоподібності. Формалізовані залежності, які використані для оцінки

параметрів α_Ω і β_Ω мають такий вигляд: $n\alpha_\Omega - \sum_{i=1}^{n} t_i^{\beta_\Omega} = 0;$

$$\frac{n}{\beta_\Omega} = \sum_{i=1}^{n} \ln t_i - \frac{1}{\alpha_\Omega} \sum_{i=1}^{n} t_i^{\beta_\Omega} \ln t_i = 0, \qquad (4.9)$$

де n – кількість інформації про час, коли прилад знаходиться в межах робочого допуску; t_i – час знаходження в допуску за результатами випробувань приладу ($i = 1, 2, ..., n$).

Рішення рівнянь проводиться з використанням методу випадкового пошуку. Знаходимо мінімальну нев'язку $\Delta E = /E_1/ - /E_2/,$ (4.10)

де E_1, E_2 – є, відповідно, першим і другим рівняннями системи .

На кожному кроці пошуку значень параметрів α_Ω і β_Ω приймаються величини $\bar{h}^{(N)} = \ell \xi^{(N)},$ (4.11)

де $\bar{h} = (\alpha_\Omega, \beta_\Omega)^N$; N – кількість пошукових кроків; ℓ – масштаб можливих меж змін $\bar{h}$; ξ – нормальний розподіл.

Якщо умова $\Delta E < \varepsilon_2$, де ε_2 – задана точність рішення, виконується, то пошук завершується. Отримані значення α_Ω і β_Ω дозволяють визначити показники (30). Розглянутий метод за допомогою (4.3) і (4.4) дозволяє оцінити і спрогнозувати стан приладів в складі БЦОМ. Необхідно виконання умов:

$$\Omega_k^c \le \Omega_{\text{доп}}^c; \;\; P(t) \ge P_{\text{зад}}; \;\; \mu_t \ge t_{\text{пол}}, \qquad (4.12)$$

де $\Omega_{\text{доп}}^c$ – допустиме значення «стрибка» $\Omega_{\text{PI;}}$ $P_{\text{зад}}$ – задана ймовірність знаходження змінної складової швидкості догляду в діапазоні на інтервалі польоту ПС $t_{\text{пол.}}$

В процесі випробувань приладів уточнюються параметри Вейбулла за наступним алгоритмом

$$\alpha_\Omega = \alpha_\Omega^{\text{ср}} - \frac{1}{n+K}(\alpha_\Omega^{\text{ср}} - \alpha_T); \quad \beta_\Omega = \beta_\Omega^{\text{ср}} - \frac{1}{n+K}(\beta_\Omega^{\text{ср}} - \beta_T), \qquad (4.13)$$

де K, α_T, β_T – визначаються в результаті випробувань кількість запусків і параметри.

Після уточнення α_Ω і β_Ω розраховуються $P(t)$, μ_t і відповідно до виразом (4.13) за результатами випробувань приладів приймається рішення про можливість його установки на борт ПС.

Оцінку $\Omega_t^з$ вводять в алгоритми управління за допомогою моделі (4.1) при значенні параметра, визначеному за результатами заводських випробувань гприладів. Це дозволяє істотно скоротити необхідний інтервал для оцінки і прогнозування швидкості догляду приладів.

Прогнозування стану БРЕО ПС здійснюється на основі апріорної математичної моделі. На сучасному етапі з'явилася можливість використовувати на борту ПС еволюційні алгоритми побудови прогнозуючих моделей, які базуються на методах самоорганізації, генетичних підходах і нейронних мережах [31, 25, 45]. Також широко використовуються методи, засновані на ідеях ідентифікації параметрів моделі з заданою структурою [44].

Застосування апріорних моделей виду (4.1) відрізняється простотою і досить високою точністю, тому що базуються на великому статистичному матеріалі.

В алгоритмах прогнозу, заснованих на ідентифікації параметрів моделі з апріорно заданою структурою, основна складність полягає в попередньому виборі адекватної апріорної інформації про структуру моделі.

Алгоритми самоорганізації - це багаторядні алгоритми, що базуються на гіпотезі селекції. Необхідно вирішити систему нормалізованих рівнянь Гаусса:

$$\begin{cases} \sum\limits_{i=1}^{N} y_i = \alpha_0 N + \alpha_1 \sum\limits_{i=1}^{N} g_{0,j}(x_i) + \alpha_2 \sum\limits_{i=1}^{N} g_{1,j}(x_i); \\ \sum\limits_{i=1}^{N} y_i g_{0,j}(x_i) = \alpha_0 \sum\limits_{i=1}^{N} g_{0,j}(x_i) + \alpha_1 \sum\limits_{i=1}^{N} g_{0,j}^2(x_i) + \alpha_2 \sum\limits_{i=1}^{N} g_{0,j}(x_i) g_{1,j}(x_i); \\ \sum\limits_{i=1}^{N} y_i g_{1,j}(x_i) = \alpha_0 \sum\limits_{i=1}^{N} g_{1,j}(x_i) + \alpha_1 \sum\limits_{i=1}^{N} g_{0,j}(x_i) g_{1,j}(x_i) + \alpha_2 \sum\limits_{i=1}^{N} g_{1,j}^2(x_i). \end{cases} \quad (4.14)$$

При введенні в базис константи від вільного члена в рівняннях можна відмовитися:

$$\begin{cases} y_i = \alpha_{0,j} g_{0,j}(x_i) + \alpha_1 g_{1,j}(x_i),\ j = 1 \ldots C_N^2. \\ \sum_{i=1}^{N} y_i g_{0,j}(x_i) = \alpha_0 \sum_{i=1}^{N} g_{0,j}^2(x_i) + \alpha_1 \sum_{i=1}^{N} g_{0,j}(x_i) g_{1,j}(x_i) \\ \sum_{i=1}^{N} y_i g_{1,j}(x_i) = \alpha_0 \sum_{i=1}^{N} g_{0,j}(x_i) g_{1,j}(x_i) + \alpha_1 \sum_{i=1}^{N} g_{1,j}^2(x_i). \end{cases} \quad (4.15)$$

Обчислення припиняються, коли досягається мінімум використовуваного групи критеріїв. Результатом є краща модель в останньому ряду. Функціональна схема алгоритму МГУА представлена на рисунку 3.14.

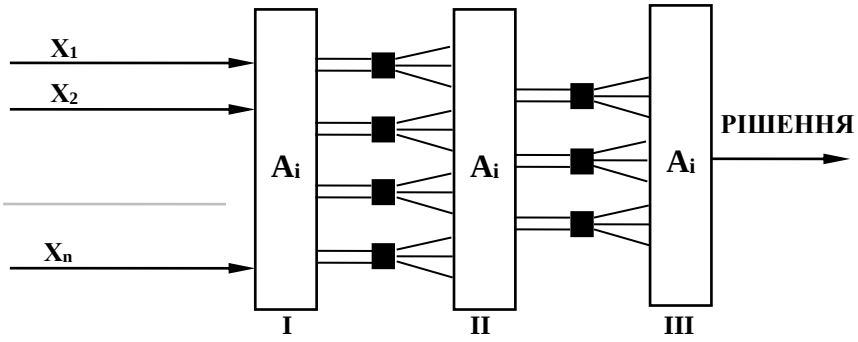


Рис. 4.1. Функціональна схема модуля головного управління авіонікою (МГУА)

де: A_i – порогові самовідбори корисної інформації; *I, II, III* – критерії самовідбору. Отримана модель використовується для прогнозу і подальшої корекції в вихідному сигналі ІНС БЦОМ ПС.

Еволюційні алгоритми реалізують механізм природного відбору. В процесі функціонування алгоритму пристосованість моделей-претендентів в цілому зростає. Еволюційні алгоритми застосовуються для вирішення завдань, що мають багато локальних мінімумів, а їх недоліками, при використанні в системах діагностики на борту ПС, є підвищені вимоги до продуктивності обчислювальних засобів при реалізації на борту ПС, а також необхідність обліку особливостей проектування алгоритмів (вдалий вибір групи критеріїв селекції алгоритму самоорганізації [50], вибір функції пристосованості генетичного алгоритму).

Таким чином, розроблений підхід до задачі синтезу системи діагностики вимірювальної апаратури ПС. Технологія синтезу продемонстрована на прикладі створення автоматичної системи діагностики приладів ВК перспективного ПС. Для оцінювання та прогнозування стану приладів використовуються апріорні моделі, алгоритми ідентифікації та еволюційні алгоритми побудови прогнозуючих моделей.

При використанні подібної структури в системі управління необхідно враховувати деякі особливості рисунок 4.2. Отже, необхідність постійно мати адекватну модель, її доводиться будувати заново або коригувати наявну модель при виникненні суттєвих змін зовнішнього і внутрішнього середовища.

Така необхідність пояснюється вимогами швидко реагувати на зміни, зумовлені постійно мінливих умов функціонування. Остання особливість призводить до необхідності тримати ситуацію під постійним контролем, використовувати постійну корекцію і не дозволяє застосовувати комплекс довгострокових заходів, використовувати ефективні сценарії.

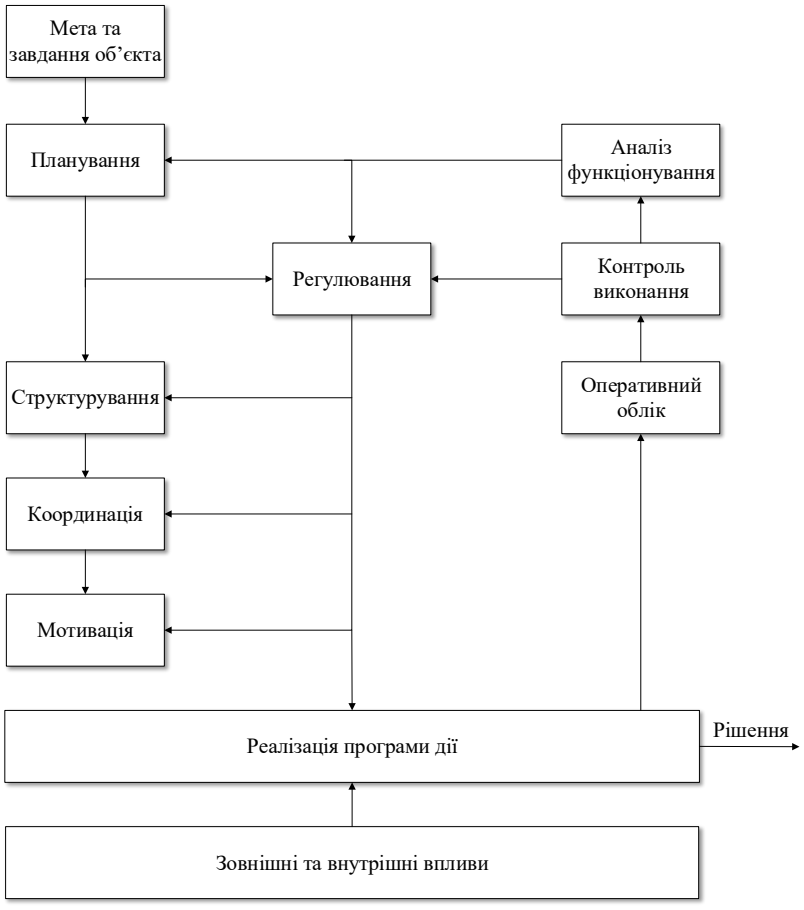


Рис. 4.2 Функціональна схема циклу управління БРЕО

При функціонуванні системи, можливо, цілеспрямовано змінювати її стан за допомогою підсистем управління.

Управління динамічним об'єктом здійснюється з будь-якою метою, яка реалізується за допомогою групи різних критеріїв або в формалізованому вигляді як функціонал якості. Шляхом мінімізації або максимізації функціоналу якості визначаються оптимальні параметри об'єкта.

Функціональні системи реалізують принцип саморегуляції [5]. Вони вибірково об'єднують підсистеми і їх елементи з метою досягнення корисних результатів. У функціональній системі присутня зворотна інформація про результати її дії, що забезпечує пристосувальний ефект, який призводить до досягнення мети, яка є системоутворюючим фактором.

Механізми корекції параметрів системи необхідно синтезувати з урахуванням можливостей реалізації саморегуляції.

У колі зворотної аферентації основним алгоритмом є алгоритм побудови прогнозуючих моделей, заснований на методі самоорганізації.

Прогнозуюча модель має вигляд:

$$\Phi(x) = \sum_{i=1}^{N} a_i \, \mu_{ni}(fix), \tag{4.16}$$

де n – число базисних функцій; μ_n – базисні функції з параметризовані множини F_p. $F_p = \{a_i \mu_i(f_i x) \mid i = 1, L\{\}\}$, набір базисних функцій. Кожній базисній функції ставиться у відповідність двомірний вектор параметрів $(a, f)^T$, де a – амплітуда, f – частота, доцільно здійснювати на основі аналізу простих прогнозуючих моделей.

Використання модифікацій алгоритму самоорганізації дозволяє підвищити точність прогнозу, а отже, і точність діагностики приладового комплексу ПС. Таким чином, розроблена перспективна система діагностики (управління) приладового комплексу ПС, яка дозволяє здійснювати діагностику протягом усього його життєвого циклу.

4.2 Методика прогнозування показників надійності БРЕО за даними експлуатаційної підтримки

Для прогнозування кількості відмов за період експлуатації розглянуто три підходи. Перший призначений для прогнозування відмов відновлюваних елементів за середнім значенням параметра потоку відмов елемента сумарного напрацювання за контрольний період, заснований на використанні закону Пуассона. Другий для прогнозування відмов виробів, що не відновлюються, на основі експлуатації λ характеристик і розрахунку ймовірності безвідмовної роботи за прогнозований період. Третій на основі експлуатації функції розподілу стерених відмов.

Контроль та прогнозування кількості відмов відновлюваних елементів за розподілом Пуассона

1) За статистичними даними про усунення в експлуатації відмови підконтрольного елемента (вузла), отриманий для сукупності всіх модулів, що експлуатуються, визначається середнє значення параметра потоку відмов за формулою:

$$\overline{\omega}_{\text{р}} = \frac{1}{k}\sum_{j=1}^{k}(\omega_{\text{р}})_j, \qquad (4.16)$$

Де $(\omega_{\text{р}})$- параметр потоку відмов елемента за j - й період експлуатації (місяць, квартал, рік), що передує контрольованому, який має номер k.

Допустима кількість відмов елемента n_{dk} за контрольований період виходить із рішення рівняння: $P = (n_{dk}, \overline{\omega}_{\text{р}}, t_{sk}) = P_d$ (4.17)

яке при розподілі числа відмов за законом Пуассона має вигляд

$$P = \frac{\sum_{n=0}^{n_{dk}} (\overline{\omega}_{\text{р}} t_{dk})^n}{n!} exp(-\overline{\omega}_{\text{р}} t_{dk}) \qquad (4.18)$$

де $P(\cdot)$ – закон розподілу ймовірності безвідмовної роботи; $t_{dk} = m\sum T_{i.k}$ (при $i = \overline{l,m}$)- сумарне напрацювання елемента за контрольований період; N – кількість модулів біля парку ПС, що пройшли підконтрольну експлуатацію; t_{ik} – напрацювання i-го модуля за контрольований період; m – кількість однотипних елементів на двигуні;

P_d – допустима ймовірність безвідмовної роботи елемента, яку на основі аналізу зарубіжного досвіду можна приймати рівною $P_d = 0{,}975$.

Якщо фактична кількість відмов n_{fk} за контрольований період перевищує отриману рішення рівняння (4.19) величину n_{dk}, тобто

$$n_{fk} > n_{dk}\,, \tag{4.19}$$

то при подальшій експлуатації даного елемента ВМД приймається рішення щодо розробки заходів, що запобігають зниженню рівня надійності.

Покладемо, що число відмов елемента n_{dw} за прогнозований період експлуатації визначаємо шляхом вирішення рівняння типу (4.20)

$$P_d = \frac{\sum_{n=0}^{n_{d\cdot p}} (\bar{\omega}_b t_{dw})^n}{n!}\, exp(-\bar{\omega}_b t_{dw})\,, \tag{4.20}$$

де t_{dw} – передбачуване сумарне напрацювання елемента за період прогнозу, яке приблизно можна оцінити за допомогою співвідношення

$$t_{dw} \approx t_{dk}\left(1 + \frac{N_1 - N_2}{N}\right) \tag{4.21}$$

де N_1 – кількість модулів, що проходять експлуатацію в прогнозований період; N_2 – число модулів, що підлягають списуванню в той же період.

Число запасних елементів для сукупності модулів $(N + N_1 - N_2)$ можна у першому наближенні прийняти рівним величині n_{dw}.

Отже, прогнозування показників надійності невідновлюваних елементів на основі екстраполяції λ характеристики.

Для кожного попередньо обґрунтованого інтервалу напрацювання модулів Δt_i, що починається зі значення t_i, визначається сумарний потік відмов підконтрольного елемента за формулою

$$\omega_s(t_i) = \frac{n(\Delta t_i)}{N_{wk}(t_i)\Delta t_i} \tag{4.22}$$

де $n(\Delta t_i)$ кількість елементів, що відмовили в інтервалі Δt_i;

$N_{wk}(t_i)$ - число справних до початку цього інтервалу елементів.

З усіх попередніх моменів контролю напрацювань елемента t_i вибирається не менше трьох мінімальних значень потоку відмов , $\omega_{s\,min}(t_i)$, за яких оцінюється середня величина інтенсивності раптових відмов за

допомогою співвідношення: $\lambda_v = \frac{1}{3}\sum_{j=1}^{3} \omega_s(t_i)$ (4.23)

і знаходяться значення інтенсивності поступових відмов

$$\lambda(t_i) = \omega_s(t_i) - \lambda_v \geq 0 \quad (4.24)$$

За вибіркою значень $\lambda(t_i)$ методом найменших квадратів, виконується адекватна апроксимація λ характеристики елемента за допомогою спеціально підібраної функції $\lambda(t)$.

На основі теореми множення ймовірностей, що застосовується по відношенню до раптових та поступових відмов для прогнозованого інтервалу напрацювання $\Delta t_w = t_{wt}$, визначається ймовірність безвідмовної роботи елементу.

$$P_w = exp(-\lambda_v \Delta t_w)\, exp\left(-\int_{t_k}^{t_w} \tilde{\lambda}(t)dt\right) \quad (4.25)$$

де t_k – напрацювання елемента на момент контролю; t_w - прогнозоване напрацювання елементу.

За умови, що ймовірність безвідмовної роботи прогнозованої за формулою (51) не нижче допустимої величини $(P_w \geq P_d)$, кількість запасних елементів на період прогнозу перебуває як кількість їх можливих відмов у цей період.

$$n_s = n = (1 - P_w)N_{wk}(t_k), \quad (4.26)$$

де $N_{wk}(t_k)$ - кількість справних елементів парку модулів в момент контролюрівня надійності.

Прогнозування показників надійності невідновлюваних елементів на основі екстраполяції функції розподілу поступових відмов.

Виконується перевірка відповідності розподілу статистичної сукупності напрацювання підконтрольного елемента до виникнення поступових відмов щодо двопараметричному закону Вейбулла за допомогою критерію[147] $$S = \left\{\frac{\sum_{n=\frac{r}{2}-1}^{r-1} \frac{t_{n+1}-t_n}{M_n}}{\sum_{n=1}^{r-1} \frac{t_{n+1}-t_n}{M_n}}\right\} < S_k \quad (4.27)$$

де r – сумарна кількість поступових відмов елемента модулів БРЕО, зареєстрованих на момент контролю; M_n, S_k – величини, виникнення визначених за таблицею, наведеною у роботі [147] .

При виконанні нерівності (4.27) гіпотеза про застосування розподілу Вейбулла не відкидається при 95%-му рівні значимості.

Послідовно для обумовленого напрацювання елемента t_i; відомий значенням інтенсивності відмов, що визначаються з виразу (4.24), вирахуються статистичні оцінки ймовірностей безвідмовної роботи

$$P(t_i) = exp\left\{-\sum_{j=1}^{i} \frac{\lambda(t_j)+\lambda(t_{j-1})}{2} - (t_j - t_{j-1})\right\} \quad (4.28)$$

де слідує приймати $\lambda(t_0) = 0$.

Приймаючи для апроксимації розподілу поступових відмов двопараметричний закон Вейбулла, методом найменших квадратів обчислюються парамитри цього закону b і θ за співвідношенням

$$b = \frac{\left(\sum_{j=1}^{m} X_j\right)^2 - m\sum_{j=1}^{m} X_j^2}{m\sum_{j=1}^{m} X_j Y_j - \sum_{j=1}^{m} X_j \sum_{j=1}^{m} X_j}, \quad (4.29)$$

$$ln\theta = \frac{1}{m}\left(\sum_{j=1}^{m} Y_j - \frac{1}{b}\sum_{j=1}^{m} X_j\right), \quad (4.30)$$

де $Xj = ln\{ln[1/P(t_j)]\}$, $Y = lnt_j$; m – число вибраних інтервалів напрацювання на момент контролю рівня надійності.

Таблиця 4.1

Значення коефіцієнтів M_n і S_{kr} із значення коефіцієнта α розрахункових параметрів M_n та ρ залежно від висоти h.

n\r	3	4	5	6	7	8
1	1.2164	1.1507	1.1157	1.1039	1.1007	1.1082
2	0.8630	0.7067	0.6454	0.6123	0.5916	0.5773
3		0.6796	0.5324	0.4743	0.4428	0.4229
4			0.5833	0.4429	0.3873	0.3570
5				0.5228	0.3877	0.3341
6					0.4806	0.3499
7						0.4493
S_k	0.95	0.76	0.86	0.73	0.8	0.71

M_n

h	ρ					
	0.0	0.2	0.4	0.6	0.8	1.0
0.05	0.530	0.0606	0.0606	0.0700	0.0750	0.0790
0.10	0.110	0.1215	0.1336	0.1446	0.1542	0.1622
0.20	0.243	0.2700	0.2922	0.3122	0.3299	0.3455
0.30	0.402	0.4442	0.4766	0.5055	0.5311	0.5535
0.40	0.596	0.6488	0.6930	0.7322	0.7668	0.8000
0.50	0.837	0.9011	0.9557	1.0047	1.0503	1.1315

Для прогнозованого інтервалу напрацювання елемента $\Delta t_w = t_w - t_k$ визначається ймовірність його безвідмовної роботи за формулою

$$P_w = exp(-\lambda_b \Delta \bar{t}_w) exp\left(-\left(\frac{t_w - t_k}{\theta}\right)\right), \quad (4.31)$$

Число запасних елементів на період прогнозу знаходимо з виразу (4.26).

При невиконанні нерівності (4.27) перевіряється гіпотеза на можливість застосування нормального розподілу. Для чого спочатку визначаються параметри нормального закону: середнього відпрацювання на відказ $\tilde{T}$ і дисперсії σ за формулами:

$$\tilde{T} = \tilde{T_1} - \alpha(\tilde{T_1} - R); \quad (4.32)$$

$$\sigma^2 = \sigma_1^2 + \alpha(\tilde{T_1} - R^2),$$

де $$\tilde{T_1} = \frac{1}{n}\sum_{j-1}^{n} t_j \; ; \; \sigma_1^2 = \frac{1}{n-1}\sum_{j=1}^{n}(t_j - \tilde{T_1})^2, \quad (4.33)$$

$t_j (j = \overline{1,n})$ – поточні напрацювання до відмови; n – число відмов елементів.

Коефіцієнт α визначається за таблиці залежно від параметрів $\rho = \frac{\sigma_1^2}{(T_1 - R^2)}$ і $h = \frac{(M-n)}{M}$, де $M = mN$ – загальна кількість елементів: m – число однотипних елементів модуля; N – число модулів, які працювали за контрольний період; R – ресурс модуля. $\frac{X^2 - S}{\sqrt{S}} < 3$ (4.34)

де $S = k + 1 - 2$ –число степенів свободи; n – число відказів i-му діапазоні напрацювання;

$$X^2 = \sum_{i=1}^{k} \frac{(n_i - M_i q_i)^2}{M_i q_i}, \text{при } i = \overline{1,k} \text{ значення критерію } X^2;$$

$$q_1 = \frac{1}{\sqrt{2\pi\sigma}} \int_{t_i}^{t_{i-1}} exp\left(-\frac{t - \tilde{T}}{2\sigma^2}\right) dt$$

ймовірність відмови у i-му інтервалі; $M_i = mN_i$ – кількість елементів, працювавших в i-му інтервалі за контрольний період; m – кількість однотипних елементів на модулі N_i – кількість модулів, які працювали в i-му інтервалі за контрольний період.

Для прогнозованого інтервалу напрацювання знаходимо ймовірність безвідмовної роботи

$$P_w = exp(-\lambda_b \Delta t_w) exp\left(-\int_{t_k}^{t_w} \frac{1}{\sqrt{2\pi\sigma}} exp\left(-\frac{(t-\tilde{T})^2}{2\sigma^2}\right) * \left[1 - F\left(\frac{t-\tilde{T}}{\sigma}\right)\right]^{-1} dt\right) \quad (4.35)$$

Розрахунок числа запасних елементів здійснюється за формулою (4.18).

Проблема контролю ефективності заходів, вкладених у підвищенні надійності, базується на системному підході, що дозволяє оцінити вплив окремої чи групи заходів на ефективність роботи як конкретної ЕП (експлуатаційне підприємство, тобто авіаційна компанія), так і всієї галузі в цілому. Ефективність заходів оцінюється в першу чергу за їх впливом на показники роботи АТ(авіаційна техніка), безпеку та регулярність польотів, а також рівень надійності за допомогою аналізу таких характеристик як, λ-інтенсивність відмов, T_n напрацювання на відмову в польоті, P_n-коефіцієнт регулярності польотів, $\tilde{t}$ - середній час затримки рейсу з технічних причин.

У розробленій АСН (автоматизована система нагляду) продуктивність заходів пропонується оцінювати за допомогою коефіцієнтів:

$$K_{fij} = \frac{X_{oj} - X_{bj}}{X_{bj}}, \qquad K_{\tilde{b}ijk} = \frac{X_{oj} - X_{\tilde{b}jk}}{X_{\tilde{b}jk}} \quad (4.36)$$

де K_{fij} - коефіцієнт продуктивності його заходу за j-м показником надійності, X_{oj} – значення *j*-го показника за звітний період, X_{bj}- значення *j*-го показника у базовому періоді; $K_{\tilde{b}ijk}$ – коефіцієнт відхилення продуктивності *i*-го заходу за *j*-м показником надійності у звітному періоді від очікуваного значення, $X_{\tilde{b}jk}$ – значення прогнозованого *j*-го показника у звітному періоді на базі *k*-го базового періоду.

Використання АСН для збору даних про продуктивність заходів у різних ЕП дозволяє виявляти причини їх недостатньої ефективності, складати рекомендації щодо розробки конкретних заходів з урахуванням попереднього досвіду та специфіки льотно-технічної діяльності ЕП.

Якщо захід полягає у впровадженні нового методу діагностики, то його ефективність в АСН оцінюється коефіцієнтом виявлених відмов [K_v]:

$$K_v = \frac{N_{vj}}{N_j} \quad (4.37)$$

де N_j - загальна кількість відмов, які повинні бути виявлені за

допомогою даного способу діагностики; N_{vj} – кількість виявлених відмов. Для оцінки ефективності модифікованого методу діагностики використовується коефіцієнт

$$K_f = \frac{K_1 - K_2}{K_2}, \tag{4.38}$$

де K_1, K_2 – коефіцієнти виявлення відмов у звітному та базовому періодах. Для оцінки ефективності заходів також можна скористатися ймовірнісними критеріями. Зокрема, якщо для інтенсивності відмов на дострокове зняття двигуна за КПН (коефіцієнт придатності до надійності) λ_D – частота до проведення заходу становила λ_1, а після його проведення – λ_2, то для їх виправлення в АСН використовують критерій у вигляді:

$$u = \frac{2}{\sqrt{\frac{1}{t_1} + \frac{1}{t_2}}} \left(arcsin \sqrt{\lambda_1 - \frac{1}{2t_1}} - arcsin \sqrt{\lambda_2 + \frac{1}{2t_2}} \right), \tag{4.39}$$

де $\lambda_1 > \lambda_2$; t_1 та t_2 – загальні напрацювання до проведення заходу та після його втілення.

Величина u наближено розподілена за нормальним законом з параметрами $m = 0, \sigma = 1$. Для неї назначається критична межа на певному рівні значущості α: $u_k = u_{1-\alpha}$, де α – квантиль нормального розподілу при ймовірності, що дорівнює $1 - \alpha$.

Якщо $u > u_{1-\alpha}$, то припускається, що проведений захід має позитивний ефект, якщо ж $u < u_{1-\alpha}$, то можна припускати відсутність ефекту від заходу. На практиці також можливі випадки, коли захід не дає належного ефекту або є негативним. Використання цього критерію дозволяє виявити такі випадки та враховувати їх при впровадженні цього заходу в інших ЕП.

Організаційні заходи мають значну інерційність, тому їх ефективність повинна оцінюватися після значного періоду експлуатації, який визначається індивідуально в кожному конкретному випадку. Практика показує, що цей період у середньому становить 4 – 6 місяців. Важливим етапом при прийнятті рішення про впровадження або розробку заходу для підвищення рівня

надійності є попередня оцінка їх ефективності. Особливо це важливо за наявності кількох альтернативних заходів. Попередню оцінку ефективності заходів можна отримати, використовуючи марківську модель процесу експлуатації.

Під час експлуатації модулі можуть переходити з робочого стану S_1 до стану відмови S_2, що призводить до надзвичайної ситуації зі зниженням довговічності обладнання, а також до стану відмови S_3, який може бути усунений під час експлуатації. Відповідно до принципу зворотної зв'язку між виробництвом та експлуатацією, після дострокового зняття модулів вони піддаються необхідним доробкам для повернення до робочого стану $(S_2 \rightarrow S_1)$. Ефективна діагностика технічного стану модулів під час експлуатації дозволяє перенести частину відмов з категорії зі скороченням довговічності обладнання до категорії з можливістю усунення в процесі експлуатації$(S_2 \rightarrow S_3)$.

Якщо припустити, що перехід модулів з одного стану в інший відбувається незалежно від попереднього кроку і у випадкові моменти часу, то вищезгаданий процес можна визначити як марківський процес з безперервним часом і дискретними станами [49]. Граф стану такого процесу зображений на рис. 53, де символами f_{ij} позначені щільності ймовірностей відповідних переходів: f_{12} щільність ймовірності переходу $(F_x(\text{ПВП}))$ з працездатного стану до стану зі скороченням довговічності обладнання (D-[ДСД] $(S_1 \rightarrow S_2)$);

f_{21}- F_x $S_2 \rightarrow S_1$; f_{13}- F_x $S_1 \rightarrow S_3$;

f_{31}- F_x $S_3 \rightarrow S_1$; f_{23}- F_x $S_2 \rightarrow S_3$;

Зазначені F_x можна знайти за статистичними даними, зареєстрованими протягом контрольного періоду експлуатації модулів.

Для графа, наведеного на рис. 53, S_1, S_2 і S_3ймовірності станів мають наступний сенс: $P(t)$ ймовірність безвідмовної роботи модулів; $F_D(t)$ та $F_{\text{b}}(t)$ ймовірність відмов відповідних категорій. Дані ймовірності повинні відповідати системі диференціальних рівнів Колмогорова:

$$\begin{cases} \frac{dP(t)}{dt} = -(f_{12} + f_{13})P(t) + f_{21}F_D(t) + f_{31}F_{\text{þ}}(t) \\ \frac{dF_D(t)}{dt} = f_{12}P(t) - (f_{21} + f_{23})F_D \\ \frac{dF_{\text{þ}}(t)}{dt} + f_{12}P(t) + f_{23}F_D(t) - f_{31}F_{\text{þ}}(t) \end{cases} \quad (4.40)$$

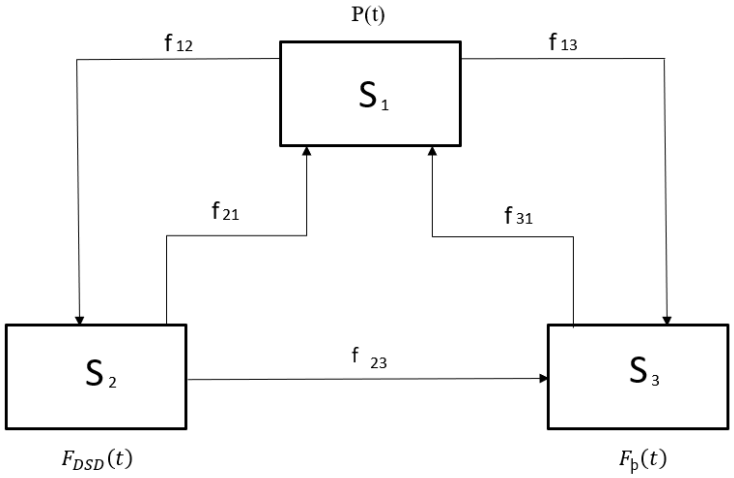


Рис. 4.2 Граф станів експлуатації БРЕО (авіоніки)

Систему рівнянь складено за таким простим правилом графа станів.

Ліва частина кожного рівняння містить похідну ймовірності стану, а права має стільки членів, скільки переходів пов'язано з даним станом. Якщо стрілка переходу спрямована зі стану, то відповідний член має знак «-» , якщо в стан, то знак «+». Кожен член правої частини дорівнює добутку щільності ймовірності переходу за цією стрілкою та ймовірності того стану, з якого виходить стрілка.

Будь-яке з трьох рівнянь системи (4.40) можна замінити умовою нормування ймовірностей: $P(t) + F_D(t) + F_{\text{þ}(t)} = 1$, оскільки стани S_1, S_2, S_3 утворюють повну групу подій.

Початкові умови для системи рівнянь (4.37) слід задавати на основі аналізу вихідного стану БРЕО, який за $t = 0$ можна охарактеризувати рівняннями $P(0)=1$, $F_D(0) = F_{\text{þ}(0)} = 0$. Інтегрування системи (4.38) за умови вказівки на початкових умовах дасть змогу визначити ймовірності станів як функції часу, які необхідні для аналізу ефективності заходів, спрямованих на зменшення числа достроково знятих модулів за рахунок їх доопрацювання і переведення відмов із категорії D у розряд þ.

Кількісна оцінка ефективності заходу може бути проведена кількома методами різної складності та достовірності. Вибір того чи іншого із цих методів залежить від особливостей заходу, можливості одержання вихідних даних і необхідної достовірності.

Розрахунок показників ефективності процесу технічної експлуатації ПС проводиться на кожному організаційному рівні структури АТБ. Для їхнього розрахунку за звітний період використовуються дані, отримані в системі оперативного управління процесом використання парку ЗС і містяться в обліково-звітній документації АТБ і АРЗ.

Попередньо визначаються такі проміжні величини: частота потрапляння в *i*-е

стан процесу технічної експлуатації на r-му рівні $\pi_1 = \frac{n_i^{(r)}}{\sum_{k=1}^{N_r} n_k^{(r)}}$, (4.41)

де $n_i^{(r)}$ і $n_k^{(r)}$ – число потраплянь у *i*-ті, *k*-ті стан процесу технічної експлуатації на r-му рівні; N_r– число станів процесу технічної експлуатації *r*-го рівня;

– середній час перебування ПС в *i*-му стані процесу технічної експлуатації $\mu_i = t_i / n_i$, (4.42)

де t_i- сумарний час перебування в *i*-ом стані процесу технічної експлуатації;

– середні трудовитрати в *i*-му стані процесу технічної експлуатації $\widetilde{t_\iota} = t_i / n_i$, (4.43)

де *ti* - сумарні трудовитрати в *i*-му стані;

– середні витрати коштів у станах процесу технічної експлуатації $\widetilde{C_\iota} = C_i / n_i$, (4.44)

де C_i- сумарні витрати коштів на технічне обслуговування в *i*-му стані, стані процесу технічної експлуатації;

– середнє число відмов і несправностей (загальне), виявлених в *i*-му стані процесу технічної експлуатації з -*j* причини,

$$\widetilde{d_{ij}} = d_{ij}/n_i \ , \qquad (4.45)$$

де $\widetilde{d_i}$ - число відмов і несправностей, виявлених у *i*-му стані з *j*-ї причини;

– середнє число відмов і несправностей (загальне), виявлених у *i*-му стані процесу технічної експлуатації, $\widetilde{d_i} = d_i/n_i \ , \qquad (4.46)$

де d_i - число відмов і несправностей у *i*-му стані; середнє число затримок відправлень у *i*-му стані з *j*-ї причини $\widetilde{m_{ij}} = m_{ij}/n_i \qquad (4.47)$

де m_{ij} - число затримок в *i*-м стані з *j* причини.

Розрахунок показників ефективності процесу технічної експлуатації на *r*-му рівні (*r*=1,2,3) проводиться таким чином:

– напрацювання на відмову і несправність, виявлені в *i*-му стані з *j*-ї причини, $T_{ij}^{(r)} = \frac{\pi_1^{(r)}\mu_{1P}}{\sum_{i=1}^{Nr}\pi_1^{(r)}\widetilde{d_{ij}}}, \qquad (4.48)$

де π_1 - частота призначення в рейс; μ_{1P} - середній наліт у рейсі;

– напрацювання на відмову і несправність, виявлені в *i*-му стані,

$$T_i^{(r)} = \frac{\pi_1^{(r)}\mu_{1P}}{\sum_{i=1}^{Nr}\pi_1^{(r)}\widetilde{d_i}}, \qquad (4.49)$$

– коефіцієнт регулярності вильотів з урахуванням затримок у *i*-му стані з *j*-ї причини,

$$P_{ij}^{(r)} = 1 - \frac{\sum_{i=1}^{N_k} m_{ij}}{n_i} \qquad (4.50)$$

де N_k - число станів затримок; коефіцієнт використання ПС

$$K_U^{(r)} = \frac{\pi_1^{(r)}\mu_{1\Pi}}{\sum_{i=1}^{Nr}\pi_i\mu_i}, \qquad (4.51)$$

– коефіцієнт можливого використання ПС (коефіцієнт погодинної справності)

$$K_B^{(r)} = \frac{\pi_1^{(r)}\mu_1 \sum_{j=1}^{N_r} \pi_j^{(r)}\mu_i}{\sum_{i=1}^{N_r} \pi_j^{(r)}\mu_1}, \quad (4.52)$$

де π_i – частота попадання в стан резерву *Г*, не використання справного ПСка *А* і простоїв за метеоумовами *М* на *r*-му рівні; μ_j- середній час перебування у станах *Г* , *А* , *М* ;

– Питомі простої у станах технічного обслуговування та ремонту

$$K_P^{(r)} = \frac{\sum_{k=1}^{N_r} \pi_k^{(r)}\mu_k}{\pi_1^{(r)}\mu_{1P}}; \quad (4.53)$$

– питома трудоємність технічного обслуговування та ремонту

$$\tau_Y^{(r)} = \frac{\sum_{i=1}^{N_r} \pi_i \tau_i}{\pi_1^{(r)}\mu_{1P}}; \quad (4.54)$$

– питома вартість технічного обслуговування та ремонту

$$C_Y^{(r)} = \frac{\sum_{i=1}^{N_r} \pi_i C_i}{\pi_1^{(r)}\mu_{1P}}. \quad (4.55)$$

Розробка ефективних заходів щодо запобігання відмовам є дуже трудомістким та нетривіальним завданням, що вимагає залучення фахівців відповідного профілю, мобілізації їхнього професійного досвіду та інтуїції. Вирішення цих завдань не завжди піддається математичній формалізації і вимагає застосування методів експертної оцінки. З цією метою у державних органах управління необхідно мати автоматизовану систему експертної оцінки заходів. У рамках такої системи вирішуються завдання прийняття рішень за наявності альтернативних проєктів на основі думки групи експертів. Така система розроблена в рамках реалізації АСН і вирішує наступні завдання:

– збір даних про заходи щодо підтримки заданого рівня надійності АТ в експлуатації (БД відмов та заходів);

– формування БД експертів;

– обробку та аналіз експертних карток .

Процедура проведення експертизи полягає в наступному. Колектив фахівців, які займаються цією проблемою, розбивається на дві частини: розробників заходів та експертів. Перша група фахівців незалежності один від одного розробляє проєкти заходів з проблеми, що виникла. Одночасно по кожному заходу розробляється ката експертної оцінки даного заходу, до якого включені питання, що відображають різні сторони проблеми: технічні, наукові, економічні, комерційні та ін. Кожному питанню (критерію) становлять свою масу у спільній оцінці вирішення проблеми. При формуванні питань, що включаються до експертної картки, також необхідно забезпечити їхню автономність та незалежність.

Основна частина питань картки містить:

– оцінку проєкту заходу за певним критерієм у заздалегідь заданій системі відліку (наприклад, у десятибальній, де 1-мінімальна, 10-максимальна оцінка критерію);

– рівень значущості критерію при оцінці заходу , що визначає його масу в загальній оцінці вирішення проблем (сумарна маса всіх критеріїв має становити 100 %).

Потім проводиться незалежна експертиза проєктів цього заходу, яка полягає в аналізі групою експертів розроблених проєктів заходів та опрацювання їх експертних оцінок.

При оцінці заходів експертами можливі помилки двох видів: систематичні та випадкові. Експерт, схильний до помилок першого виду, встановлює оцінки, які стійко відхиляються від "справжніх" оцінок у бік збільшення або зменшення. Для їх корекції в системі використовуються поправочні коефіцієнти, що розраховуються на підставі статистичного аналізу всіх експертиз, у яких брав участь даний експерт. Помилки другого виду характеризуються величиною дисперсії, тобто розкиду в оцінці заходу.

Використовуючи аналіз отриманих даних оцінюються як проєкти заходів і експерти. Поправочні коефіцієнти для помилок першого виду

розраховуються за таким алгоритмом. У всіх експертизах цього експерта обчислюється відхилення оцінок, проставлених експертом, від середніх оцінок заходу. Потім проводиться усереднення відхилень оцінок, визначення дисперсії відхилень та довірчих інтервалів. Розрахунок всіх оцінок проводиться за кожним критерієм окремо. Обчислення поправочних коефіцієнтів до оцінок здійснюється на основі відхилень оцінок з урахуванням кількості експертиз, в яких експерт приймав участь.

В якості оцінок проєктів заходів використовуються абсолютні, нормовані за масвми, за експертами, ранжировані, багатокритеріальні та комплексні оцінки.

Абсолютні оцінки розраховуються на підставі оцінок, проставлених експертами за кожним критерієм при рівності масових коефіцієнтів.

$$A_i = \frac{1}{m}\frac{1}{r_i}\sum_{j=1}^{m}\sum_{k=1}^{r_i} o_{ijk} \qquad (4.56)$$

де o_{ijk} – оцінка i-го проєкта по j-му критерію k-го експерта; m – число критерієвів r_i – число експертів по i-му проєкту.

Ранг, присвоєний проєкту заходу, визначається шляхом сортування оцінок порядку зменшення

$$RA_i = rang(A_i)$$

Нормовані оцінки, крім усереднення за експертами і критеріями, здійснюють приведення критеріїв за їхніми масовими коефіцієнтами. При нормуванні за масами критеріїв у розрахунку оцінок використовують індивідуальні масові коефіцієнти, усереднені за експертами, які проводили експертизу цього проєкту заходу. Нормовані за масовими коефіцієнтами оцінки обчислюються за формулою

$$Nl_i = \frac{1}{r_i}\sum_{j=1}^{m}\sum_{k=1}^{r_i} o_{ijk}\, vl_{ij} \qquad (4.57)$$

де $v1_{iJ} = \frac{1}{r_i}\left(\sum_{k=1}^{r_i}(v_{ijk}/\sum_{k=1}^{m} v_{ijk}\right)$ – індивідуальний масовий коефіцієнт j-го критерія i-го проєкта ; v_{ijk}-маса j-го критерія i-го проєкта k-го експерта

Ці оцінки є дуже чутливими до помилок фахівців-експертів, оскільки в цьому разі оцінки та рівні значущості критеріїв з оцінки проєктів

визначаються самими експертами.

При нормуванні за вибіркою масові коефіцієнти критеріїв усереднюються за всіма проєктами заходу або за всіма заходами цієї проблеми, що сприяє більш точному оцінюванню значущості критеріїв у межах цієї проблеми, оскільки у формуванні масових коефіцієнтів беруть участь усі фахівці в цій галузі знань. Вираз для їх розрахунку має вигляд

$$N2_i = \frac{1}{r_i}\sum_{j=1}^{m}\sum_{k=1}^{r_i} o_{ijk} v2_j, \qquad (4.58)$$

де $v2_j = \frac{1}{n_v}\sum_{i=1}^{n_v}\frac{1}{r_i}\sum_{k=1}^{r_i}(v_{ijk}/\sum_{j=1}^{m} v_{ijk})$ індивідувальний масовий коефіцієнт j-го критерія по виборці з n_v проєктів.

Для виключення систематичних відхилень в оцінках експертів вихідні оцінки зміщуються на величину поправочних коефіцієнтів, пропорційних розрахунковим відхиленням в оцінках експертів, які залежать від кількості експертиз, у яких експерт брав участь. За достатньої кількості експертиз (>20) поправочні коефіцієнти практично дорівнюють розрахунковим відхиленням. Розрахунок відхилень для абсолютних, нормованих з індивідуальними масовими коефіцієнтами і нормованих за вибіркою оцінок здійснюється за формулами

$$t_{jk} = \frac{1}{r_e}\sum_{i=1}^{r_e}\left(o_{ijk} - \frac{1}{r_i}\sum_{k=1}^{r_i} o_{ijk}\right)$$

$$t_{jk} = \frac{m}{r_e}\sum_{i=1}^{r_e}\left(o_{ijk} v1_{ij} - \frac{1}{r_i}\sum_{k=1}^{r_i} o_{ijk} v1_{ij}\right)$$

$$t_{jk} = \frac{m}{r_e}\sum_{i=1}^{r_e}\left(o_{ijk} v2_j - \frac{1}{r_i}\sum_{k=1}^{r_i} o_{ijk} v2_j\right) \qquad (4.59)$$

де r_e – число проєктів, в яких експерт приймав участь

Для обчислення поправочних коефіцієнтів можна використати формулу

$$p_{jk} = B_{re} t_{jk}$$

де B_{re} – коефіцієнт, залежний від числа проєктів, по яким експерт проводив експертизу [0…1].

Ранжовані оцінки застосовуються в разі, якщо необхідно порівняти кілька альтернативних проєктів заходів. Для отримання оцінок при цьому необхідно, щоб експертиза всіх проєктів проводилася одними й тими самими експертами або проводилося нормування за експертами. У результаті аналізу розраховується ранг заходу і рівень достовірності оцінки (коефіцієнт конкордації). При отриманні ранжированих оцінок як вихідні дані можна використовувати абсолютні або нормовані оцінки, проставлені проєктам заходів кожним експертом

$$a_{ik} = \frac{1}{m}\sum_{j=1}^{m} o_{ijk}, \qquad n1_{ik} = \sum_{j=1}^{m} o_{ijk} v1_{ij}, \qquad n2_{ik} = \sum_{j=1}^{m} o_{ijk} v2_{j}. \tag{4.60}$$

При цьому оцінкам визначаються вихідні ранги $(RA_{ik}, RN1_{ik}, RN2_{ik})$ по кожному експерту та визначають сумарні ранги заходів щодо всіх експертів

$$RSA_i = \sum_{k=1}^{r} RA_{ik}, \qquad RSN1_i = \sum_{k=1}^{r} RN1_{ik}, \quad RSN2_i = \sum_{k=1}^{r} RN2_{ik}, \tag{4.61}$$

де r - число експертів, які брали участь в експертизі.

Для перевірки, чи добре узгоджуються ці n_v ранжування один з одним використовується коефіцієнт конкордації W, що обчислюється за формулою

$$W = \frac{12\sum_{i=1}^{n_v} D_i^2}{r^2(n_v^3 - n_v) - rT}, \tag{4.62}$$

де $D_i = \sum_{k=1}^{r} R_{ik} - \frac{1}{n_v}\sum_{k=1}^{r}\sum_{l=1}^{n_v} R_{ik}$ – відхилення сумі рангів, що приписується всім експертами i-му проєкту від середнього значення цих сум рангів; $T = \sum_{l=1}^{q}(t_l^3 - t_l)$; t_l – число однакових рангів у l-му ранжуванні;

q - число груп пов'язаних рангів; $R_{ik} = RA_{ik}$ (або $RN1_{ik}$ або $RN2_{ik}$);

n_v - кількість проєктів у вибірці; r – кількість експериментів;

Розглянуті вище інтегральні оцінки дозволяють оцінити заходи за кількома критеріями з урахуванням (або без) значущості (маси) критеріїв.

Для порівняльного аналізу проєктів заходи у системі використовуються також методи вирішення багатокритеріальних завдань, зокрема метод порівняння на основі порядкової шкали оцінок. Сутність методу полягає у наступному. Якщо $S = \{s_1, s_2, \dots, s_n\}$ проєкти заходу, що характеризуються критеріями $X_1, X_2, \dots, X_n$, тоді оцінки за показниками утворюють векторні оцінки виду $y = (x_1, x_2, \dots, x_m)$.

Кожному проєкту $s_i \varepsilon S$ відповідає векторна оцінка $y_i = f(s_i) = (x_{1i}, x_{2i}, \dots, x_{mi})$, де x_{ji} – оцінка і-го проєкту по показнику

$X_j; x_{ij} = Xj, i = \overline{1,n}, j = \overline{1,m}.$

Якщо розглядати векторні оцінки як абсолютні оцінки заходу, то

$$x_{ij} = \frac{1}{r_i}\sum_{k=1}^{r_i} o_{ijk}\,. \tag{4.63}$$

При розгляді різних нормованих за індивідуальними масами та вибіркою оцінок, а також абсолютних оцінок з поправкою на експерта та нормованих за вибіркою та експертами оцінок проєктів відповідні показники можна визначити за формулами:

$$x_{ij} = \frac{1}{r_i}\sum_{k=1}^{r_i} o_{ijk} v1_{ij},\ \ x_{ij} = \frac{1}{r_i}\sum_{k=1}^{r_i} o_{ijk} v2_{ij},$$

$$x_{ij} = \frac{1}{r_1}\sum_{k=1}^{r_i} (o_{ijk} + P_{ijk}),\ \ x_{ij} = \frac{1}{r_1}\sum_{k=1}^{r_i} (o_{ijk} + P_{ijk}) v2_j\,. \tag{4.64}$$

Аналіз проєктів заходів здійснюється на основі аналізу матриці векторних оцінок усіх проєктів за всіма показниками $|x_{ji}|$ шляхом покоординатного порівняння пар векторних оцінок проєктів із S. Можливі три результати порівняння: проєкт заходу s_i більш кращий, ніж проєкт s_g $(s_i > s_g)$, якщо всі оцінки вектора $f(s_i)$ не гірші, а принаймні одна оцінка краща за оцінки вектора $f(s_g)$;

1. проєкти заходу s_i і s_g еквівалентні у сенсі переваги $(s_i = s_g)$, якщо оцінки за показниками однакові;

2. проєкти заходу s_i і s_g незрівнянні $(s_i \neq s_g)$, якщо оцінки принаймні за двома показниками l і k такі, що одночасно $x_{li} > x_{lg}$ і $x_{li} < x_{lg}$, $l, k = \overline{1, m}$, $l \neq k$.

Сукупність незрівнянних проєктів заходів та проєктів, еквівалентних їм у сенсі уподобання, утворюють область компромісів S_a. Для виконання необхідних операцій проєкти заходу перенумеровуються і приписують їм відповідні векторні оцінки. Векторна оцінка $f(s_i)$ по черзі порівнюється із векторними оцінками $f(s_g)$. Здійснюється покоординатне порівняння двох векторних оцінок. Якщо проєкти заходу s_i кращі ніж s_g, то проєкти s_g виключають з S; якщо s_g не менш переважно, його залишають у безлічі S. Потім для порівняння обирають наступний проєкт заходу s_{g+l}. Після порівняння з останнім проєктом s_g вибирають наступний s_{i+l} і порівнюють із рештою s_g. У результаті залишаються проєкти заходи, що утворюють безліч непідпорядкованих проєктів $\{s^*\}$. В отриманій підмножині проєкти заходу є найкращими в вибірці, що розглядається, їм присвоюється 1-й ранг. Виділивши ці проєкти з вихідної множини і повторивши процедуру порівнювання аналізу, можна отримати другу групу непідлеглих проєктів, якій надається 2-й ранг. І далі, аналогічним чином, проводиться присвоєння рангів усім проєктам заходу вихідної множини

$$R_i = rang(S) \ .$$

Комплексні оцінки будуються з урахуванням розглянутих вище оцінок. Процедура комплексного аналізу аналогічна методиці отримання ранжированих оцінок, тільки замість оцінок експертів використовуються оцінки, отримані різними методами. Коефіцієнт конкордації тут застосовується з метою оцінки ступеня узгодженості різних методів. Крім

цього коефіцієнт конкордації можна використовувати для оцінки ступеня узгодженості думок експерта при використанні декількох методів одночасно.

4.3. Розробка редукованої динамічної експертної системи для автоматизованого контролю бортового устаткування

Системи контролю базуються на аналізі кількісних і альтернативних ознак.

При контролі за альтернативною ознакою не потрібно знати фактичне значення контрольованого параметра, а потрібно встановити факт відповідності або невідповідності цього параметра встановленим вимогам.

При експлуатації сучасних ПС необхідно реалізувати систему контролю працездатності авіоніки.

Для цього застосовуються різноманітні системи контролю на різних етапах експлуатації бортового обладна. Автоматизовані бортові системи контролю, що включають вбудовані засоби інструментального контролю і системи інформаційного контролю, системи автоматичного контролю бортового устаткування і ін. Використовуються ієрархічні системи засобів контролю, які добре зарекомендували себе на практиці, в яких оцінюється працездатність і достовірність інформації окремих систем і комплексу бортового обладнання в цілому. Однак при вирішенні завдання контролю бортового устаткування доцільно знати не тільки момент відмови бортових систем, але і передбачити момент виникнення аварійної ситуації, а також інтервали недостовірної роботи обладнання.

Вирішення цього завдання за допомогою апріорних прогнозуючих моделей вимагає проведення тривалих дорогих експериментів, не дозволяє враховувати особливості конкретних систем і здійснювати ефективний контроль високоманеврових ПС.

Тому для здійснення контролю бортового устаткування перспективних маневрених ПС доцільно використовувати комплексні системи контролю на базі ДЕС, які дозволяють враховувати режими польоту ПС, мають багату

базу даних і група оціночних критеріїв.

В сучасних БРЕО для забезпечення високої якості формованої інформації використовується апріорна структурна адаптація, під якою розуміється вибір найкращої конфігурації оцінюваної частини повного вектора стану, а також матриць моделі досліджуваної системи для конкретних умов. Адаптація здійснюється за допомогою коваріаційного аналізу на основі мінімаксного критерію якості.

Для ефективного виділення динамічних складових часто потрібні дорогі і трудомісткі льотні експерименти. Наприклад, для достовірного виділення компонент динамічного дрейфу необхідна велика кількість льотних експериментів.

Для використання в системі контролю доцільно формувати редукований вектор стану з використанням концепції системного синтезу [47, 58]. Базуючись на концепції системного синтезу здійснюється скорочення числа параметрів моделі, що характеризують досліджуваний об'єкт, виключаючи їх з вектора стану моделі.

З вектора стану моделей виділяються ключові параметри за допомогою загальних і спеціальних критеріїв [60]. Загальні критерії є універсальними, а спеціальні критерії, як правило, визначають якість досліджуваного процесу на основі будь-яких фізичних характеристик, притаманних аналізованого процесу.

При функціонуванні ПС відбираються параметри, які мають максимальні ступеня спостережливості і керованості [42], що дозволяє відбирати для здійснення контролю БЦОМ тільки ефективно керовані і достовірно оцінюючі параметри.

В процесі функціонування бортового устаткування, склад редукованого вектора стану моделі може змінюватися. Компоненти редукованого вектора стану, що підлягають контролю, вибираються на основі аналізу якісних характеристик кожної компоненти, які залежать від режиму польоту ПС, зовнішніх збурювань, власного стану БЦОМ, а також можливостей

безпосереднього вимірювання за допомогою наявних засобів. Зазвичай в вектор стану для здійснення контролю включають параметри, що характеризують працездатність БРЕО. Але в редукованому векторі стану використовуються тільки частина цього вектора стану – ключові параметри: безпосередньо вимірювані і компоненти, що ефективно спостерігаються.

Для здійснення всеохоплюючого контролю бортового устаткування доцільно використовувати концепції, які закладені в ДЕС і інтелектуальні технології. В якості інтелектуальної технології використовуються алгоритми акцептора дії. Реалізація функції контролю в ДЕС передбачає її модифікацію з метою зменшення обсягу пам'яті використовуємо структуру системи управління з інтелектуальною компонентою, побудована на основі теорії функціональних систем, передбачає побудову прогнозуючих моделей стану об'єкта управління і моделей зовнішнього середовища.

Побудувавши моделі зовнішнього середовища функціонування бортового обладнання, з'являється можливість розробляти сценарії управління на тривалі проміжки часу його функціонування. Сценарії управління припускають реконфігурування обладнання ПС. Функціональна схема системи контролю БРЕО ПС представлена на рисунку 4.3.

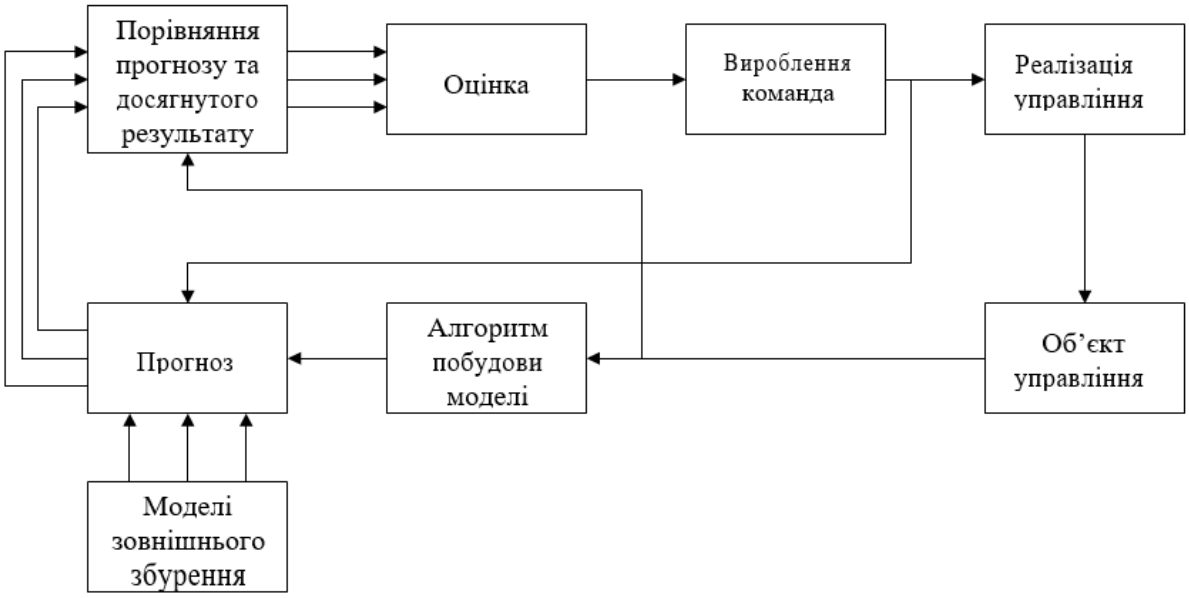


Рис. 4.3. Модель функціональна схема системи контролю з урахуванням моделей зовнішнього середовища

На рис. 4.3 в блоці прогнозу здійснюється прогнозування ключових змінних стану БРЕО ПС з урахуванням різних моделей зовнішнього середовища. Далі прогноз стану БЦОМ порівнюється з критичними значеннями його стану, які визначаються в залежності від режиму польоту ПС.

Функціональна структура рисунку 4.3 використана при синтезі редукованої ДЕС ПС. Структура редукованої ДЕС з інтелектуальною компонентою представлена на рисунку 4.4.

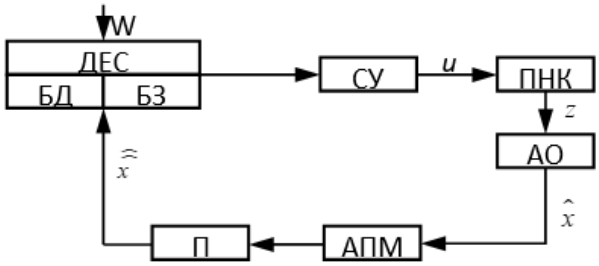


Рис. 4.4: Структура редукованої ДЕС з інтелектуальною компонентою

СУ – система управління; П – алгоритм прогнозу; БД – база даних; БЗ – база знань; w - вектор зовнішніх збурювань; u – вектор управління; z – вектор вимірювань; x оцінка редукованого вектора стану БРЕО; x- скорочений вектор прогнозних значень стану БРЕО.

ДЕС ПС працює на основі інформації БД і БЗ. Інформація про результати функціонування БРЕО ПС надходить в БД, де виконується початкова обробка даних для забезпечення ефективної побудови моделей, виявлення і відсіювання аномальних вимірювань, формування вимірювань в вибірках та ін. В БД інформація зберігається в упорядкованій формі в двох частинах пам'яті - довгострокової і короткострокової пам'яті. У довгостроковій пам'яті зберігаються шаблони, тобто сукупність даних, відповідно чітко обумовленому режиму функціонування ПС і прогнозуючі моделі, які використовуються для конкретних умов польоту. У короткостроковій пам'яті зберігаються всі динамічні дані, які оперативно оновлюються під час вступу нових вимірів.

Попередньо оброблені в БРЕО вибірки надходять в акцептор дії, де

відбуваються оцінювання ключових параметрів, побудова прогнозуючих моделей і прогноз його стану, а також звірення результатів прогнозу з реальними результатами дії.

Оцінювання ключових параметрів БРЕО ПС здійснюється за допомогою алгоритму оцінювання, наприклад фільтра Калмана. Для отримання нерозходжуючихся оцінок в практичних додатках використовують різні адаптивні модифікації фільтра Калмана. В якості алгоритму побудови прогнозуючих моделей використовуються нейронні мережі, алгоритми самоорганізації і генетичні алгоритми [26, 50, 115].

Отримані прогнозуючі моделі для даної конкретної ситуації використовуються для отримання прогнозних значень ключових параметрів БРЕО. Результати прогнозу надходять в БЗ, де порівнюються зі значеннями, встановленими для виконуваного режиму функціонування ПС. Також прогнозними значеннями ключових параметрів БРЕО поповнюються БД ДЕС. З ДЕС інформація про майбутні порушеннях функціонування КБО передається для формування керуючого впливу на КБО.

Як у випадку плавного виходу параметрів за межі допуску, так і в разі різких збоїв відновлення навігаційного комплексу ПС проводиться шляхом підстроювання коефіцієнтів або параметрів алгоритмічного забезпечення КБО. Виявлення неприйнятного рівня похибок здійснено системою контролю ПС з використанням алгоритму самоорганізації. Відновлення працездатності системи проведено за допомогою регулятора в структурі ІНС або прогнозу і компенсації похибок на виході ІНС. Вимірювальний комплекс з функцією регенерації представлений на рисунку 4.5.

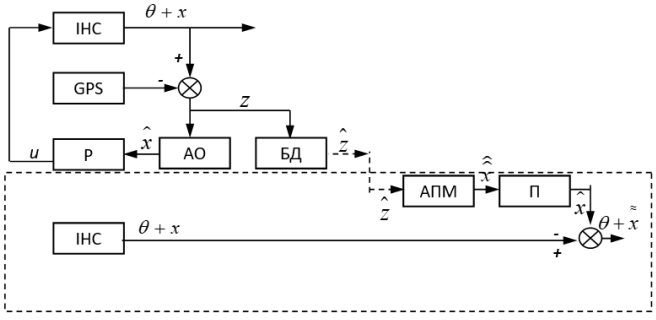


Рис. 4.5. Вимірювальний комплекс з функцією регенерації

θ – справжня навігаційна інформація; х – похибки ІНС; z – вектор вимірювань; $\hat{z}$ – вимірювальна вибірка, АО – алгоритм оцінювання;
АПМ – алгоритм побудови моделей; БД – база даних; П – алгоритм прогнозу, Р – регулятор; $\hat{x}$ – оцінка вектора похибок ІНС; $\hat{\hat{x}}$ – прогноз похибок ІНС; $\tilde{\tilde{x}}$ – помилка прогнозування; u – вектор управління.

Для діагностики нелінійних моделей КБО ПС використаний алгоритм побудови прогнозуючої моделі похибок з використанням ГА або алгоритму самоорганізації.

В сучасних КІО КБО для забезпечення високої якості формованої інформації використовується апріорна структурна адаптація, під якою розуміється вибір найкращої конфігурації оцінюваної частини повного вектора стану, а також матриць моделі досліджуваної системи для конкретних умов [9].

Адаптація здійснюється за допомогою коваріаційного аналізу на основі критерію якості мінімального та максимального. Для ефективного виділення динамічних складових вектора стану часто потрібні дорогі і трудомісткі льотні експерименти. Наприклад, для достовірного виділення компонент динамічного дрейфу необхідна велика кількість льотних експериментів.

Для використання в системі контролю доцільно формувати редукований вектор стану з використанням концепції системного синтезу. У відповідність з цією концепцією необхідно раціональне скорочення числа параметрів моделі, що характеризують досліджуваний об'єкт. Здійснити таке скорочення можна шляхом використання різних критеріїв. У векторі стану моделей виділяються ключові параметри за допомогою загальних і спеціальних критеріїв. Загальні критерії є універсальними, а спеціальні критерії, як правило, визначають якість досліджуваного процесу на основі будь-яких фізичних характеристик, притаманних аналізованого процесу.

Потім, в залежності від режиму польоту ПС, формуються різні моделі. Наприклад, при вчиненні інтенсивного маневрування ПС вибирають параметри, які мають максимальні ступені спостережливості і керованості

[13, 91], що дозволяє відбирати для здійснення контролю КБО тільки ефективно керовані і достовірно оцінюються параметри.

В процесі функціонування КБО склад редукованого вектора стану моделі може змінюватися. Компоненти цього вектора, що підлягають контролю, вибираються на основі аналізу якісних характеристик кожної компоненти, які залежать від режиму польоту ПС, зовнішніх збурень, власного стану КБО, а також можливостей безпосереднього вимірювання за допомогою наявних засобів. Зазвичай в вектор стану для здійснення контролю включають параметри, що характеризують роботоздатність КБО. Але в редукованому векторі стану використовуються тільки частини цього вектора стану – ключові параметри: безпосередньо вимірювані і компоненти, що ефективно спостерігаються .

ДЕС ПС працює на основі інформації бази даних і бази знань. Інформація про результати функціонування КФС ПС надходить в БД, де виконується початкова обробка даних для забезпечення ефективної побудови моделей, виявлення і відсіювання аномальних вимірювань, формування вимірювань в вибірках та ін.

В БД інформація зберігається в упорядкованій формі в двох частинах пам'яті – довгострокової і короткострокової. У довгостроковій пам'яті зберігаються шаблони, тобто сукупність даних, відповідна чітко обумовленому режиму функціонування ПС, і прогнозуючі моделі, що використовуються для конкретних умов польоту. У короткостроковій пам'яті зберігаються всі динамічні дані, які оперативно оновлюються під час вступу нових вимірів.

Попередньо оброблені в КБО вибірки надходять в акцептор дії, де відбуваються оцінювання ключових параметрів КБО, побудова прогнозують моделей і прогноз його стану, а також звірення результатів прогнозу з реальними результатами дії.

Отримані прогнозуючі моделі використовуються для отримання прогнозних значень ключових параметрів КБО. Результати прогнозу

надходять в БЗ, де порівнюються зі значеннями, встановленими для виконуваного режиму функціонування ПС. Також прогнозними значеннями ключових параметрів КБО поповнюються БД ДЕС [21, 34]. З ДЕС інформація про майбутні порушення функціонування КБО передається в СУ для формування керуючого впливу на КБО.

Запропоновано систему контролю і діагностики КБО з функцією відновлення , рисунок 4.6.

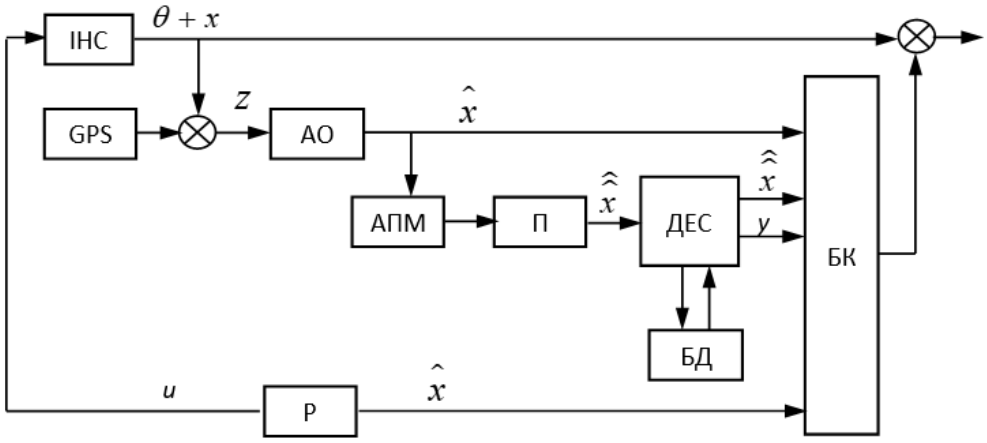


Рис. 4.6 Система контролю і діагностики КБО з функцією відновлення

БУ – блок управління корекцією; у – керуючий сигнал ДЕС. Діагностуються перевищення порогових значень похибок ІНС, парирування яких дозволяє утримати ІНС в області достовірних значень навігаційної інформації ПС, запобігши розвиток ситуації, що призводить до втрати працездатності КБО. При реалізації системи контролю використана структура ДЕС і FDI-метод, який передбачає використання нечіткої логіки. Передбачається використання даних з БД випробувань КБО, які надходять на вхід математичної моделі, а потім вихідні вектори параметрів моделі порівняти з даними випробувань. Одержуваний вектор відхилень піддається аналізу базою нечітких правил ДЕС, а в результаті формується діагностичне рішення про стан КБО. Елементи нечіткої логіки застосовуються спільно з методом діагностичних матриць (матриць Л. А. Урбана).

Моделі в векторній формі мають вигляд:

$$\delta x = A\text{-}1 \cdot B \cdot \delta y, \qquad (4.65)$$

де δx – вектор стану КБО; δy – вектор діагностичних ознак вимірюваних параметрів КБО; B і A – матриці коефіцієнтів, які встановлюють кількісний взаємозв'язок параметрів на виділених режимах функціонування КБО. Діагностична матриця конкретного КБО представляє собою таблицю чисельних значень і, що дозволяють на основі відхилень частини вимірювальних параметрів визначати відхилення безпосередньо не вимірюваних параметрів стану.

Створення бази нечітких правил здійснюється на основі діагностичної матриці, рядки якої лягли в основу створення БЗ і функцій приналежності відповідних лінгвістичних змінних.

4.4. Модель інформаційного процесу проєктування інтегрованої модульної авіоніки комплексу бортового обладнання.

Процес проєктування – це складний ітераційний процес послідовного характеру, який тісно пов'язаний з процесом оцінки безпеки. Весь процес розділений на етапи і рівні. Етапи проєктування зазначені в моделі загальними формулюваннями і не суперечать ДЕСТ 34.601-90 [54]. У моделі поділ на рівні передбачає розмежування процесів проєктування ПС, КБО, систем і модулів один від одного і визначає механізми взаємозв'язків між ними. Між рівнями проєктування існують жорсткі ієрархічні зв'язки. Це забезпечує єдність концепцій проєктування модулів у рамках системи, систем у рамках КБО. Зв'язки між рівнями відображені в документації, яка містить вимоги. Вимоги до будь-якої складової частини виробу формуються поєднанням вимог верхнього рівня і приватних вимог. Виділимо два основних види вимог:

– Технічні вимоги.
– Вимоги безпеки.

Технічні вимоги визначають основні технічні характеристики, а саме: функції, конструктивне виконання, дизайн виробу, міцність та ін.

Вимоги безпеки закріплюють певні характеристики надійності виробу,

формують перелік заходів і процедур для забезпечення і гарантії безпеки проєктування і подальшої експлуатації. Вимоги безпеки на перший погляд мають менш важливий пріоритет, ніж вимоги технічні. Такий стан справ у більшості галузей промисловості. Через високу ймовірність загибелі людей у разі відмови обладнання ПС в авіаційній промисловості вимоги безпеки мають надвисокий пріоритет.

Представлена модель, рисунок 4.7 – є поєднаною моделлю процесів проєктування і оцінки безпеки. З неї можна бачити, як тісно сучасний процес проєктування КБО пов'язаний з процесом оцінки його безпеки.

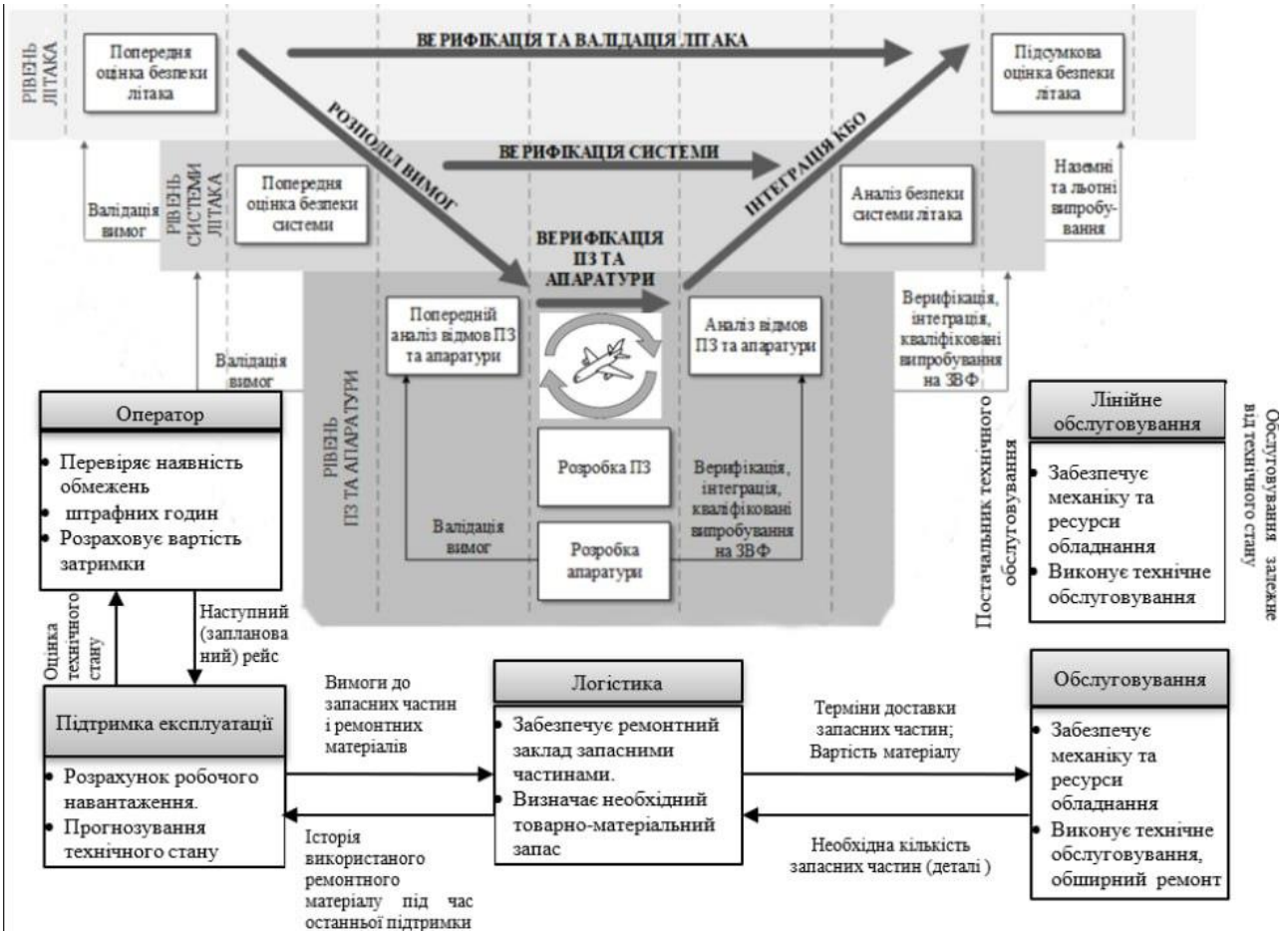


Рис. 4.7 Модель процесу проектування та експлуаиації комплексу бортового обладнання

Численні оцінки безпеки на кожному з рівнів проєктування виконують роль головного критерію переходу між ними. Причому критерієм обмежений не лише процес завершення етапу проєктування, а і його початок. Настільки висока контролююча роль процесу оцінки безпеки є результатом реалізації концепції наскрізного проєктування авіоніки [65, 66].

Багато систем управління польотами в цивільній авіації були успішно розроблені за допомогою спеціального обладнання та програмного забезпечення. Основні засади комп'ютерного програмного та апаратного забезпечення для авіоніки потребують докладного дослідження як, зокрема, теоретико-методологічних принципів розробки та практичного застосування програмного забезпечення.

Обґрунтування базових засад програмного забезпечення в авіоніці набуває розвитку по мірі того, як проводяться дослідження та набувається досвід щодо автоматизації та архітектури підтримки програмних продуктів. Специфіку в галузі структурного аналізу обумовлює термін архітектура, що описує типологію методів, їх функціонал та взаємовідносини складових частин системи, що моделюється.

Функціональні засади розробки та експлуатації програмного забезпечення обумовлюють аналіз та вироблення різноманітних рішень щодо удосконалення, новації тощо діяльності авіаційних виробів та авіаційних підприємств зокрема. Зазначені засади формуються на ідеях інтеграції, яка є складовою частиною комплексного аналізу бізнес-процесів.

Дослідження систем авіоніки, що має модульну багаторівневу структуру при цьому класичні підходи до розробки систем експлуатації повітряних суден є малоефективними, що потребує розробки специфічної методики, яка враховує предметну сферу, що досліджується. Це пояснюється необхідністю інтеграції програмного забезпечення експлуатації суден цивільної авіації зі сторонніми програмно-апаратними комплексами, які використовуються щодо контролю за польотами. При цьому велике значення в даному контексті набувають наступні практично значущі конструкції та компоненти: архітектура програмного забезпечення та наступна за нею V-образна модель процесу проектування ПС, а також єдність та цілістність між структурними компонентами, рисунок 4.8.

В роботах [1, 2] наведені результати досліджень щодо розробки системи підтримки рішень управління повітряним рухом, що потребує

ітеративного процесу проектування. Окрема підсистема проектування виконує виокремлення, що направлені на стандартні платформи, які можуть бути використані в декількох типах додатків одночасно. У нашому дослідженні ми використовуємо щодо сумісного використання декількох додатків та повторного використання обчислювальних ресурсів, що насамперед впливає на практично-дослідницьку частину нашого дослідження.

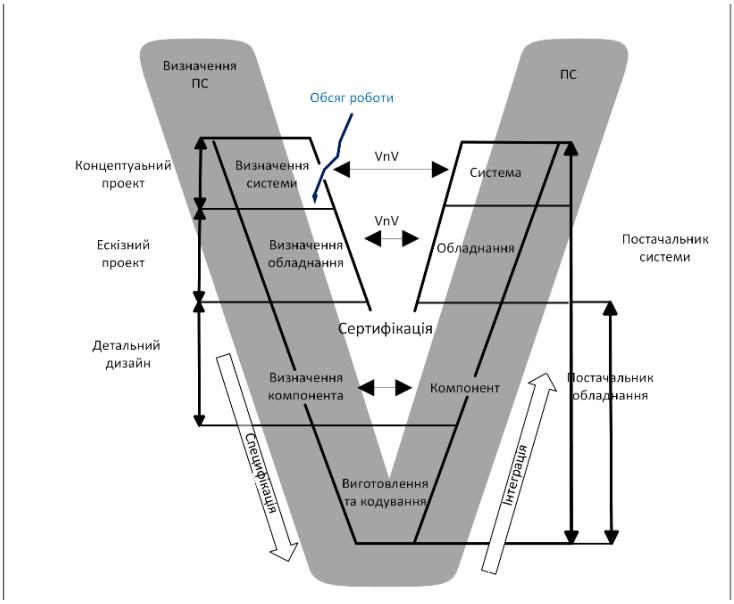


Рис. 4.8Структурна V-модель з розмежуванням процесів верифікації, валідації та сертифікації систем IMA

В роботі [68] наведені результати дослідження сертифікації систем обробки в авіоніці та деякі питання інтегрувальної діяльності та аналізу програм забезпечення. Це дозволило нам сформулювати вимоги до платформи візуалізації, що підтримує декілька розділів операційної системи реального часу (RTOS) на багатоядерному процесорі, що використовується в авіоніці.

В роботі [17] наведені результати дослідження відновлення спостереження вимог на основі пошуку щодо еволюційних обчислень (CEC). Вказано, що спостереження вимог є багатоцільовою задачею пошуку, обумовленого призначенням кожної вимоги одному або декільком

програмним елементам (елементам коду, документації API та коментарям). Це дозволяє обґрунтувати призначення недомінуючого генетичного алгоритму сортування (NSGA-II).

В роботі [18] наведені результати дослідження архітектури AECS Airbus A-380, Boeing 787, Sukhoi Superjet 100 и MC-21. Вказано, що ці системи розроблені з аналогічним дизайном та містять центральну обробку та передачу даних. Інтерфейси ARINC 825, ARINC 664, ARINC 429 використовуються для обміну даними між ними. Проте наступна модернізація таких систем обмежена через низьку гнучкість поза розподілом мережи. Все це дозволяє стверджувати, що доцільним є проведення дослідження щодо функціональних засад розробки та експлуатації програмного забезпечення в авіоніці з врахуванням нагальних питань надійності системи, передачі даних, зниження рівня шуму, зменшення ваги та габаритів кабельної мережі.

З розвитком інтегрованої модульної авіоніки, динамічна реконфігурація не тільки надає переваги щодо використання ресурсів та конфігурації літака, так і використовується як діючий засіб щодо управління збою ресурсу. В роботах [69] наведені результати дослідження процесів динамічної реконфігурації в авіоіці. В дослідженні ретельно проаналізовані елементи процесу динамічної реконфігурації. Досліджено системи мережи Петри, яку широко використовують як інструмент проведення моделювання. Даний метод валідації динамічної реконфігурації IMA переважний щодо початку проектування системи.

Серед перспективних стандартів розробки та подальшої експлуатації програмного забезпечення в авіоніці та, загалом, в авіакосмічній галузі є стандарт SPASE. Головною метою зазначеного стандарту є забезпечення сумісності різних видів програмного обладнання, передачі даних та управління інформацією щодо вирішення конкретних завдань в авіаційній та космічній галузях [8]. Подальшого вивчення потребують питання, які стосуються оптимізації процесів розробки зазначеного програмного

забезпечення з урахуванням сучасних тенденцій у розвитку інформаційних технологій.

Однак питання, які стосуються створення програмних комплексів для керування окремими процесами в цивільній авіації (у т.ч., контролем польотів), є недостатньо дослідженими у вітчизняній літературі. Останнє обґрунтовано специфікою цієї галузі та тенденцією, відповіднод до якої домінуюче місце на ринку IT-технологій належить іноземним компаніям [9, 10].

На практиці постачальники аерокосмічного обладнання зацікавлені у використанні багатоядерних процесорів в своїх системах. Декілька ядер, що інтегровані в один пристрій, дозволяють з'єднати більше функцій на одному процесорі та в одному обладнанні. Розробка та використання найбільш складного обладнання авіаційною промисловістю щодо виконання важливих програмних функцій повітряних суден обумовлює нові питання щодо безпеки та сертифікації .

Проблема визначення функціональних засад розробки та експлуатації програмного забезпечення в авіоніці базується на ґрунтовних елементах неоднакової практики застосування програмних рішень.

Зокрема, система застосування динамічної реконфігурації та інші модульні конструкції також потребують застосування функціональних та узагальнених засад при безпосередній експлуатації, що дозволить уникнути багатьох недоліків та несправностей тощо.

Низка архітектур та стандартів до розробки IMA, які використовують вимоги специфікації 6532 ARINC [15]. Специфікація 653 ARINC визначає на високому рівні примірник програмного забезпечення для архітектури IMA. Широке прийняття та підтримка ARINC 653 очевидна також у середовищі майбутнього повітряного потенціалу (FACE) для США у програмах військової авіоніки. Ці та інші стандарти IMA ставлять нові вимоги до архітектури програмного забезпечення, особливо впровадження RTOS (Real-Time Operating System), що надаються постачальником COTS [15].

На рисунку (4.9) побудована інформаційна система процесів проектування та експлуатації бортового обладнання

Рис. 4.9. Інформаційна система процесів проектування та експлуатації бортового обладнання

ACARS (Aircraft Communication Addressing and Reporting System) - система зв'язку, адресації та звітності повітряного судна, яка дозволяє ПС відправляти та отримувати повідомлення через радіо або супутниковий зв'язок.

- ACM (Airport Capacity Management) - управління пропускною здатністю аеропорту для оптимізації використання його ресурсів.

- ADS-B (Automatic Dependent Surveillance-Broadcast) - автоматичне спостереження з використанням трансляції, яке дозволяє літакам транслювати своє місцезнаходження та іншу інформацію.

- AIM (Aeronautical Information Management) - управління аеронавігаційною інформацією для підтримки безпечного і ефективного повітряного руху.

- AMSS (Aeronautical Mobile Satellite Service) - аеронавігаційна рухома супутникова служба, яка надає зв'язок для авіації через супутники.

- ASMGCS (Advanced Surface Movement Guidance and Control Systems) - розширена система керування рухом на поверхні аеропортів для підвищення безпеки та ефективності.

- ATM (Air Traffic Management) - організація повітряного руху, включаючи всі служби, що забезпечують безпечне і ефективне переміщення повітряних суден.

- ATN (Aeronautical Telecommunication Network) - аеронавігаційна телекомунікаційна мережа, призначена для обміну даними між повітряними суднами та наземними станціями.

- OME (Однозначного відповідника у загальноприйнятому списку немає) може бути помилково вказано або потребує уточнення.

- DVOR (Doppler VHF Omnidirectional Range) - доплерівська VHF система омнідирекційного діапазону, яка використовується для навігації.

- FIXM (Flight Information Exchange Model) - модель обміну польотною інформацією, стандартизований формат для обміну даними про польоти.

- GBAS (Ground-Based Augmentation System) - наземна система посилення, яка покращує точність, надійність і доступність навігаційних сигналів GNSS.

- GNSS (Global Navigation Satellite System) - глобальна навігаційна супутникова система, яка включає різні супутникові системи, такі як GPS (США), GLONASS (Росія), Galileo (ЄС) і Beidou (Китай).

- ILS (Instrument Landing System) - система посадки за приладами, яка дозволяє літакам здійснювати посадку за будь-яких погодних умов.

- LDACS (L-band Digital Aeronautical Communication System) - цифрова аеронавігаційна система зв’язку в L-діапазоні, спрямована на покращення зв'язку між повітряними суднами та наземними станціями.

- PDS (Pre-Departure Sequencer) - секвенсор перед відправленням, система для оптимізації розкладу вильотів.

- RPA (Remotely Piloted Aircraft) - дистанційно пілотований літальний апарат, інший термін для безпілотного літального апарату.

- TSAT (Target Startup Approval Time) – цільовий час затвердження запуску, використовується в управлінні повітряним рухом.

- TTOT (Target Takeoff Time) – цільовий час зльоту, ключовий елемент в плануванні повітряного руху.

over Internet Protocol) - голос через Інтернет-протокол, технологія, що дозволяє здійснювати голосові дзвінки через Інтернет замість традиційних телефонних ліній.

- VPN (Virtual Private Network) - віртуальна приватна мережа, забезпечує зашифроване з'єднання через Інтернет для безпечного доступу та передачі даних.

- VTT (Variable Taxi Time) – змінний час таксі, враховує різні фактори, які можуть вплинути на час, необхідний літаку для переміщення від терміналу до злітно-посадкової смуги і навпаки.

- WXXM (Weather Information Exchange Model) – модель обміну інформацією про погоду, стандартизований формат для обміну погодною інформацією в авіації.

Більшість доступних інструментів не підтримують автоматичного оновлення посилань відслідковування. Однією з проблем в обслуговуванні програмного забезпечення є автоматична підтримка відслідковування вимог. Процес генерації відслідковування вимог займає багато часу та схильний до помилок [4]. Варіантом подолання відповідних труднощів може бути практика певних компаній щодо накопичення історій змін з попереднього досвіду обслуговування.

Протягом останнього десятиліття виробники оригінального обладнання розглянули питання про використання систем на базі COTS (Commercial Orbital Transportation Services) на підставі роботи елементів повної вартості життєвого циклу персоналізованих систем. У той же час, спостерігається тенденція відходу від розробки систем на основі об'єднаних архітектур, де кожна окрема підсистема виконує спеціальну функцію. Простежується направлення щодо загальних обчислювальних платформ, які можуть

використовуватися в декількох типах додатків і, в деяких випадках, можуть одночасно запускати кілька додатків. Наведений підхід, відомий як інтегрована модульна авіоніка або IMA, призводить до меншої кількості підсистем, які займають менше місця і знижують вартість та енергоспоживання. Ряд розробників програмних рішень для цивільної авіації стикаються з деякими проблемами щодо їх оптимізації на основі використання принципів IMA.

Отже нами була запропонована методолгія інформаційної підтримки процесів проектування та експлуатації ПС.

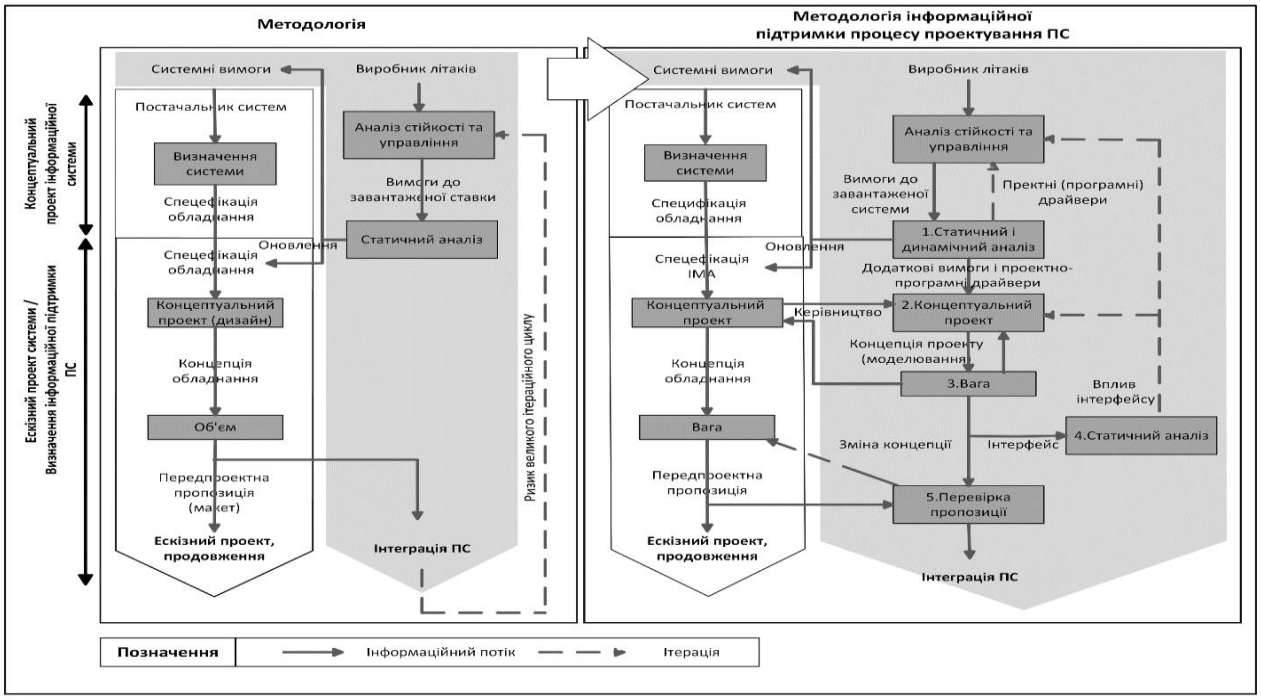


Рис. 4.10. Методологія інформаційної підтримки процесу проектування та експлуатації ПС

Сучасні засоби інформатизації дають можливість автоматизувати окремі процедури, які виконуються в рамках процесу організації та управління польотами у цивільній авіації. Управління та контроль за операціями, на думку деяких авторів [9], здійснюється на основі реалізації адаптаційних багатогіпотезних алгоритмів, що оцінюються на базі методу Байєса. Вищезазначені алгоритми забезпечують моніторинг траєкторії руху у тривимірній системі координат.

Формування програмного забезпечення з врахуванням ризик-

орієнтованого підходу, що використовується в експлуатації повітряних судин, виходячи з наступних передумов [12]:

– ідентифікація загроз (список небезпечних – подій) та факторів;

– ідентифікація ризику подій (оцінювання ризиків);

– визначення шляхів для зменшення ризиків (за методом управління рівнями);

– оцінка економічних витрат на зниження ризиків до прийнятного рівня;

– створення апостеріорної (активної) та апріорної (проактивної) експертної бази знань для оцінювання ризиків з метою прийняття проактивно коригуючих рішень, спрямованих на їх зниження;

– превентивне (проактивне) прогнозування небезпеки;

– ідентифікація ризиків із застосуванням поняття «загроза» та «небезпека» як можливого проактивного прогнозування стану системи при можливих небезпечних ситуаціях за видами загроз.

Важливими критеріями ефективності програмного устаткування є модульність, простота інтеграції, легкість експлуатації та адаптивності системи, що у сукупності набуває пріоритетності в галузі програмного забезпечення. Стандартизація програмного забезпечення обумовлює технологічність процедури PNP. Технологія Plug and Play насамперед була галузевим стандартом для автоматичної обробки та експлуатації програмного обладнання, що спрощувало дію та взаємодію програм та їх комплектуючих [6].

Проте при виникненні помилок у системах IMA механізми реагування, наприклад, відмовостійкості компонентів тощо першими почнуть реагувати.

RelatedWork для динамічної реконфігурації (рис. 15).

На основі ARINC 653 (помилки) в IMA та механізмах відповіді класифікуються, як показано у рис. 16.

У випадках неможливості вирішення помилки механізмами реагування, помилка викликає динамічну реконфігурацію. В таких випадках реконфігурацію відносимо до ПЗ, тому що апаратний збій стає

невідворотним [14].

Динамічна реконфігурація ІМА може змінювати завдання в залежності від вимог та дієво відновлюватися від відмови [2]. Це робить систему гнучкішою, знижує надмірність обладнання та вартість позапланового обслуговування. Крім того, залученість людського фактору збільшує складність динамічної реконфігурації.

Найбільш доцільним методом контролю є дублювання, тому що обидва автомата ідентичні, їхні входи об'єднані і обидва працюють від єдиної системи синхронізації. Мажоритарні схеми застосовуються для контролю найбільш відповідальних вузлів у тих випадках, коли інші способи застосувати важко. Найбільш поширені на практиці способи, засновані на зіставленні вихідних сигналів (числовий контроль за модулем, корегувальні коди). Кодовий контроль за модулем відрізняється від числового тим, що в якості контрольних слів використовуються залишки від ділення суми цифр даного слова на обраний модуль контролю (парність-непарність).

Розробили класифікацію методів апаратурного контролю, рисунок 4.10

Рис. 4.10. Класифікація методів апаратурного контролю

Визначено набір показників, які можна використовувати при проєктуванні систем діагностування, а також для порівняння варіантів систем діагностування. Вимоги ефективності передбачають вимоги з надійності, аналіз яких для більшості випадків показує необхідність діагностування та відновлення. Від системи діагностування потрібно зафіксувати місце виникнення дефекту і сформувати команду на реконфігурацію об'єкта. Класифікація методів і принципів функціонального діагностування показана на рис. 19.

Рис. 19. Класифікація методів функціонального діагностування

Корегувальні коди застосовуються при передачі і зберіганні інформації, при цьому автомат *A* інформацію не переробляє. У той же час виходить менший обсяг апаратури і підвищується ефективність її роботи. Між тестами для перевірки та діагностичними тестами немає чіткої межі. Як правило, діагностичні тести будуються на базі тестів для перевірки, тобто використовують відомості про стан пристрою, одержані в результаті застосування тестів для перевірки. Для діагностики складних пристроїв, особливо якщо необхідного часу діагностики замало, необхідно застосовувати умовні послідовні тести. Пристрій, що підлягає діагностиці, розглядається як кінцевий автомат з відомою внутрішньою структурою.

Методи функціонального діагностування можна класифікувати за декількома критеріями, такими як тип вимірювань, вид знаходження несправностей, способи діагностування:

За типом вимірювань:

пряме діагностування: безпосереднє вимірювання параметрів або величин, що характеризують стан системи;

косвене діагностування: визначення стану системи на основі вимірювання параметрів, які не є безпосередніми ознаками стану системи, але пов'язані з ним.

За видом знаходження несправностей:

модельне діагностування: використання математичних моделей для прогнозування стану системи та виявлення відхилень від очікуваних значень;
сенсорне діагностування: виявлення несправностей на основі вимірювань, здійснюваних за допомогою датчиків або інших засобів спостереження;
задане діагностування: встановлення можливих несправностей на основі заздалегідь заданих критеріїв або порігових значень.

За способами діагностування:

аналітичне діагностування:використання аналітичних методів для аналізу даних та виявлення відхилень;
емпіричне діагностування: виявлення несправностей на основі досвіду, експертної оцінки або попередніх знань про систему.

При контролі технічного стану об'єкта в просторі параметрів визначаються їхні поточні значення (коефіцієнти передавальних функцій, постійні часу та ін.) і оцінюється відхилення їх від номінального значення. Номінальні значення зазвичай бувають відомі, труднощі пов'язані зі складністю вимірювань поточних значень. При контролі в просторі сигналів перевіряється відхилення вихідних сигналів об'єкта і його блоків від теоретичних значень. При цьому проблема, навпаки, полягає в необхідності безперервного визначення номінальних значень вихідних сигналів для

поточних значень вхідних сигналів. Основне завдання ідентифікації полягає в отриманні або уточненні математичного опису об'єкта за вимірюваннями його вхідного і вихідного сигналів. Методи ідентифікації спрощуються при використанні апріорної інформації про номінальні значення параметрів справного об'єкта і моделі дефектів. При розпізнаванні образів поточний стан об'єкта характеризується вектором у просторі діагностичних ознак, які розділені на області, що відповідають тим чи іншим дефектам.

Метод словників дефектів або діагностичних таблиць (таблиць несправностей) передбачає складання списків найбільш характерних несправностей і відповідних їм значень діагностичних ознак. Зазвичай під несправністю розуміють неприпустиме відхилення деякого параметра від номінального значення, а її ознакою служить перевищення граничного рівня деяким сигналом. Тому даний метод застосовується і в просторі ознак, і в просторі сигналів.

Функціональне діагностування динамічних об'єктів в просторі сигналів точніше відповідає змістовній меті перевірки правильності функціонування об'єкта. В першу чергу для об'єктів, призначення яких – у перетворенні вхідних сигналів на вихідні. При використанні простору параметрів виконання цієї основної функції перевіряється побічно за значеннями параметрів об'єкта. У просторі сигналів правильність вироблення вихідних сигналів перевіряється безпосередньо шляхом їхнього аналізу. Одним з методів при цьому є аналіз характеристик сигналів (амплітуди, частотних властивостей та ін.). Недоліками цієї групи є необхідність апріорної інформації про характеристики вихідних сигналів, неминуча залежність характеристик від вхідних сигналів (поведінка яких заздалегідь найчастіше невідома), недостатня повнота контролю.

Зазначені недоліки не притаманні групі методів, заснованій на використанні алгебраїчних інваріантів. У цих методах діагностування здійснюється шляхом перевірки деяких алгебраїчних співвідношень (контрольних умов), яким має задовольняти сукупність вихідних сигналів

об'єкта, доповнена при необхідності надлишковими сигналами. Інваріантність контрольних умов полягає в тому, що при відсутності дефектів вони зобов'язані виконуватися для будь-яких вхідних сигналів і в будь-який момент часу. Загальна схема функціонального діагностування на основі алгебраїчних інваріантів показана на рисунку 4.11.

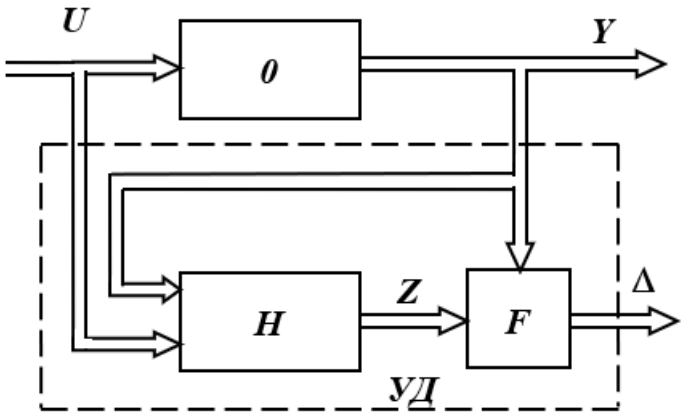


Рис. 4.11. Функціональне діагностування на основі алгебраїчних інваріантів

На пристрій діагностування надходять вхідні U і вихідні Y-сигнали об'єкта, що перевіряється O. На їхній основі блок H виробляє допоміжні (надлишкові) сигнали Z, що задовольняють алгебраїчному рівнянню $\Delta=F(Y,Z)=0$, яке інваріантне щодо вектора вхідних сигналів. Якщо в результаті збою вектор Y буде спотворений, це призведе до появи ненульового сигналу Δ. На практиці замість порівняння з нулем аналізується перевищення допуску (через похибки) $|\Delta|\leq\varepsilon$. Найбільш відомими методами цієї групи є дублювання і резервування. При цьому в якості блоку H виступає другий примірник об'єкта (Y на вхід не надходять), діагностування полягає в порівнянні виходів за умовою $\Delta=Y-Z=0$. Для зменшення апаратурної надмірності можна використовувати не дублікат об'єкта, а його спрощену або апроксимувальну модель, зменшувати число виходів, перевіряти правильність функціонування в окремих режимах. Сюди ж належить побудова за структурними або функціональними схемами логічних моделей, що відбивають причинно-наслідкові зв'язки між блоками. Тут застосовується і кінцево-автоматна апроксимація. Іншим напрямком є побудова моделей, що

використовують в якості діагностичних ознак внутрішні, безпосередньо не спостерігаються координати об'єкта. Іноді вихідні сигнали можуть задовольняти відомому алгебраїчному співвідношенню виду $F(Y)=0$, при цьому пристрій діагностування буде містити тільки блок F. Такі об'єкти мають природну надмірність. Існують критерії аналізу об'єкта діагностування з виявленням в них природної надмірності, яка дозволяє будувати прості пристрої діагностування за рахунок індивідуальних особливостей об'єкта.

У методі надлишкових змінних вводиться штучна надмірність так, щоб розширена сукупність змінних задовольняла контрольній умові.

У контексті авіаційних систем надлишкові змінні використовуємо для забезпечення надмірності або надійності системи. Це може включати введення додаткових компонентів, дублювання функцій або створення резервних шляхів для забезпечення нормальної роботи системи навіть у випадку виникнення несправностей. В авіоніці можуть бути використані надлишкові системи управління, де кожна система має свої власні датчики та виконавчі пристрої. Це дозволяє системі продовжувати працювати в разі відмови однієї з систем, оскільки інша може продовжувати забезпечувати необхідні функції.

Такий підхід забезпечує високий рівень надійності та безпеки авіаційних систем шляхом уникнення потенційних однопунктових відмов і забезпечення резервних шляхів для забезпечення безперебійної роботи.

Одним із способів забезпечення надійної роботи пристрою є проведення тестування, тобто перевірки пристрою шляхом подачі на вхід спеціальних (тестових) впливів і аналіз реакцій на виході. Значна частина фізичних дефектів проявляється у вигляді одиночних константних несправностей. Існують ефективні методи тестування цього класу. Тестування більш складних класів стикається зі значними труднощами.

Процес контролю поетапний. Він включає в себе сприйняття значень контрольованих параметрів (отримання інформації про стан організації), зіставлення значень контрольованих параметрів з допусками і видачу

результатів зіставлення.

Процес контролю передбачає наявність об'єкта контролю, з одного боку, і засоби контролю, з іншого, будемо розуміти під об'єктом контролю будь-який тип об'єкта, інформацію про відповідність стану якого пропонованим вимогам належить отримати, а під засобами контролю – будь-які спеціалізовані ресурси і технічні засоби, що служать для отримання і перетворення цієї інформації. Система контролю - сукупність об'єкта і засоби контролю.

При розгляді процесів контролю, а також при проектуванні засобів - систем контролю виникають наступні завдання:

1. передбачення очікуваного результату процесу контролю в умовах, що включають елемент випадковості;

2. синтез оптимальних засобів контролю, тобто вибір оптимальних з точки зору результату процесу контролю параметрів і елементів застосовуваних засобів;

3. оптимальна організація процесу і системи контролю.

Тому, розроблено алгоритм роботи процесу контролю рис.21

Функціонування систем контролю носить стохастичний характер з огляду на стохастичних властивостей об'єктів контролю (в іншому випадку відпала б необхідність контролю) та обмежених точності і надійності засобів контролю. Тому при формалізації опису функціонування систем контролю для оцінки їх ефективності слід використовувати математичний апарат теорії ймовірностей і теорії оптимального управління.

Таким чином, рішення задач, пов'язаних з дослідженням ефективності систем контролю, передбачає вирішення таких питань: вибору показників ефективності; побудови математичної моделі контролю; дослідження корисності результатів виконання операцій; кількісної оцінки показників ефективності системи контролю, рисунок 4.12.

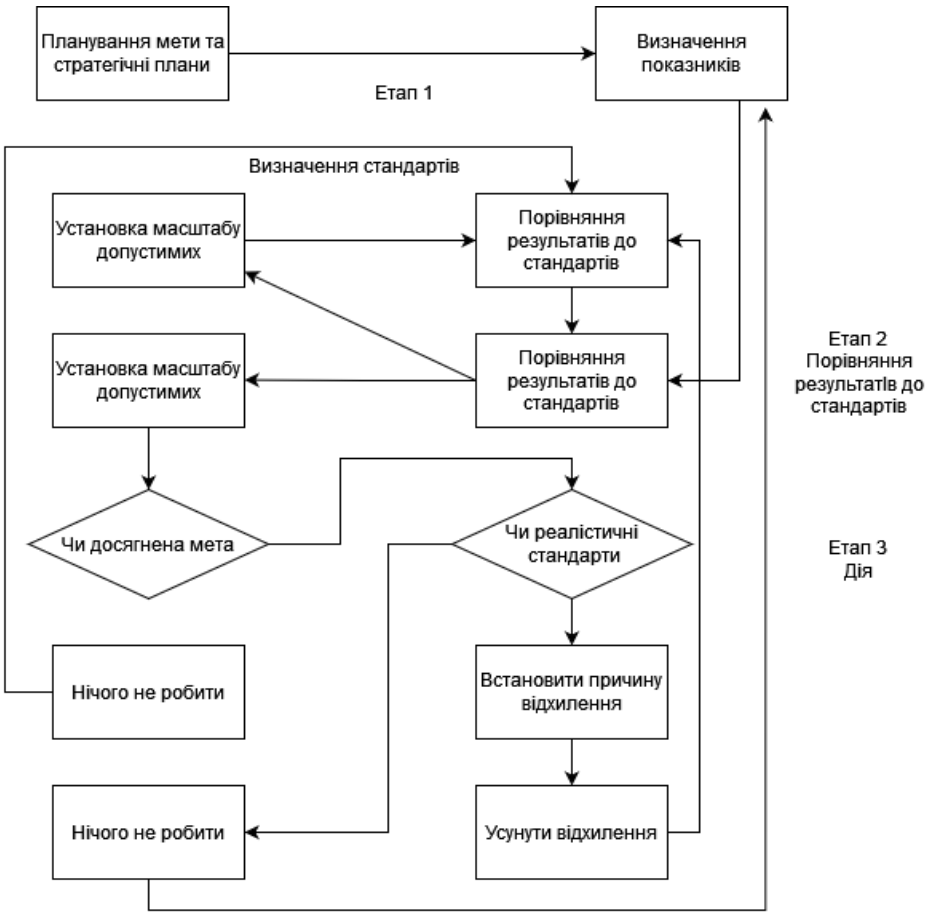


Рис. 4.12 Модель циклу управління якістю бортового обладнання відповідно до стандартів – алгоритм роботи процесу контролю

Отже, один із ключових аспектів у дослідженні операцій полягає у виборі такого показника ефективності, який би виявляв високу чутливість до параметрів операції, раціональні значення яких необхідно визначити, або які справляють найбільш істотний вплив на досягнення позитивного результату. Водночас цей показник має бути достатньо простим для забезпечення його зручного обчислення та подальшого аналітичного опрацювання

ІНС використовуються в складі КБО. Корекція ІНС в вихідному сигналі проводиться за допомогою алгоритмів оцінювання та прогнозу. Працездатність алгоритмів перевіряється за допомогою тестової моделі похибок ІНС [37].

$$x_k = \Phi x_{k-1} + W_{k-1}, \tag{91}$$

де:

$$x_k = \begin{bmatrix} \delta V_k \\ \varphi_k \\ \varepsilon_k \end{bmatrix}; \ \Phi = \begin{bmatrix} 1 & -gT & 0 \\ \frac{T}{R} & 1 & T \\ 0 & 0 & 1-\beta T \end{bmatrix}; \ \mathrm{W}_{k-1} = \begin{bmatrix} B \\ 0 \\ \omega_{k-1} \end{bmatrix}.$$

Тут – похибки ІНС у визначенні швидкості, φ_k – кути відхилення ГСП від площини горизонту, ε_k – швидкість дрейфу ГСП; g – прискорення вільного падіння; B – зміщення нуля акселерометра; R – радіус Землі; T – період дискретизації; β – середня частота випадкового зміни дрейфу; W_{k-1}-вектор зовнішніх збурень, що представляє собою дискретний аналог білого гаусового шуму.

Представлені результати математичного моделювання помилок ІНС, фільтра Калмана, алгоритму прогнозу. Для здійснення прогнозу похибок ІНС побудова моделі проводиться за допомогою алгоритму самоорганізації і ГА. Так як в результаті проведеного моделювання за допомогою цих алгоритмів отримані близькі по точності результати, представлені результати роботи тільки одного алгоритму.

На рис. 4.13 – 4.15 представлені результати роботи адаптивного фільтра Калмана.

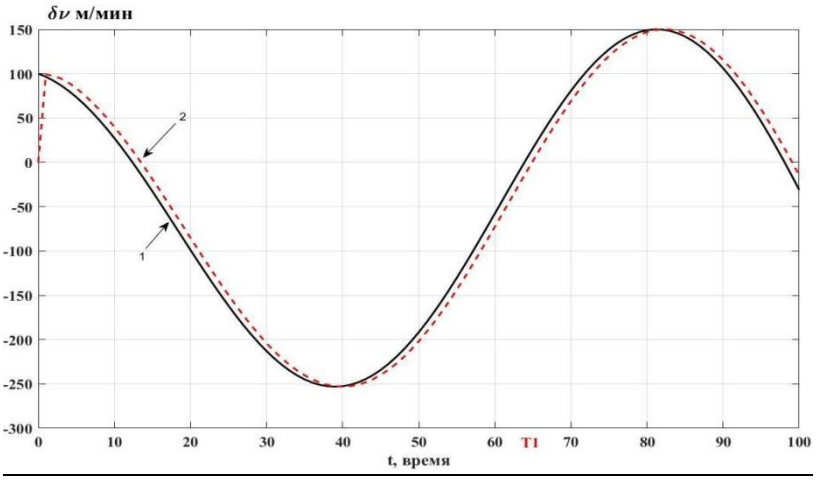


Рис. 4.13. Похибка ІНС у визначенні швидкості і її оцінка

1 – модель помилок ІНС у визначенні швидкості; 2 – оцінка швидкості за допомогою фільтра Калмана.

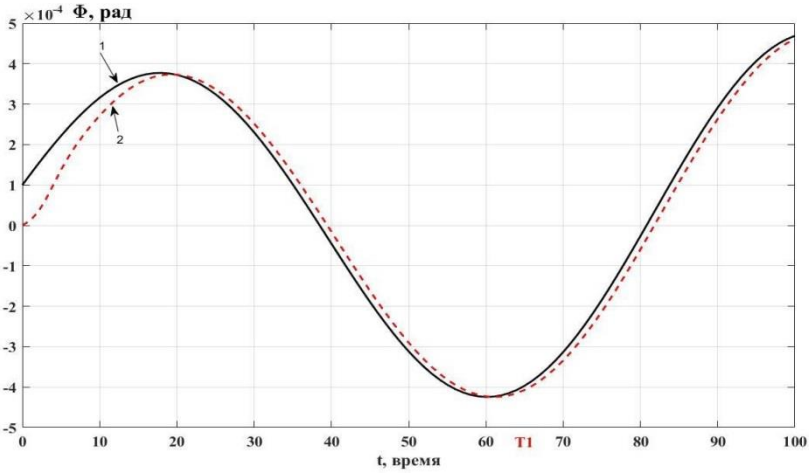


Рис. 4.14. Кут відхилення ГСП і його оцінка

1 – модель помилок ІНС у визначенні кута відхилення ГСП; 2 – оцінка кута відхилення ГСП за допомогою фільтра Калмана.

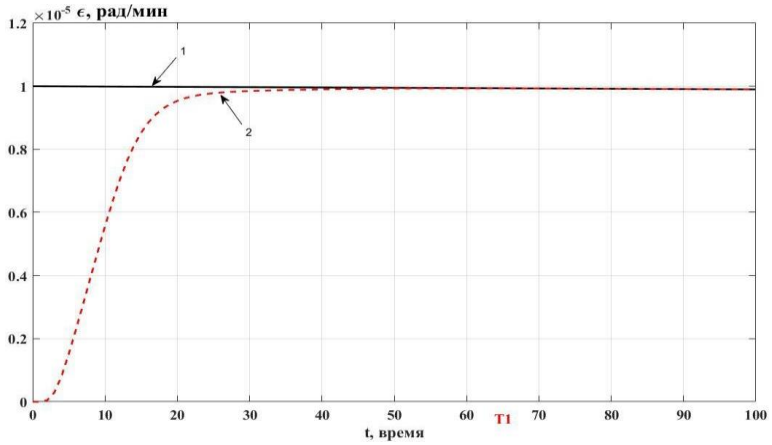
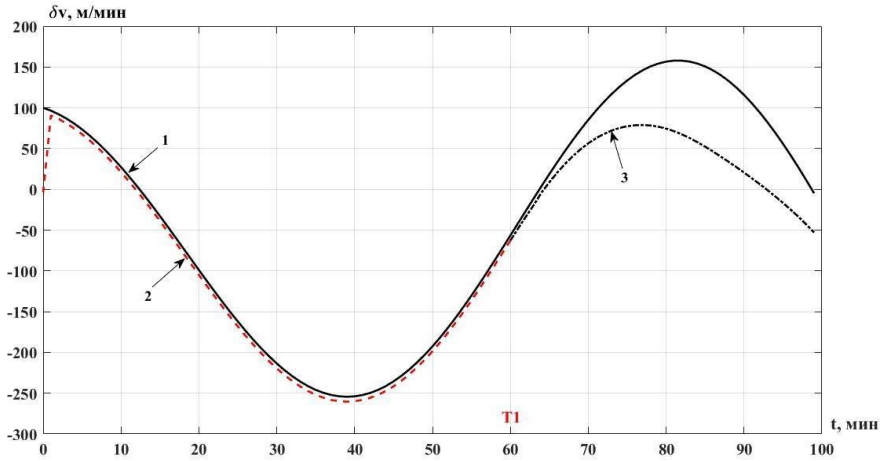


Рис. 4.15. Швидкість дрейфу ГСП і її оцінка

1 – модель помилок ІНС у визначенні дрейфу; 2 – оцінка дрейфу за допомогою фільтра Калмана.

У фільтрі Калмана використана апріорна модель похибок ІНС в формі . Застосовувати цю модель для прогнозу не представляється можливим, тому що в практичних додатках характер зміни похибок істотно змінюється, особливо при здійсненні ПС маневрів. Тому для здійснення прогнозу доцільно будувати модель в процесі польоту по наявній вимірювальній вибірці або вибірці оцінок похибок ІНС. На рис. 4.16 – 4.18 представлені результати моделювання похибок ІНС і прогноз по вибірці оцінок. Вибірка оцінок сформована на інтервалі 0-Т (0-60 хв.).



4.16. Похибка ІНС у визначенні швидкості і її прогноз

1 – модель помилок ІНС у визначенні швидкості; 2 – оцінка швидкості за допомогою ГА; 3 – прогноз за допомогою алгоритму ГА.

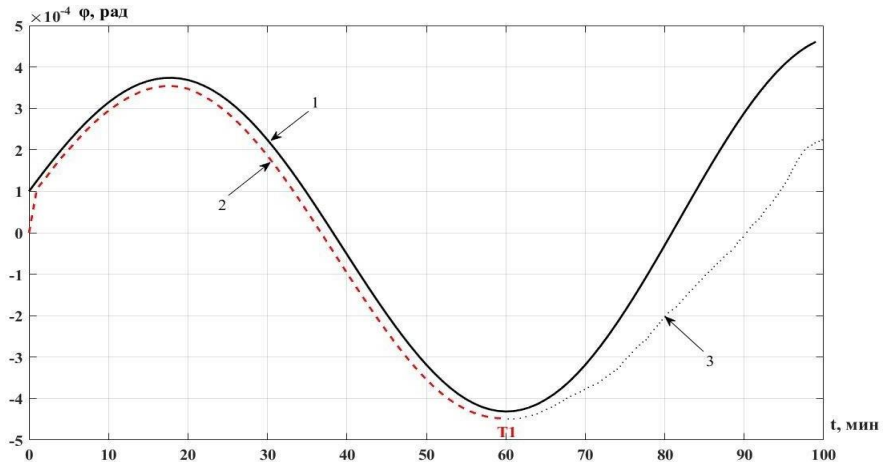


Рис. 4.17. Кут відхилення ГСП і його прогноз

1 – модель помилок ІНС у визначенні кута відхилення ГСП; 2 – оцінка кута відхилення ГСП за допомогою ГА; 3 – прогноз за допомогою алгоритму ГА.

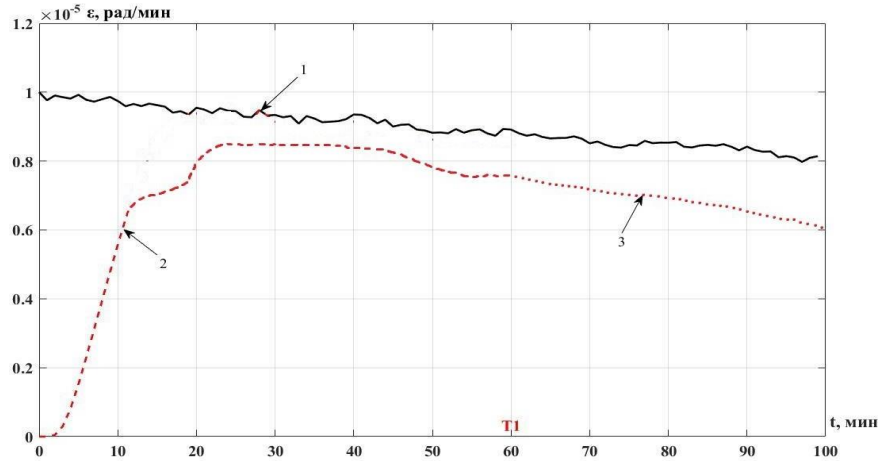


Рис. 4.18. Швидкість дрейфу ГСП і її прогноз

1 – модель помилок ІНС у визначенні швидкості дрейфу ГСП; 2 – оцінка швидкості дрейфу ГСП за допомогою ГА; 3 – прогноз за допомогою алгоритму ГА.

На рис. 4.19 – 4.21 представлені результати моделювання похибок ІНС і модель ГА, побудована на основі вимірювальної вибірки z. З 60 хв. здійснений короткостроковий прогноз.

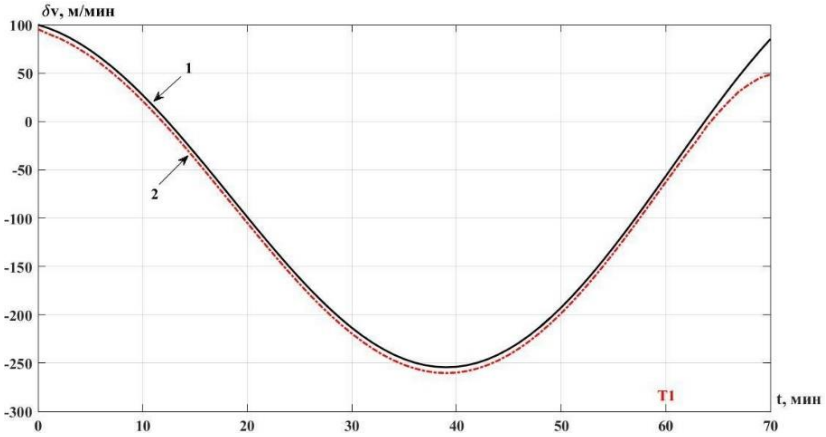


Рис.4.19. Похибка ІНС у визначенні швидкості і її короткостроковий прогноз

На рис. 4.20 позначено: 1 – модель помилок ІНС у визначенні швидкості; 2 – прогноз швидкості за допомогою ГА.

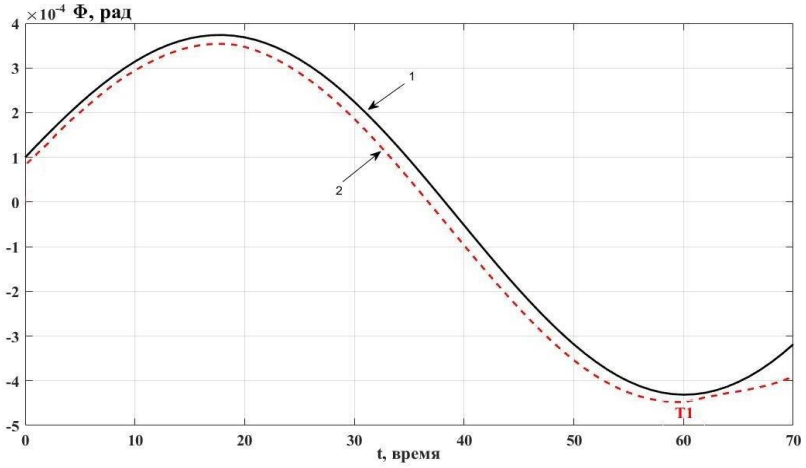


Рис. 4.20. Кут відхилення ГСП і його короткостроковий прогноз

На рис. 66 позначено: 1 – модель помилок ІНС у визначенні кута відхилення ГСП; 2 – прогноз кута відхилення ГСП за допомогою ГА.

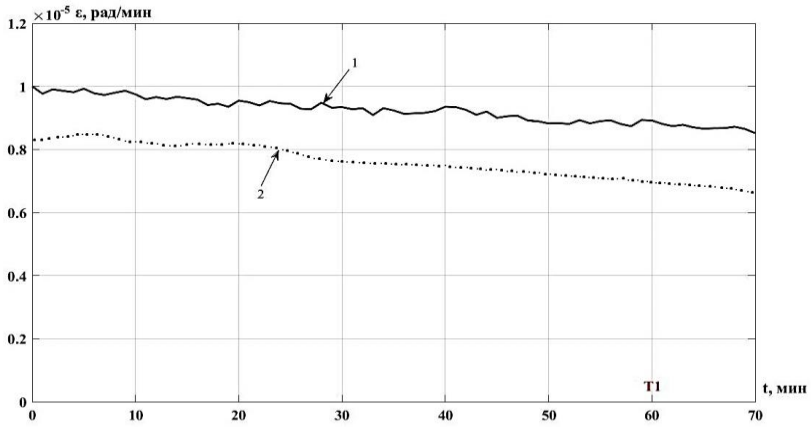


Рис. 4.21. Швидкість дрейфу ГСП і її короткостроковий прогноз

1 – модель помилок ІНС у визначенні швидкості дрейфу ГСП; 2 – прогноз швидкості дрейфу ГСП за допомогою ГА.

На рис. 4.22 – 4.24 наведені результати моделювання похибок ІНС, модель ГА, на основі оціночної вибірки і довгостроковий прогноз. Для побудови моделі використана коротка вибірка.

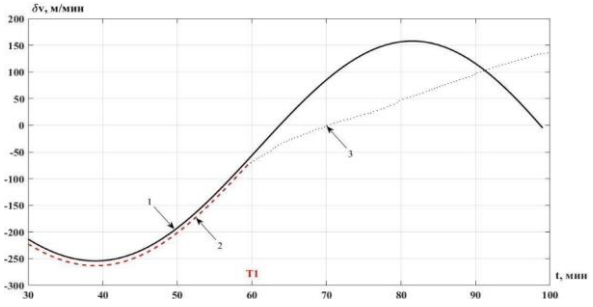


Рис. 4.22. Похибка ІНС у визначенні швидкості і її довгостроковий прогноз1 – модель помилок ІНС у визначенні швидкості; 2 – оцінка швидкості з ГА; 3 – прогноз.

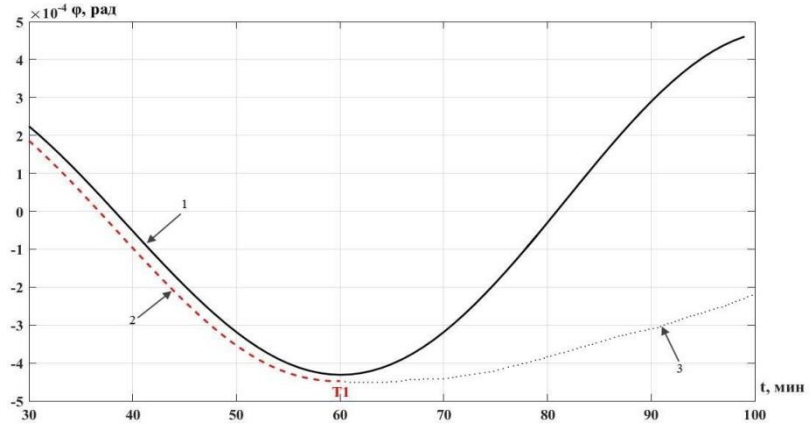


Рис. 4.23. Кут відхилення ГСП і його довгостроковий прогноз

1 – модель помилок ІНС у визначенні кута відхилення ГСП; 2 – оцінка кута відхилення ГСП за допомогою ГА; 3 – прогноз.

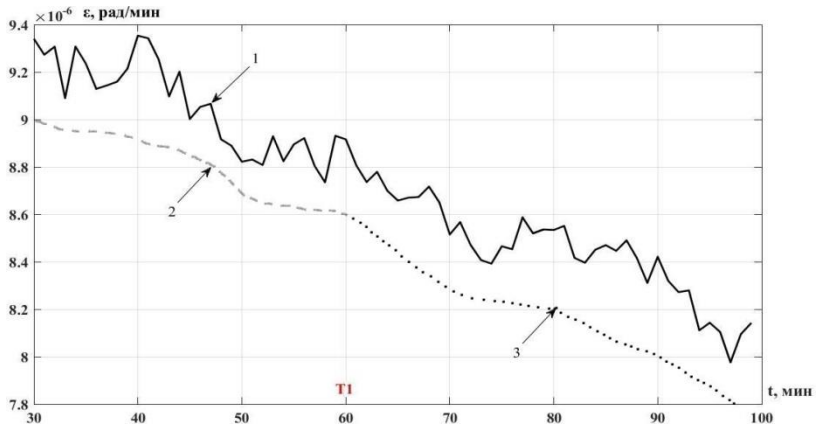


Рис. 4.24. Швидкість дрейфу ГСП і її прогноз

На рис. 70 позначено: 1 – модель помилок ІНС у визначенні швидкості дрейфу ГСП; 2 – оцінка швидкості дрейфу ГСП за допомогою ГА; 3 – прогноз.

На рис. 4.25 – 4.27 представлені результати моделювання похибок ІНС і короткостроковий прогноз по моделі, отриманої на основі оціночної вибірки.

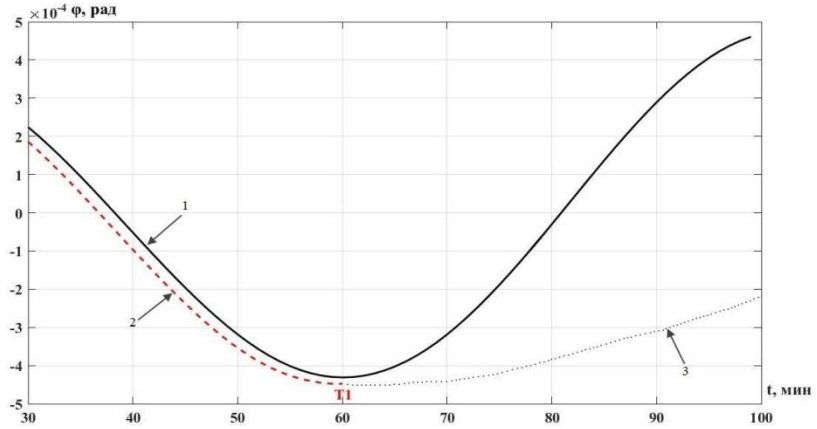


Рис. 4.25 Похибка ІНС у визначенні швидкості і її прогноз

1 – модель помилок ІНС у визначенні швидкості;

2 – прогноз швидкості за допомогою алгоритма ГА.

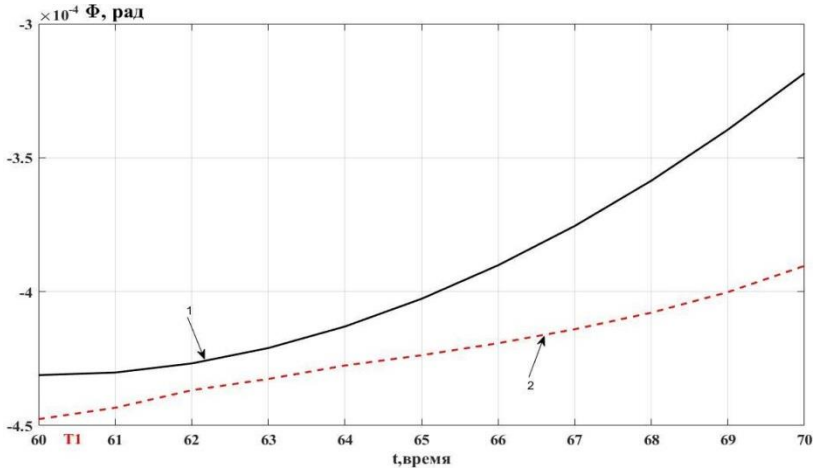


Рис. 4.27. Кут відхилення ГСП і його прогноз

1 – модель помилок ІНС у визначенні кута відхилення ГСП; 2 – прогноз кута відхилення ГСП за допомогою алгоритму ГА.

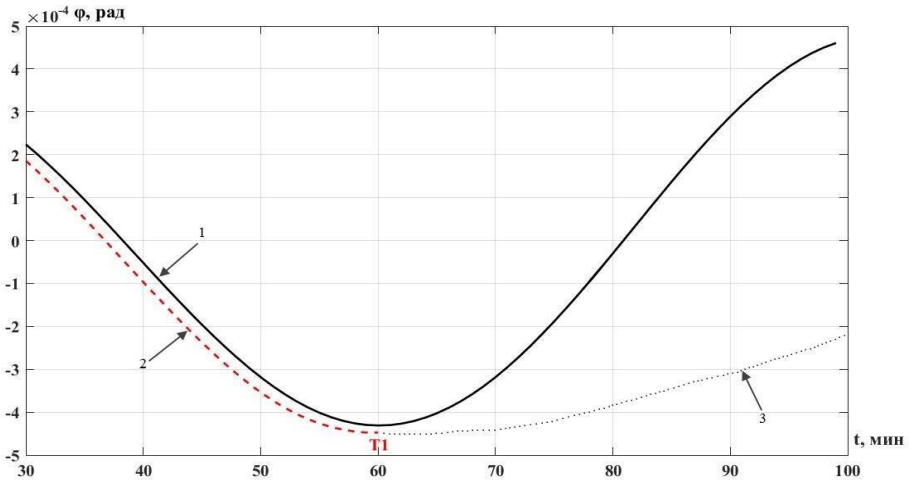


Рис. 4.27. Швидкість дрейфу ГСП і її прогноз

позначено: 1 – модель помилок ІНС у визначенні швидкості дрейфу ГСП; 2 – прогноз швидкості дрейфу ГСП за допомогою ГА.

Алгоритми функціонування ДЕС представлені на рисунку 4.28.

Подана схема ілюструє функціональну структуру системи формування, контролю та оцінки вихідних сигналів в контурі управління об’єктом на основі динамічної експертної системи (ДЕС). Вхідними впливами на систему виступають сигнали від датчиків, навколишнього середовища, зовнішні збурення та білий шум. На основі цих даних формується вектор вихідних сигналів, який надходить до об’єкта управління. За умови невдалого результату здійснюється корекція через виробітку нового управлінського впливу. Паралельно з цим, для оцінки стану об’єкта та адекватності реакції

системи, створюється модель похибок за допомогою генетичного алгоритму (ГА), яка передається в модулі ДЕС. У межах ДЕС результати оцінюються, порівнюються з математичною моделлю та базою знань, після чого використовуються для формування накопичуваного досвіду системи [70-74]. Отримані протоколи оцінок і знання дозволяють покращувати алгоритми управління й адаптувати систему до зовнішніх змін, рисунок 4.28.

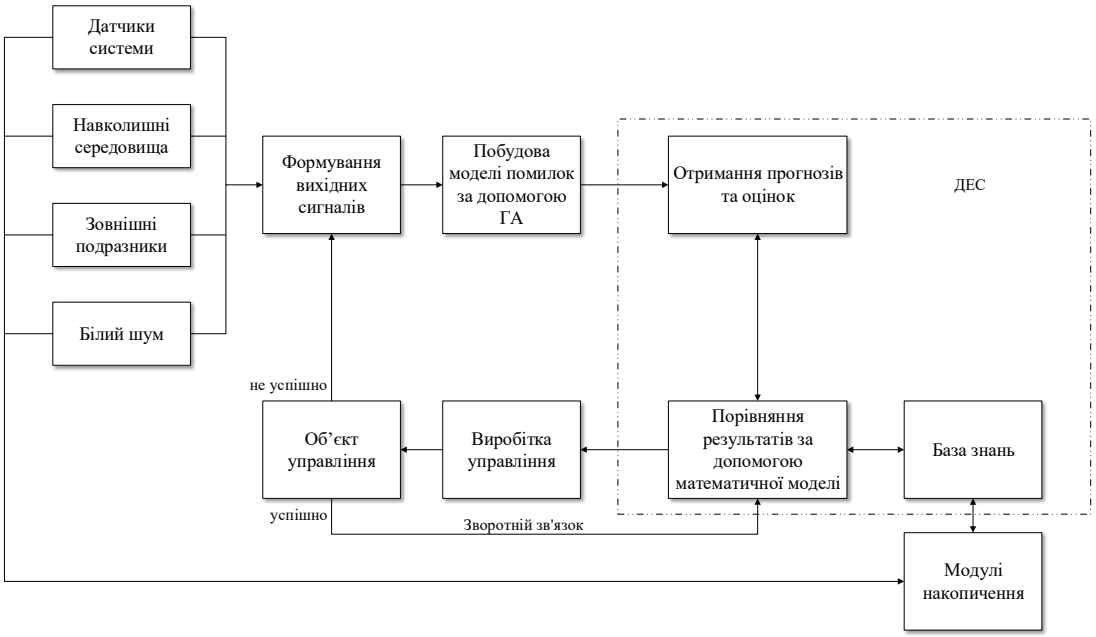


Рис. 4.28. Модель системи контролю та діагностики стану КБО на основі ДЕС з нечіткою логікою

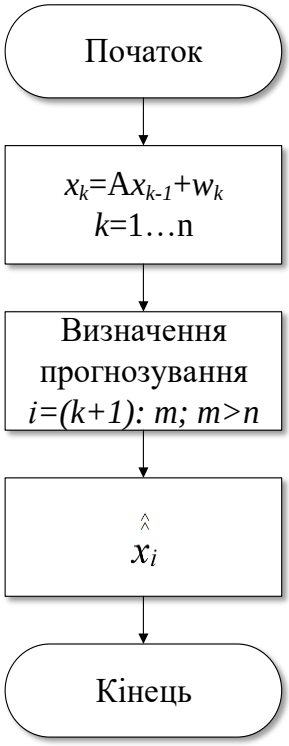


Рис. 4.29. Алгоритми прогнозу

Діагностична матриця конкретної КБО на обраному режимі є таблицею чисельних значень коефіцієнтів і, яка дозволяє за відхиленнями деяких вимірювальних параметрів визначати відхилення безпосередньо не вимірюваних параметрів стану , таблиця 4.1

Таблиця 4.1

Фрагмент діагностичної матриці ІНС

№	D_N_2	D_G_v	D_N_2	D_T_2	D_P_2	D_T_3	D_P_4	D_P_6	D_F_c	D_G_t	D_R	RESULT
1	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	ETALON
2	-0,18	0,12	0,16	0,14	0,00	-0,08	-0,33	-0,33	0,41	-0,03	-0,06	АКС 1%
3	-0,46	0,31	0,52	0,39	0,00	-0,27	-0,93	-0,93	1,11	-0,11	-0,18	АКС 3%
4	-0,68	0,57	0,85	0,65	0,00	-0,46	-1,57	-1,57	1,87	-0,19	-0,31	АКС 5%
5	0,19	0,07	-0,28	0,04	0,00	0,16	0,82	0,82	-0,66	0,33	0,49	АКС1 1%
6	0,78	0,24	-0,93	0,18	0,00	0,45	2,57	2,53	-2,08	1,03	1,54	АКС1 3%
7	1,16	0,39	-1,61	0,27	0,00	0,85	4,27	4,28	-3,60	1,76	2,62	АКС1 5%
8	-0,10	0,84	0,13	0,85	0,00	-0,07	0,53	0,51	0,26	0,72	1,02	АКС2 1%
9	-0,29	2,46	0,36	2,49	0,00	-0,15	1,58	1,58	0,79	2,16	3,01	АКС2 3%
10	-4,28	6,80	2,16	0,36	0,00	-1,16	-0,11	-0,14	4,55	3,21	5,14	АКС2 5%
11	-0,78	-0,06	0,06	-0,07	0,00	-0,03	-0,18	-0,18	0,11	-0,10	-0,15	АКС3 1%
12	-2,33	-0,20	0,16	-0,18	0,00	-0,08	-0,53	-0,53	0,30	-0,34	-0,47	АКС3 3%
13	-4,07	-0,37	0,25	-0,32	0,00	-0,13	-0,96	-1,00	0,55	-0,59	-0,84	АКС3 5%
14	0,68	0,21	-0,12	0,25	0,00	0,07	0,52	0,53	-0,28	0,31	0,46	АКС4 1%
15	2,04	1,20	-0,41	0,58	0,00	0,21	1,64	1,64	-0,93	0,94	1,38	АКС4 3%
16	3,41	1,43	-0,71	0,96	0,00	0,36	2,71	2,71	-1,8	1,62	2,30	АКС4 5%

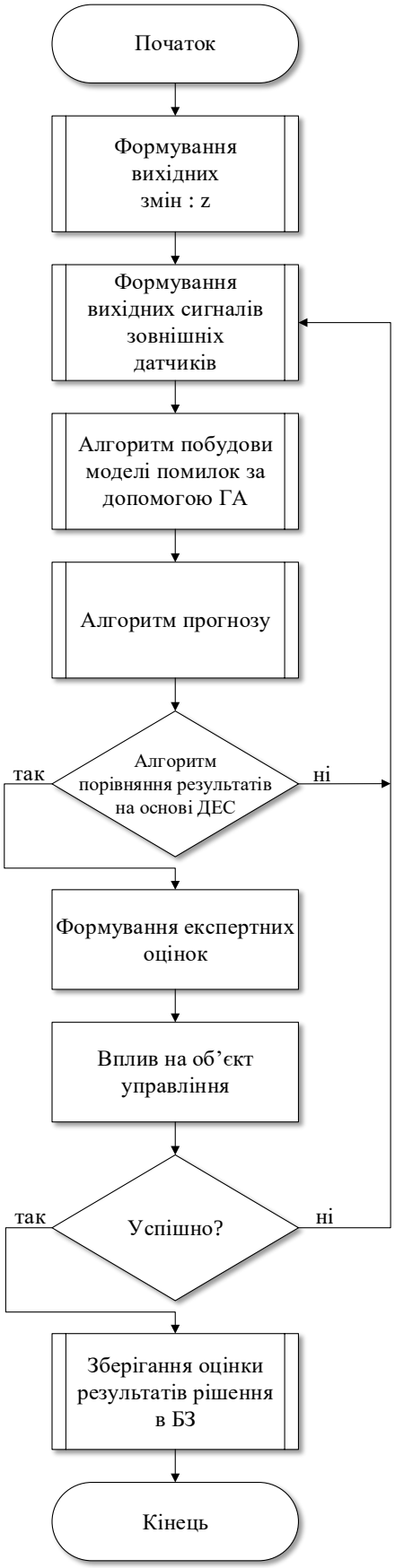


Рис. 4. 30. Основні алгоритми роботи

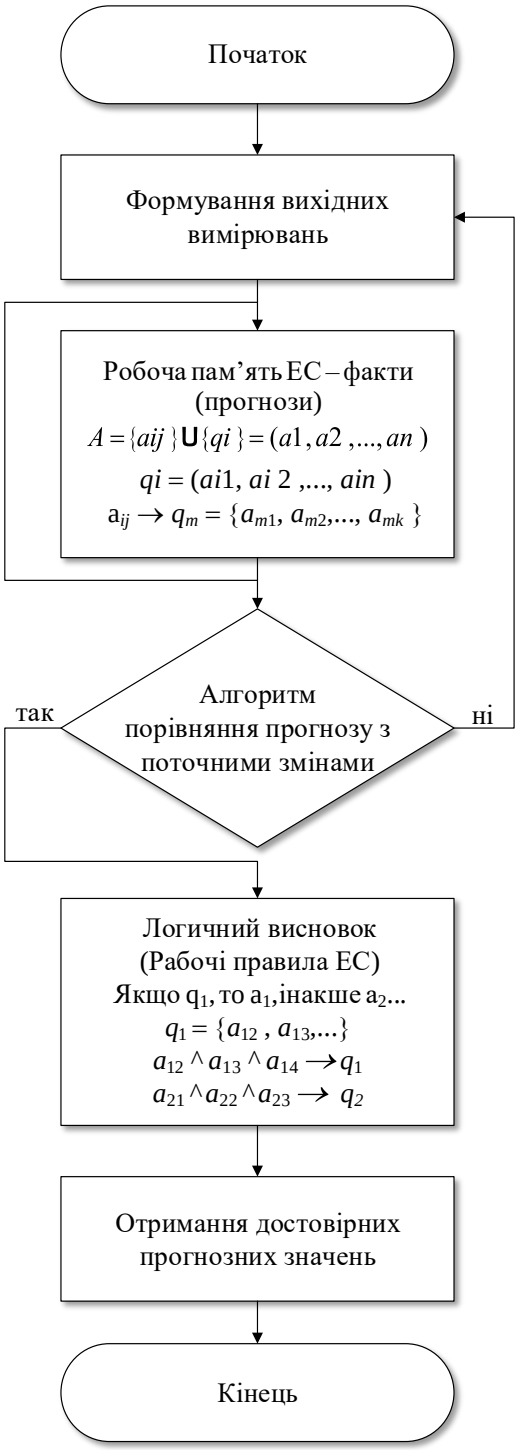


Рис. 4.31 Алгоритми поріняння за допомогою ДЕС

Нехай в процесі контролю технічного стану ІНС були заміряні наступні параметри: n_2 – похибка у визначенні широти місцевості (%); G_v – похибка у визначенні швидкості ПС v_x; T_2^* – похибка визначення курсу; P_2^* – оцінка швидкості дрейфу ГСП ε_x; T_3^* – похибка визначення тангажу; T_4^* – похибка визначення крену; P_4^* – оцінка швидкості дрейфу ГСП ε_y; F_c – похибка у визначенні довготи місцевості; G_t – похибка у визначенні швидкості ПС v_y; R – уявне прискорення.

У таблиці 4.2 представлені параметри відхилень і відповідні їм лінгвістичні змінні: *LN* – дуже мале; *MN* – невелике; *Z* – близько нуля; *MP* – середнє; *LP* – дуже велике. Процес тестування працездатності ДЕС перевіряються за тими рядками бази правил, які не увійшли до навчальної множини: по двох попереднім рядкам (з відхиленням параметрів стану ІНС в вузлах на 1 і 3% відповідно).

Таблиця 4.2

Фрагмент бази нечітких експертних правил

Атрибути і їх значення	**Результат**
Якщо	То =АКС
Якщо	То =АКС1
Якщо	То =АКС2
Якщо	То = АКС3
Якщо	То = АКС4

В процесі наповнення БД, крім поточних вимірювань, додатково використані прогнозні значення аналізованих параметрів КБО. ДЕС з алгоритмом побудови прогнозуючих моделей, методом діагностичних матриць і правил нечіткої логіки дозволяє визначати ступінь достовірності навігаційної інформації ПС, а блок прийняття рішень, регулятор відновлюють працездатність КБО, використання оцінок і прогнозних значень похибок ІНС підвищує достовірність навігаційної інформації ПС.

Представлені результати роботи алгоритму діагностики похибок ІНС. На рис 4.31. представлена похибка ІНС у визначенні швидкості, швидкість

дрейфу ГСП, оцінки адаптивним фільтром Калмана, а з моменту Т1 – прогноз, отриманий за допомогою ГА. При моделюванні для отримання похибки ІНС використана тестова математична модель. Так як в реальних умовах є інформація про оцінку, то вона використовується для діагностики стану КБО. Запропоновано використовувати прогноз похибки ІНС, щоб заздалегідь виявити момент виходу систем із зони стійкої роботи.

Застосування ДЕС значно підвищує ефективність діагностування КБО ПС. Оперативний аналіз різноманітної інформації про поточну ситуацію дозволяє приймати рішення про стан і перспективи даного КБО. Результат роботи системи контролю та діагностики КБО з функцією відновлення представлений на рис. 4.31..

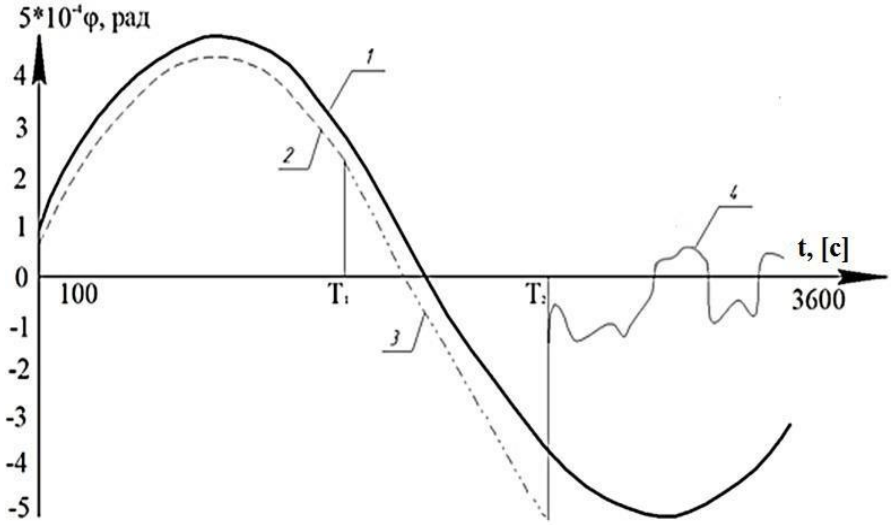


Рис. 4.31. Кут відхилення ГСП, його оцінка фільтром Калмана, прогноз і похибка ІНС з регулятором

позначено: 1 – математична модель зміни кута відхилення ГСП щодо супроводжуючого тригранника; 2 – оцінка кута відхилення ГСП адаптивним фільтром Калмана; 3 – прогноз кута відхилення ГСП; 4 – зміна кута відхилення ГСП при корекції в структурі ІНС.

На основі прогнозу похибок ІНС КБО ПС в момент часу Т2 приймається рішення про переключення робочого контуру корекції (на інтервалі Т1-Т2 використана корекція в вихідному сигналі ІНС, а з моменту Т2 використана корекція в структурі ІНС за допомогою регулятора). У порівнянні з ІНС КБО використання ІНС з розробленою системою контролю

в стохастичних умовах дозволяє підвищити точність за результатами математичного моделювання в середньому на 8-11%.

Досліджено особливості вирішення задач контролю і діагностики КБО з використанням нечіткої ДЕС, процес формування БД, а також реалізація FDI-методу. Побудована діагностична матриця ДЕС.

База нечітких правил сформована на основі діагностичної матриці, рядки якої лягли в основу створення БЗ і функцій приналежності відповідних лінгвістичних змінних.

Для 1% -го відхилення оцінки швидкості дрейфу ГСП, на основі бази нечітких правил ДЕС і операцію перетину нечітких множин. Тоді отримаємо:

$$\min = \begin{pmatrix} \mu_z(\Delta T_2^{(k)}) \,\& & \mu_z(\Delta P_4^{(k)}) \\ \&\mu_z(\Delta P_6^{(k)}) & \&\mu_z(\Delta F_c^{(k)}) \end{pmatrix} \to 0{,}85. \qquad (4.66)$$

Достовірність прийняття рішення про справності ІНС складає 0,85.

Для рядка діагностичної матриці, відповідної 3% відхилення оцінки швидкості дрейфу ГСП, отримаємо

$$\min(\mu_{\mathbf{Z}\in_i}) \to 0{,}59. \qquad (4.77)$$

Достовірність прийняття рішення про справність ІНС складає 0,59. При несправному стані ІНС коефіцієнт довіри становить 0,24. У розглянутій ситуації висока ймовірність виходу ІНС із зони стійкої роботи. Результати аналізу ДЕС збігаються з результатами досліджень, проведених в ході стендових випробувань ІНС.

Результати моделювання показали високу працездатність запропонованих алгоритмів. Аналіз результатів моделювання продемонстрував, що розроблене алгоритмічне забезпечення систем контролю і діагностики дозволяють підвищити ефективність навігаційних систем і КБО ПС за рахунок завчасного виявлення моменту, коли інформація стає недостовірною і утримання КБО в зоні стійкої роботи.

4.5. Висновки до четвертого розділу

Розроблена модель процесу проектування комплексу бортового обладнання інтегрованої модульної авіоніки для побудови математичних методів підвищення надійності та відмовостійкості, забезпечення якості та безпеки авіаційних систем. (Ця модель дозволяє систематизувати та формалізувати процес врахування аспектів надійності та відмовостійкості на ранніх етапах проектування, що дозволяє виявляти та усувати потенційні проблеми ще до виробництва та експлуатації. Результати застосування цієї моделі можуть бути корисними для розробників, інженерів та дослідників, що працюють у сфері авіаційних технологій, сприяючи підвищенню якості та надійності авіаційних систем та забезпецюючи безпеку польотів.

Розроблено структурна модель процесу проектування складних автоматизованих систем управління комплексу бортового обладнання є важливим інструментом для вирішення проблеми виявлення несправностей автоматизованих систем. Ця модель дозволяє систематизувати та візуалізувати взаємозв'язок між етапами проектування, аналізувати ситуації виявлення аналогій та помилок, а також ефективно повідомляти та формулювати звіти щодо виявлених проблем.

Розроблено структурно-логічна модель уніфікованого автоматизованого робочого місця для підвищення достовірності контролю обчислювальної системи інтегрованої модульної авіоніки є важливим кроком у напрямку забезпечення безпеки та надійності авіаційних систем. Ця модель дозволяє систематизувати та візуалізувати робоче місце, забезпечуючи операторам зручний інтерфейс для контролю та аналізу обчислювальної системи. Вона також дозволяє виявляти та усувати потенційні проблеми ще до виробництва та експлуатації, що покращує якість та надійність авіаційних систем.

Результати застосування цієї моделі можуть бути корисними для планування та оптимізації роботи бортових систем з урахуванням обмежень обчислювальної потужності та мережі передачі даних.

Розроблені методи оцінки ймовірності безвідмовної роботи структурно-логічної моделі уніфікованого автоматизованого робочого місця для контролю функціональних елементів обчислювальної інтегрованої модульної авіоніки на виробництві є важливим інструментом для забезпечення надійності та безпеки авіаційних систем. Ці методи дозволяють оцінити ймовірність виявлення та усунення потенційних проблем у роботі функціональних елементів ще до виробництва та експлуатації, сприяючи підвищенню якості та надійності авіаційних систем.

РОЗДІЛ 5. МОДЕЛЮВАННЯ РОБОТИ ЕЛЕМЕНТІВ І ПРИСТРОЇВ ІНТЕГРАЛЬНОЇ МОДУЛЬНОЇ АВІОНІКИ

Перевірка достовірності отриманих результатів є процес статистичного моделювання згідно з методом Монте-Карло. Крім того, розробка прикладного програмного продукту для аналізу процедур оброблення експлуатаційних даних є першим етапом для пошуку шляхів використання отриманих результатів дисертаційного дослідження. Програма для моделювання процедур оброблення проектування та експлуатаційних даних та процесів прийняття рішення для цього оброблення також може бути використана для пошуку оптимальних параметрів інформційної підтримки засобів проектування та експлуатації та вдосконалення.

Аналіз авіаційно-транспортної системи України показав, що в її організаційній структурі відсутні центри, основною діяльністю яких є оброблення експлуатаційних даних щодо визначальних параметрів та показників надійності засобів інформаційної підтримки, а також характеристик систем їх експлуатації. Наявність інформаційної підтримки є запорукою покращення ефективності систем на всіх етапах життєвого циклу повітряного судна.

Тому метою цього розділу є аналіз можливостей впровадження та шляхів використання отриманих результатів дисертаційного дослідження.

Для досягнення поставленої мети в цьому розділі були вирішені такі задачі: розглянута інформаційна технологія для підвищення відмовостійкості бортового обладнання; розробили алгоритм контролю обчислювальної системи інтегрованої модульної авіоніки під час польоту повітряного судна для контролю функціональних елементів та подальшої розробки програмного забезпечення у складі уніфікованого робочого місця обчислювальної системи інтегрованої модульної авіоніки; розробили, експериментально дослідили та впровадити в практику алгоритмічне, програмне забезпечення інформаційної підтримки процесів проектування та експлуатації бортового обладнання інтегрованої модульної авіоніки для верифікації створених методів та

моделей.

5.1. Оцінка надійності обчислювальних структур інтегрованої модульної авіоніки.

Суттєвими для оцінки надійності обчислювачів класу IMA є внутрішня структура обчислювача [62-64], в якості якої розглядається схема зв'язку ФМ у виробі, і λ-характеристики надійності елементної бази ФМ, що входить у виріб. При цьому оцінка надійності виробу може бути виконана шляхом аналітичного виведення виразу для ймовірності $P(t)$ безвідмовної роботи виробу і побудови сімейства графіків залежності цієї ймовірності на заданому часовому інтервалі. Часовий інтервал характеризує очікуваний час безперервної роботи виробу в експлуатації.

На основі універсальної функціональної схеми обчислювача класу IMA на практиці розробляють різні варіанти внутрішніх структур виробів авіоніки, що реалізуються на практичному рівні, рисунок 5.1.

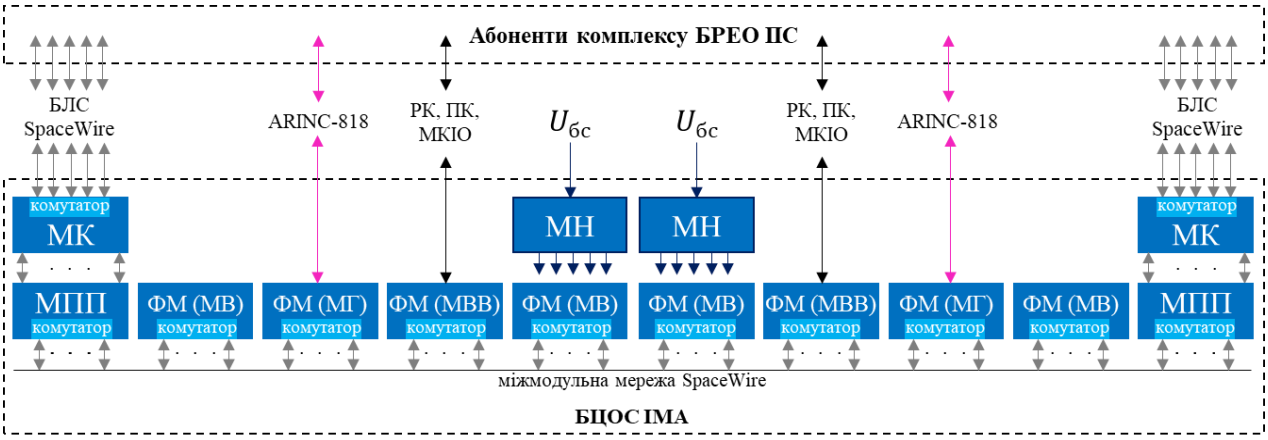


Рис 5.1 Структурно-функціональна схема міжмодульної взаємодії комплексу бортового радіоелектронного обладнання (БРЕО) на основі мережі SpaceWire в архітектурі IMA

Необхідно відзначити, що фізичне середовище поширення інформації в виробі відповідає схемі зв'язку, а логічне середовище поширення інформації організовується програмними засобами. Таким чином, не кожна фізична лінія передачі інформації виявляється задіяною в логічному протоколі обміну в ОС.

Оскільки, інтегрована модульна авіоніка – це найсучасніша систем

авіоніки для новітніх ПС. Система IMA складається зі стандартизованого апаратного забезпечення для обчислень та вводу/виводу, а також а також спільної мережі з високою пропускною здатністю, рисунок 5.2.

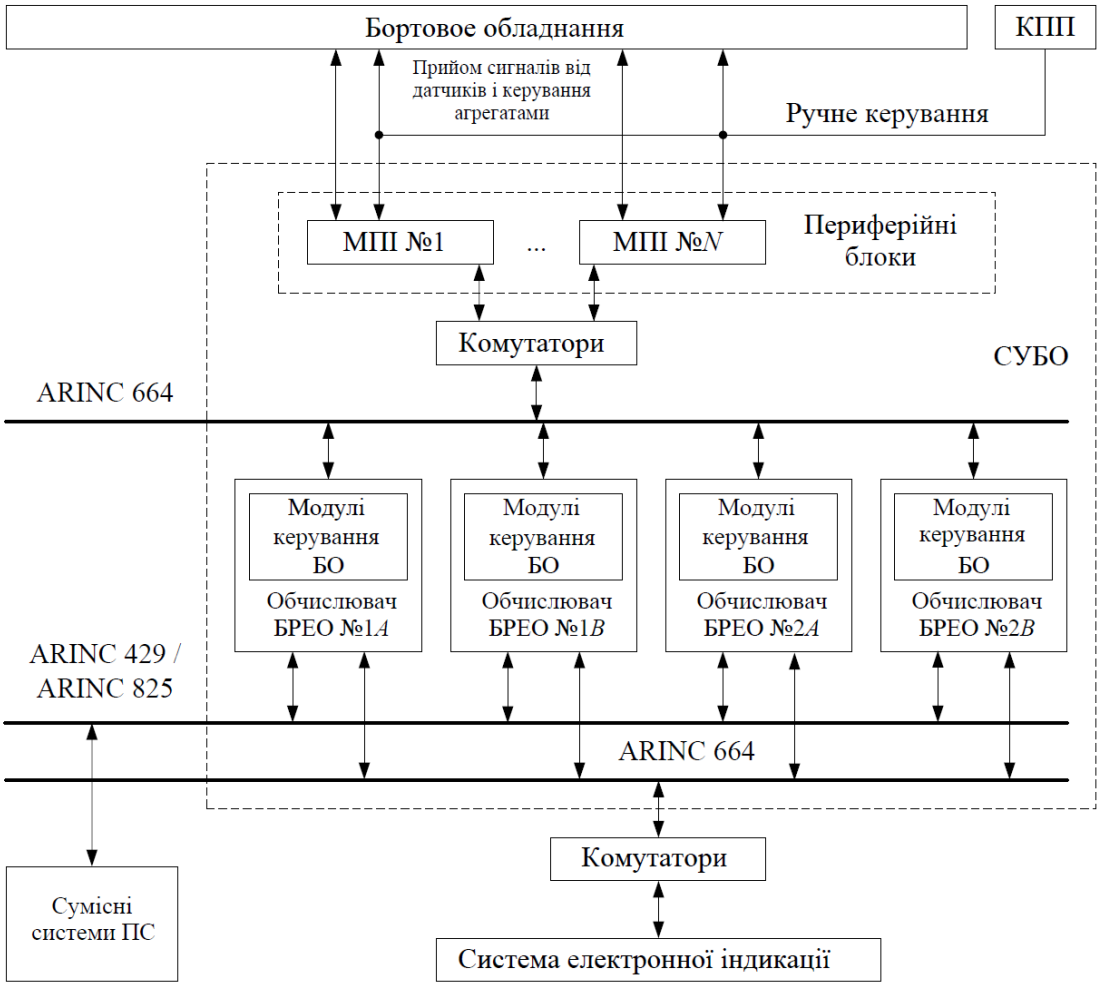
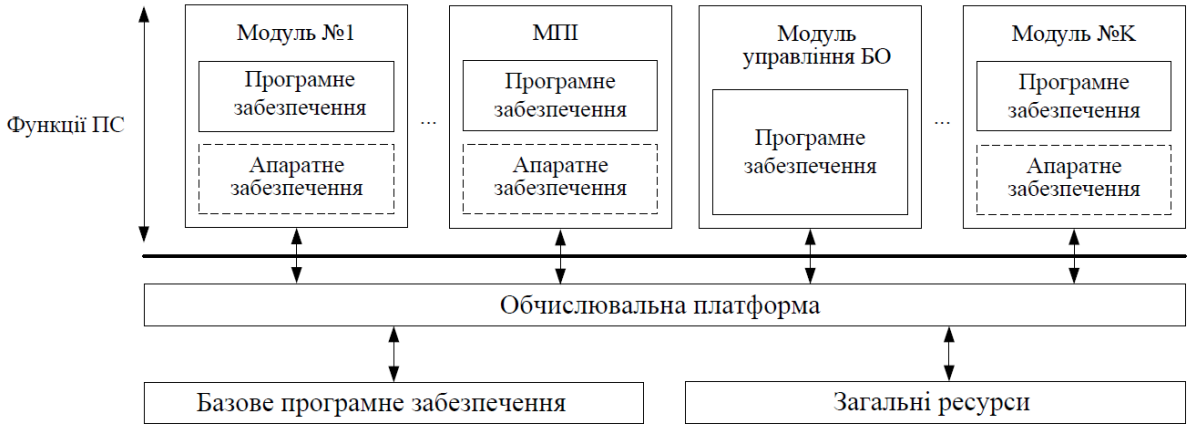


Рис. 92 – Архітектура обчислювальної магістралі IMA з використанням протоколів ARINC у СУБО ПС

Ресурси апаратного забезпечення розподіляються між функціями системи ПС, тобто IMA – це єдина система авіоніки для декількох функцій авіоніки. Її призначення визначається розподілом функцій і конфігурацією [7]. Спільне використання меншої кількості пристроїв та спільне використання меншої кількості пристроїв і меншої кількості типів пристроїв роблять системи IMA кращими за вагою і вартістю порівняно з традиційними системами авіоніки [12]. Друге покоління платформ IMA, так звані розподілені IMA, збільшують потенціал економії за рахунок розділення обчислень та вводу/виводу, а також за рахунок просторово розподіленого встановлення пристроїв IMA [3].

Однією з проблем розробцки IMA – є систем авіоніки, так званих IMA-архітектур, є складність, пов'язана зі спільним використанням ресурсів і просторовим розподілом [14, 16]. В даний час процесори, пам'ять і входи/виходи використовуються приблизно 1000 окремими функціями і периферійними компонентами, такими як датчики і виконавчі механізми. Сучасні архітектури включають близько 50 модулів IMA, а також до 1000 місць для встановлення модулів та периферійних пристроїв. Об'єднання систем, обладнання та анатомії, як показано на рисунку 5.3, є основним завданням при проектуванні при плануванні систем IMA.



Рису. 94 – Структурна схема обчислювальної платформи відповідно до концепції розподіленої IMA

Більше того, оскільки кількість функцій і можливостей електронних пристроїв безперервно зростає [6], кількість можливих варіантів архітектури варіантів архітектури стрімко зростає. Крім того, до систем висуваються вимоги щодо безпеки та до систем висуваються вимоги до безпеки та продуктивності, а також зростає важливість складних суперечливих економічних цілей проектування. На сьогоднішній день IMA-архітектури досягли такої кількості об'єктів, зв'язків, вимог і завдань, які змушують інженерів боротися з пошуком обґрунтованої і, особливо, оптимальної архітектури. Пряме і цілеспрямоване проектування систем, також відоме як системна архітектура [7], навряд чи можлива вручну. Процес ручного проектування – це ітеративна процедура спроб і помилок. В кінці процесу проектування гарантується правильність архітектури, але її оптимальність невідома. Цю ситуацію можна було б можна

покращити за допомогою методів автоматизованого проектування.

Комп'ютерне проектування архітектур IMA є активним дослідницькою областью. Дослідження в основному поділяються на моделювання, розподіл функцій та маршрутизація сигналів. Підходи до моделювання IMA – архітектур з метою імітаційного моделювання та верифікації можна знайти в [34, 35], [40], [41-43], [52] та [57]. Найчастіше ці підходи базуються на AADL [111 –113] або SysML [114]. Обидва вони видаються недостатньо жорсткими та повними для цілісної алгоритмічної підтримки. Оптимальний розподіл функцій є традиційною проблемою в інформатиційній підтримці. Ці підходи варіюються від розподілу функцій до розподілу функцій до розподілу надмірності. Взаємозв'язки різних автоматизацій зазвичай не розглядаються. Крім того, розмір, для якого демонструються методи для яких демонструються методи, як правило, не перевищує 20 функцій. Важливим є призначення сигналів до загальної системи шин. Зв'язок між розподілом функцій та розподілом мережі розподілом мережі не розглядається, а демонстраційні сфера застосування знову ж таки невелика.

Представлений фреймворк підтримує процес проектування IMA за допомогою інжинірингу на основі моделей та математичної оптимізацією на основі моделей. Таким чином, вона представляє нову модель для конкретної галузі, особливо для планування архітектури авіоніки. Крім того, методи оптимізації архітектури з різних оптимізації архітектури з різних галузей, які базуються на математичному фундаменті, що дозволяє здійснювати глобальну оптимальну та багатоцільову оптимізацію. Крім того, представлено абсолютно нові підходи до визначення розмірів пристроїв, генерації топології та повної генерації архітектури. Найважливіше те, що модель і оптимізація інтегровані в єдиний гнучкий фреймворк.

Базовою основою архітектурної структури авіоніки є специфічна модель, розроблена для планування архітектури авіоніки. Планування IMA-архітектури – це паралельний процес [32-35]. Фактична архітектура розробляється інженером-конструктором авіоніки. Планування, однак, сильно залежить від

функцій, які потрібно розмістити, периферійних пристроїв, які потрібно підключити, доступного обладнання та анатомії ПСка. Функції, обладнання ІМА та анатомія ПСка розробляються паралельно. Тому дуже важливо якомога краще розділити систему, апаратне забезпечення та систему обладнанняструктури. Більше того, на ранніх стадіях проектування вимоги можуть швидко змінюватися і потрібно порівняти кілька варіантів архітектури.

Моделювання цих вимог повинно бути достатньо жорстким щоб забезпечити автоматичну перевірку. Більше того, атрибути повинні бути змодельовані, які дозволяють проводити попередню оцінку архітектур і варіантів щодо маси або вартості.

Розроблена модель є статичною моделлю систем, обладнання та інсталяцій. Статична означає інваріантне в часі представлення функцій, сигналів і розподілу ресурсів. Структура верхнього рівня моделі зображена на рис. 94. Основними трьома рівнями є системи, обладнання та інсталяція. Вони майже незалежні. Однак, всі три рівні побудовані на одних і тих же компонентах з шару визначень, наприклад, ресурси або типи пристроїв. Архітектура операційної авіоніки складається з шару відображення етапів. Він інтегрує елементи з моделей систем, обладнання та програмного забезпечення. Може існувати декілька різних відображень. При цьому забезпечується високе повторне використання елементів моделей і легке порівняння варіантів архітектури. порівняння варіантів архітектури. Рівень сценаріїв включає параметри для оцінки, рисунок 5.4.

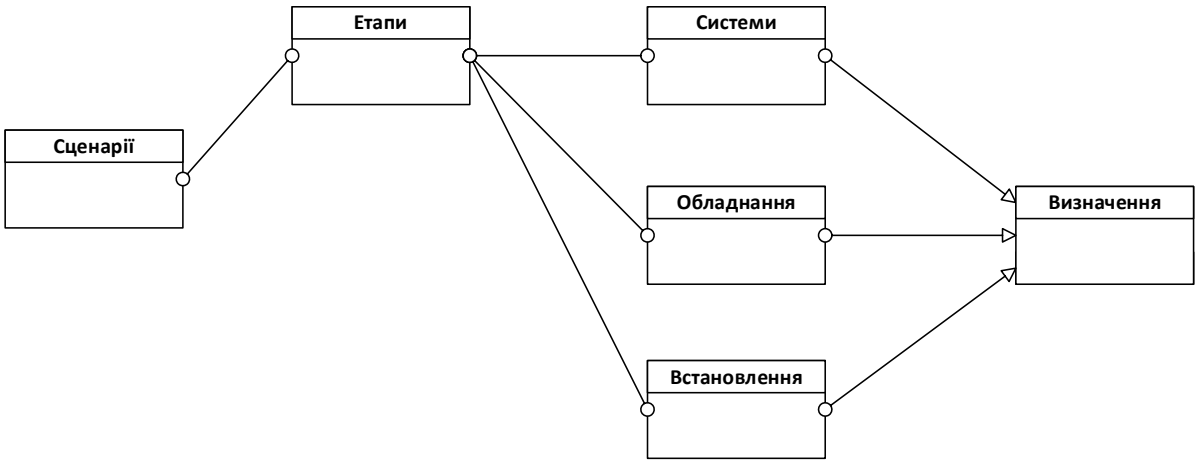
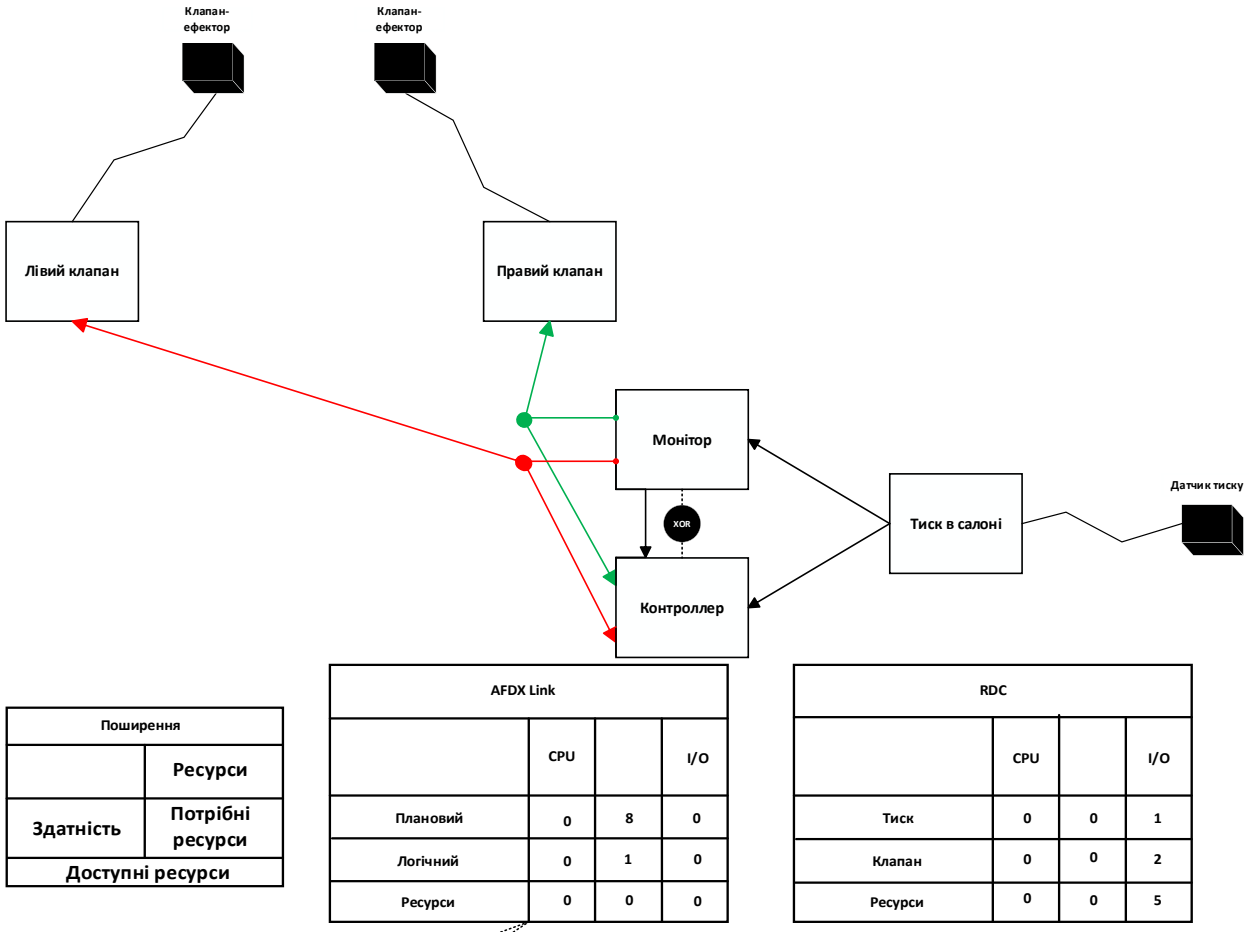


Рис. 5.4: Діаграма класів основних шарів авіоніки моделі архітектури.

При розгляді IMA основними компонентами систем є програмні функції, датчики і виконавчі механізми. Крім того, функції та периферійні пристрої обмінюються даними. Функції системи, входи/виходи для підключення периферійних пристроїв і комунікаційна магістраль повинні бути забезпечені апаратним забезпеченням IMA. Тому можемо представити структурну систему моделювання, як набір задач, що обмінюються сигналами, рисунок 5.5.



Поширення	
	Ресурси
Здатність	Потрібні ресурси
Доступні ресурси	

AFDX Link			
	CPU		I/O
Плановий	0	8	0
Логічний	0	1	0
Ресурси	0	0	0

RDC			
	CPU		I/O
Тиск	0	0	1
Клапан	0	0	2
Ресурси	0	0	5

CPM			
	CPU	Пропускна здатність	I/O
Контроллер	10	0	0
Монітор	10	0	0
Ресурси	90	0	0

Перемикач			
	CPU	Пропускна здатність	I/O
Плановий	1	8	0
Логічний	1	0	0
Ресурси	100	0	0

Рис. 5.5. Модель структурної системи апаратного забезпечення IMA

Як завдання, так і сигнали визначають необхідні ресурси. Ресурс – це зліченна кількість, яка має бути надана апаратне забезпечення хостингу. Завдання або сигнал вказує на те, скільки певного типу ресурсу споживається на хостингу. Прикладами ресурсів є потужність процесора в MIPS, пам'ять в МБ, пропускна здатність в Мбіт/с, кількість аналогових інтерфейсів або кількість шин CAN. Ця модель ресурсів дозволяє проводити серйозну перевірку. Після того, як системи призначені на апаратне забезпечення, ресурси жодного з пристроїв не повинні бути перевищені. Однак, ресурси не є достатніми для вираження операційних, безпекових та продуктивних потреб систем в експлуатації, безпеці та продуктивності. Тому було визначено вісім загальних додаткових обмежень, достатніх для того, щоб охарактеризувати всі функціональні потреби.

Периферійні обмеження пов'язують функції з периферійними пристроями. Під час роботи необхідно переконатися, що функція, яка потребує периферійного пристрою, підключено до периферійного пристрою, для цього потрібен інтерфейс вводу/виводу, а також проводки. Останнє призводить до збільшення ваги та вартості.

Обмеження пристрою, обмежують можливе відображення функції на набір пристроїв. Як варіант, набір пристроїв може бути виключений для функції, хоча ресурси можуть бути доступними.

Обмеження щодо місця встановлення. Заборонити або примусово встановлювати певні місця для функцій, тобто пристрій, на якому розміщено функцію, не повинен знаходитися в одному із зазначених місць. Наприклад, зона розриву ротора може бути виключена для функції.

Обмеження живлення визначають потужність живлення, яке пристрій повинен або не повинен мати, якщо функція функція відображена на пристрої. Наприклад: функції, критичні для безпеки, можуть бути протиставлені лише на пристроях, підключених до шини аварійного живлення.

Обмеження сегрегації забороняють дві або більше функцій на одному обладнанні. (це обмеження для надлишкових каналів системи).

Обмеження сегрегації за місцем розташування забороняють не лише одному пристрою, але й усім пристроям в одному місці для відображення двох функцій. Це пов'язано з міркуваннями просторової безпеки.

Обмеження відмінності застосовуються до функцій, які з міркувань надмірності мають бути зіставити з різними типами пристроїв.

Обмеження затримки можуть бути протиставленні до ряду функцій і сигналів, які формують критичний за часом шлях. Передбачається, що цей ланцюжок виконується періодично, і що кожне виконання має бути швидшим ніж заданий максимальний час виконання, при відображенні на апаратне забезпечення. Таким чином, час виконання і час передачі для сигналів і завдань на різних типах апаратного забезпечення повинні бути відомі з точки зору найгіршого часу виконання.Платформою для виконання функцій системи є апаратне забезпечення IMA.

Воно моделюється на апаратному рівні. Апаратне забезпечення виражається загальними пристроями та зв'язками. Воно не залежить від певної технології, кожен пристрій відповідає типу обладнання. Типи пристроїв зберігаються у межах визначень. Тип пристрою визначає фізичні характеристики, такі як вага, розміри або надійність. Крім того, тип пристрою визначає ресурси, доступні для розміщення завдань. В основному, вказується кількість і тип доступних ресурсів.

Оскільки надання ресурсів є неоднозначним, модель ресурсів поширюється для входів/виходів з блискавкозахистом або рівнями струму, існує мінімальна вимога, але всі входи/виходи з вищими рівнями можуть використовуватися як альтернативи. Для кожного типу пристрою, відповідно задачі є споживання ресурсів. Для кожного типу може бути кілька можливостей з різним споживанням ресурсів на одному типі пристрою. З'єднання – це зв'язок точка-точка між пристроями. Зв'язки надають ресурси та можливості, як і пристрої. Якщо з'єднання зв'язок є комутованою мережею або шиною, то комутатор або шина представлена додатковим пристроєм. Приклад основних обчислювальних модулів (CPM), віддалених концентраторів даних (RDC) і

мережі AFDX наведено на рисунку 5.6.

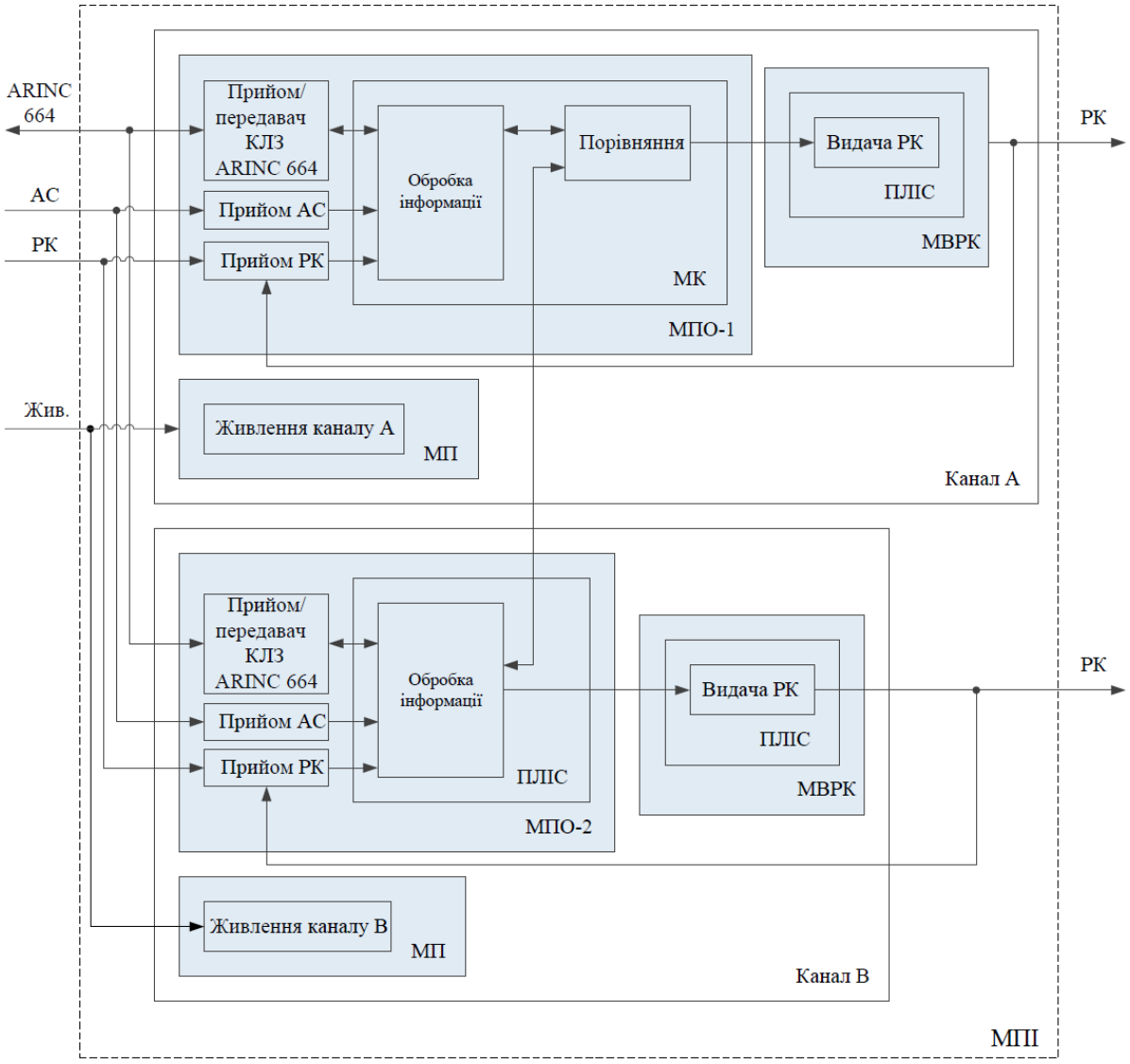


Рис. 5.6 Архітектура модулів перетворення інформації

На ньому показано ресурси, доступні на пристрої і в мережі, а також типи з'єднань а також ресурси, що споживаються при відображенні завдання контролера, монітора, тиску або клапана; або плаваючого чи булевих сигналів.

Конструкція або анатомія ПС є контейнером для авіоніки. Потрібен простір для встановлення та інфраструктура, така як фіксація, охолодження та живлення. Доступна інсталяційна структура моделюється в шарі інсталяції. Приміщення для пристроїв є місцями інсталяції. Типи пристроїв визначають кількість необхідних ресурсів необхідних ресурсів. У правильній архітектурі ресурси локацій не повинні перевищувати ресурси інфраструктури необхідними для встановлених пристроїв. Лінії зв'язку та периферійні дроти

можуть знаходитись всередині локації. З'єднання між місцями інсталяції моделюються за допомогою кабельних трас та з'єднаннями. Кабельна траса – це з'єднання "точка-точка" з фіксованою фіксованої довжини. Кабельні траси можуть розділятися і з'єднуватися на з'єднаннях кабельних трас. Траси і з'єднання утворюють топологію анатомію інсталяції. Довжину кабельних трас можна можуть бути співвіднесені з 3D-координатами, але це не обов'язково. Приклад трьох місць інсталяції, їхніх інфраструктурних ресурсів та з'єднань показано на рисунку 5.5.

Архітектура авіоніки працює тільки тоді, коли завдання зіставлені з пристроями, сигнали з каналами зв'язку, пристрої з місцями розташування, а з'єднання – до кабельних маршрутів. Це моделюється в шарі відображення шарі. Одне відображення посилається на один системний рівень, один апаратний рівень і один інсталяційний рівень. Для кожного елемента цих шарів існує об'єкт відображення у шарі відображення шарі відображення існує об'єкт відображення. Ці об'єкти прив'язки розташовані ієрархічно, тобто прив'язка завдання є наслідком прив'язки пристрою, який є наслідком прив'язки місця розташування. Призначення сигналів і проводки розбиваються на кілька сегментів вздовж їхнього маршруту. Кожен сегмент призначається ланці, пристрою або кабельному маршруту. Завдяки такій ієрархії відображення безпосередньо представляє реалізацію певної архітектури. Всі перевірки та оцінки виконуються на карті. Більше того, на одну і ту ж систему, апаратне забезпечення та елементи інсталяції можна посилатися в декількох інших мапах для вираження альтернативних варіантів. Крім того, зміни в системи, обладнання та інсталяції безпосередньо поширюватимуться на всі відображення, що на них посилаються.

Архітектор авіоніки Architect– це середовище планування для архітектури авіоніки. Він підтримує моделювання, верифікацію, оцінку та порівняння, як описано вище. Це реалізація на основі Eclipse, що використовує формальну мову моделювання даних мову формального моделювання даних ECORE для доменної моделі архітектури. Модель розширено гнучким фреймворком для

верифікації та оцінювання. Як верифікація, так і оцінювання можуть бути розширювати за допомогою довільних користувацьких правил та цілей. Правила та цілі оцінюються під час моделювання, рисунок 5.7.

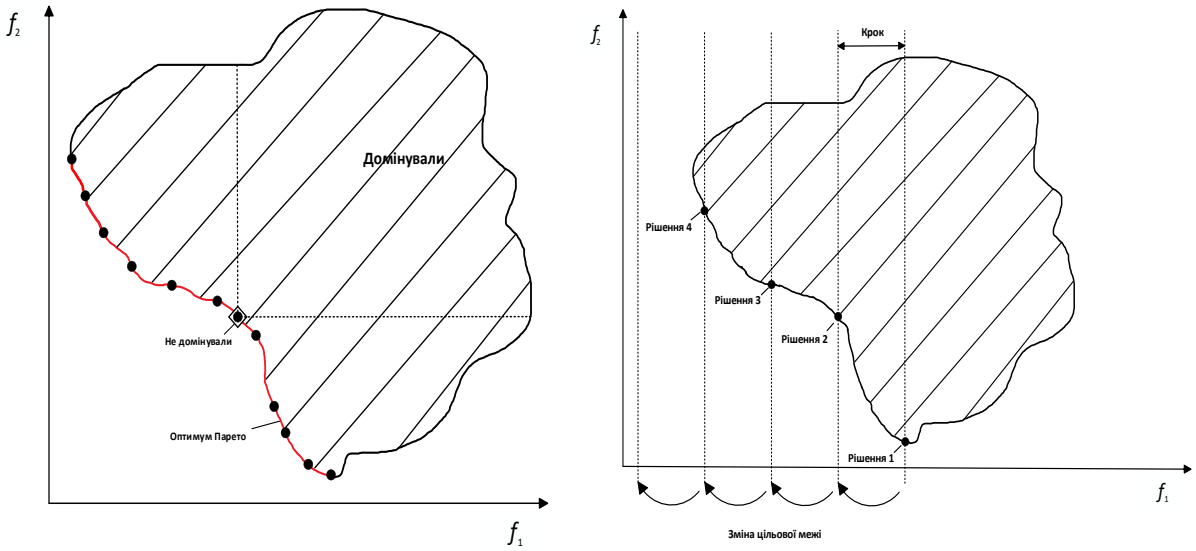


Рис. 5.7. Вибірка для двох мінімальних цілей за Парето

Моделювання здійснюється в ієрархічному редакторі або в спеціальних 2D-візуалізації для різних аспектів. 2D-редактори існують для визначень, систем, обладнання, інсталяції та мапування. Редактори та подання забезпечують безперешкодний інтерфейс до оптимізаційних процедур оптимізації, які описані нижче.

Під час проектування архітектури авіоніки виникає декілька з'являється декілька проблем проектування, де людина може втратити загальний огляд. Це питання призначення, де кількість елементів, що призначаються, або кількість цілей, або кількість перехресних зв'язків є великою, наприклад, призначення всіх завдань до модулів IMA. Для цього необхідно врахувати ресурси та обмеження приблизно 1000 завдань. Відображення може бути отримано і перевірено вручну. Якщо це відображення є оптимальним, особливо щодо складних завдань, оскільки витрати на обслуговування часто невідомі. Більше того, інженери-проектувальники стикаються з суперечливими з суперечливими завданнями проектування, де необхідно знайти найкращий компроміс. Для прискорення вирішення питань призначення та отримання покращених та перевірених архітектур передбачається оптимізація, автоматизація

проектування та автоматичне завершення архітектури.

Призначення завдань зіставляє набір завдань з пристроями встановленим в анатомічній ділянці. При цьому враховуються всі обмеження на ресурси та сегрегацію. Призначення завдань може, наприклад, мінімізувати масу пристрою за рахунок використання меншої кількості пристроїв або масу периферійних периферійних проводів. Розподіл периферійних проводів передбачає архітектуру, в якій завдання і периферія вже розподілені за місцями. Він знаходить оптимальні, наприклад найкоротші маршрути для периферійних дротів. Якщо пов'язані завдання потрібно розділити, маршрути також розділяються. Призначення з'єднань знаходить кабельні маршрути для з'єднань з апаратного рівня, якщо пристрої вже вже призначені. Якщо сигнали також призначені для каналів зв'язку дотримується просторова сегрегація. Призначення сигналів обчислює оптимальні маршрути для сигналів, якщо завдання, пристрої та канали вже призначені. Дотримуючись обмежень пропускної здатності і розділення, призначення сигналів може, наприклад, мінімізувати кількість необхідних каналів і комутаторів. Оптимізація типу пристрою дозволяє визначити оптимальну кількість пристроїв, розподіл завдань і паралельно працюючих пристроїв. Вхідними даними є набір типів пристроїв, системних завдань та анатомії. Ресурси для кожного типу пристроїв не визначені, але вказані можливі типи та верхня межа. Відповідно до ресурсів інфраструктури алгоритм вирішує, скільки екземплярів якого типу пристроїв використовується в якому місці і як розподіляються розподіляються завдання. Цілями можуть бути маса пристроїв і проводки, а також витрати. Оптимізація мережі знаходить оптимальну кількість каналів і комутаторів для заданого набору сигналів а також розташування пристроїв і завдань. Додатковим вхідним параметром є кількість портів на комутаторі. Результатом є екземпляри комутаторів і каналів зв'язку, розміщені в місцях інсталяції. Крім того, всі сигнали маршрутизуються з урахуванням обмежень на ресурси та сегрегацію. Оптимізація топології поєднує в собі тип пристрою

та оптимізацію мережі. Оскільки єдиними вхідними даними є лише завдання, сигнали, типи пристроїв та анатомія, це майже повна генерація архітектури. Він знаходить оптимальну кількість пристроїв і топологію мережі, враховуючи при цьому компроміс між вагою і вартістю пристроїв і мережі.

Вісім оптимізаційних процедур, представлених вище, розв'язуються за допомогою однієї і тієї ж математичної основи. Всі оптимізаційні процедури є задачами комбінаторної оптимізації. А загальновідоме математичне представлення та найсучасніші найсучасніші алгоритми глобального оптимального розв'язання. Всі задачі подаються у вигляді бінарних програм (БП) і розв'язуються розв'язуються за допомогою методу гілок і відсікань. Труднощі полягають у тому, щоб знайти відповідне формулювання БП для кожної задачі IMA-дизайну задачі IMA.

Нехай задано впорядкований набір взаємно попарно простих чисел: m1, m2,…, mn, mn+1, …, mn+к+r, де n, к та r відповідно кількість основних (робочих), контрольних та резервних модулів. Отримаємо формулу для кількісної оцінки показника надійності (імовірності безвідмовної роботи) за наступних умов: основна система, що складається з $m_j(j=\overline{1,n+\kappa})$, протягом часу t не відмовила;

- одночасно відмовило не більше r інформаційних та контрольних, що враховує надійність перемикаючих пристроїв, працює безвідмовно.

Використовуючи формулу ймовірності безвідмовної роботи для ковзного резервування з ненавантаженим резервом та ідеальним та враховуючи, що в кожному з членів суми під знаком останнього інтеграла буде перебувати множник ймовірність безвідмовної роботи протягом часу від початку роботи до останньої відмови, отримаємо:

$$P_{MA}^{(k)}(t)=P^n(t)+nP^{n-1}(t)\int_0^t a(\tau)P(t-\tau)P_{AH}(\tau)d\tau+n^2p^{n-1}(t)\int_0^t a(\tau)\times$$

$$\times\{\int_\tau^t a(\xi-\tau)P(t-\tau)P(t-\xi)P_{AH}(\xi)d\xi\}d\tau+\ldots+n^kp^{n-1}(t)\int_0^t a(\tau)\times$$

$$\times\{\int_\tau^t a(\xi-\tau)\times\int_\xi^t a(\gamma-\xi)\{..\int_y^t a(\Phi-y)P(t-\Phi)P_{AH}(\Phi)d\Phi\}\ldots\}d\gamma\}d\xi\}d\tau.$$

Враховуючи, що можливість безвідмовної роботи $P_1(t)=e^{-\lambda_1 t}$, ймовірність безвідмовної роботи $P_{AH}(t)=e^{-\lambda_A t}$ і частота відмов дорівнює $\lambda_1(t)=e^{-\lambda_1 t}$, $\alpha_{n+k}=[\log_2(m_{n+k}-1]+1$ отримаємо формулу для визначення ймовірності безвідмовної роботи:

$$P_{MA}^{(k')}(t)=e^{-n\lambda_1 t}\sum_{i=0}^{k+r}\left(n\frac{\lambda_1}{\lambda_A}\right)^i-n\frac{\lambda_1}{\lambda_A}e^{-(\lambda_A+n\lambda_1)t}\sum_{i=0}^{k-1}\sum_{j=0}^{k-1-i}\left(n\frac{\lambda_1}{\lambda_A}\right)^i\frac{(n\lambda_1 t)^i}{i!}, \qquad (5.1)$$

де λ_1 – інтенсивність відмов одного ТОІ визначається як:

$$\lambda_1=\alpha\lambda_I=\left\{\right]\frac{1}{n+k+r}\left(\sum_{i=1}^{n+k+r}\left[\log_2(m_i-1)\right]+1\right)\left[\right\}\lambda_I,, \qquad (5.2)$$

$\lambda_A=\lambda_A^1+k\lambda_1$ – інтенсивність відмов АН;

λ_A^1 – інтенсивність відмови перемикаючого пристрою;

$k\lambda_1$ – інтенсивність відмов контрольних ТОІ.

Вираз (5.2) враховує вплив функціонального резервування та дозволяє запропонувати метод підвищення відмовостійкості на основі використання принципу активної відмовостійкості. Суть якого полягає в можливості і здатності одного взяти на себе функції до r, які одночасно відмовили, і за умови $m_j\geq\prod_{i=1}^{r}m_{k_i}$. У формулі (68) проведемо заміну $k'=k+r$. Ця обставина не враховувалося у виразі (89). Представлена математична модель надійності дозволяє розраховувати показники надійності за допомогою простих і відомих співвідношень. У теоретичному плані ця математична модель надійності дозволяє досліджувати всі основні види резервування (структурне, інформаційне та функціональне), зумовлені властивостями :

1. Структурне резервування.

2. Інформаційне резервування. Використовується за рахунок додаткової інформації, яка отримується завдяки використанню контрольних, що вводяться за основами m_{n+k}.

З появою помилок, викликаних збоями в одному з за однією з робочою чи контрольною основою $m_j(j=\overline{1,n+\kappa})$ МА, усуваються відомими методами. Таким чином, синтезована відповідно до розробленої математичної моделі, є як і тройована мажоритарна структура нечутливою до відмов.

3. Функціональне резервування використовується у випадку, якщо виконується умова $m_j \geq \prod_{i=1}^{r} m_i$, що дає можливість резервному ТОІ взяти на себе функції одночасно до r відмовивших. У розглянутій моделі цей вид резервування враховується як добавка до k ще r резервних.

Тоді вираз (3.8) представиться у вигляді:

$$P_{MA}^{(k+r)}(t) = e^{-n\lambda_1 t}\sum_{i=0}^{k+r}\left(n\frac{\lambda_1}{\lambda_A}\right)^i - n\frac{\lambda_1}{\lambda_A}e^{-(\lambda_A+n\lambda_1)}\sum_{i=0}^{k+r-1}\sum_{j=0}^{k+r-1-i}\left(n\frac{\lambda_1}{\lambda_A}\right)^j\frac{(n\lambda_1 t)^i}{i!}.$$

При визначенні надійності, необхідно задати такі вихідні дані:

- кількість інформаційних основ (трактів), що визначається величиною розрядної сітки;
- інтенсивність відмов $\lambda_1 = \alpha\lambda_I$;
- інтенсивність відмов АН (значення коефіцієнта перерахунку $\xi^* = \frac{\lambda_1}{\lambda_A} = \frac{a}{\xi}$).

Крім цього при порівняльному аналізі надійності в та тройованій мажоритарній структурі необхідно враховувати:

- інтенсивність відмов l-байтового позиційного $\lambda_0 = 8l\lambda_I$;
- інтенсивність відмов мажоритарного елементу $\lambda_M = \gamma\lambda_I$

(або значення позиційного коефіцієнта перерахунку $\gamma^* = \frac{\lambda_0}{\lambda_M} = \frac{81}{r}$).

З урахуванням вимог до основних характеристик, при дослідженні математичної моделі надійності будемо вважати, що r =1. У цьому випадку

$$P_{MA}^{(1)}(t) = e^{-n\lambda t}\left[1 + n\frac{\lambda_1}{\lambda_A}\left(1 - e^{-n\lambda_A t}\right)\right]$$

вираз (5.2) подається у вигляді:

а вихідні дані для досліджень математичної моделі надійності в, з урахуванням реальних значень надійності існуючих завдань.

Відомо, що найбільша ймовірність безвідмовної роботи для тройованої мажоритарної структури, що містить три і враховує ймовірність безвідмовної роботи мажоритарного органу $P_M = e^{-\lambda_M t}$ досягається в немодифікованій надлишковій структурі з багаторазовим зв'язком [69]. Для цього варіанта мажоритарної структури ймовірність безвідмовної роботи дорівнює:

$$P(t) = [3P_0^2(t) - 2P_0^3(t)] \times [3P_M^2(t) - 2P_M^3(t)], \qquad (5.3)$$

де $P_0(t) = e^{\lambda_0 t} = e^{8l\lambda_l t}$ – ймовірність безвідмовної роботи l-байтового позиційного.

Зазначимо, що дублювання з усіх варіантів структурного резервування в забезпечує заданий рівень стійкості до відмови при введенні мінімальної кількості ΔA додатково введеного обладнання. Для даного варіанту відмовостійкості обчислювальної структури відмовостійкість АЛП дорівнює:

$$P_{ПСЧ}(t) = 1 - \left[1 - P_0(t)\right]^2. \qquad (5.4)$$

Нехай $\lambda_M << \lambda_0$. Проведемо порівняльний аналіз надійності однобайтової $(l = 1)$при k=2, r=1. Враховуючи критерій мінімальності апаратурної надмірності, отримаємо наступний набір основ $(n = 4)$: $m_1 = 3, m_2 = 4, m_3 = 5, m_4 = 7, m_5 = 11, m_6 = 13, m_7 = 17$.

При цьому $\sum_{i=1}^{4} m_i = 420 > 2^8 = 254$ та НСД $(m_i, m_j) = 1$ для $i \neq j$; $i, j = \overline{1, n+k+1}$.

Інтенсивність відмов в сім разів менше інтенсивності відмов тобто $\lambda_1 = 7\lambda_A$;

Враховуємо значення таблиці 5.1.

Таблиця 5.1

Аналіз ефективності застосування ІМА-модулів щодо зменшення кількості додаткового обладнання

l	ПСЧ			МА		Виграш у кількості обладнання [%]
	модуль	ІМА модуль	Додатко-ве обладна-ння [%]	$k=1$	Додатко-ве обладна-ння [%]	
1	8	24	200	23,5	197	3
2	16	48	200	35	173	27
3	24	72	200	45	162,5	37,5
4	32	96	200	55	158	42

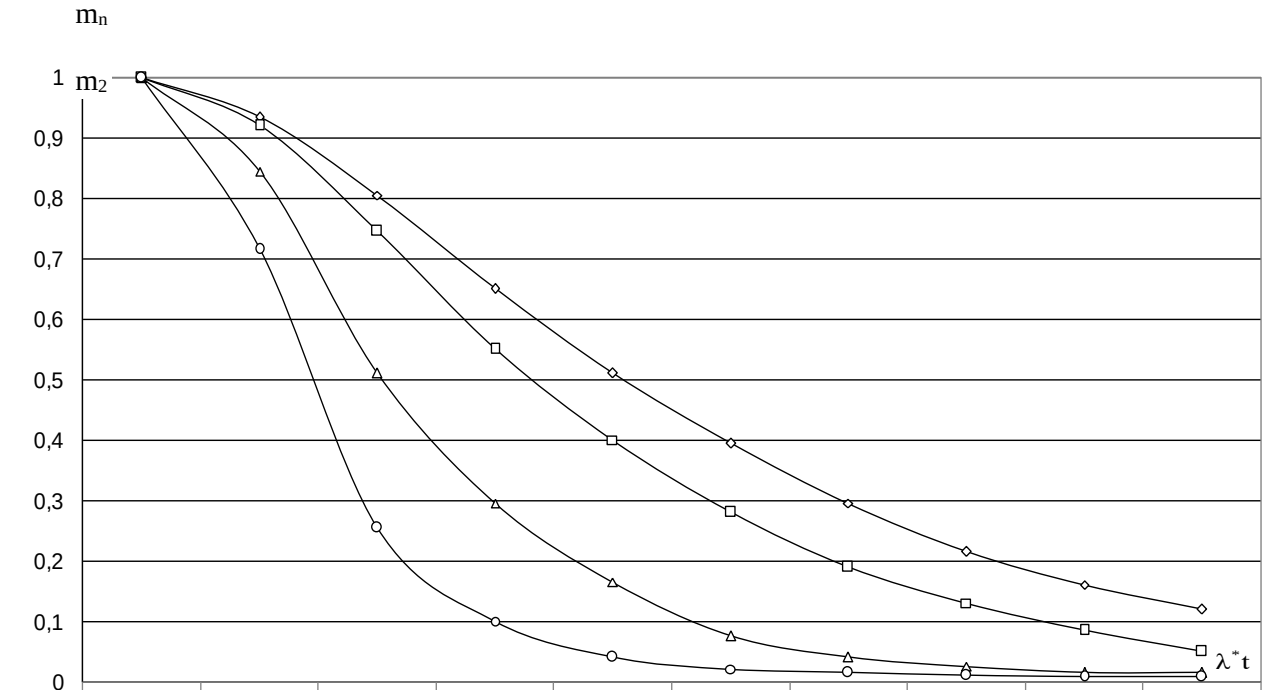
У таблицях 5.2–5.5 наведено результати порівняльного аналізу відмовостійкості та експлуатаційної продуктивності для серійних зразків позиційних модулів ІМА та традиційних арифметико-логічних пристроїв (АЛП), що використовуються у складі первинної цифрової частини (ПСЧ).

Зокрема, таблиця 5.2 демонструє зміну імовірності безвідмовної роботи для традиційної моделі (МО) та модульної архітектури (ІМА) за різних значень інтегрального параметра надійності, що дозволяє оцінити загальну стійкість до відмов на рівні серійного виробництва.

Таблиця 5.2

Результати порівняльного аналізу відмовостійкості серійного виробництва

λ*t	P (λ*t)	
	МО	ІМА
0	1	1
0,001	0,999992	0,999999
0,005	0,9998	0,999999
0,01	0,9992	0,99999
0,05	0,9660	0,9999
0,1	0,8911	–
0,2	0,6975	0,9260
0,3	0,5120	–
0,4	0,3634	0,7750



	0	0,2	0,4	0,6	0,8	1	1,2	1,4	1,6	1,8
l=1	1	0,935	0,805	0,65	0,51	0,395	0,295	0,215	0,16	0,12
l=2	1	0,92	0,745	0,55	0,4	0,28	0,19	0,13	0,085	0,05
l=3	1	0,845	0,51	0,295	0,165	0,075	0,04	0,025	0,015	0,015
l=4	1	0,715	0,255	0,1	0,04	0,021	0,015	0,01	0,008	0,008

Рис.5.8 Графіки залежності при $P_{АЛП}\left(\lambda^{*}t\right)\ l=\overline{1,4}$

У таблиці 5.3 представлено порівняння продуктивності та надійності виконання арифметичних операцій для позиційних модулів у залежності від ступеня резервування, де видно значне зростання кількості операцій на секунду при використанні модульної архітектури IMA в порівнянні з традиційною АЛП у ПСЧ.

Таблиця 5.3

Значення показників продуктивності та відмовостійкості для серійного зразка позиційних модулів при реалізації арифметичних операцій

Показник відмовостійкості	ПСЧ (АЛП)	МА		
		m_i=1	m_i=2	m_i=3
Користувацька продуктивність	3 опер./сек	96 опер./сек	510 опер./сек	121 опер./сек
P (λ*t)	0,966	0,9999	0,9999	0,9999

Таблиця 5.4 представляє порівняльні значення імовірності безвідмовної роботи (надійності) двох систем (позначених як I та II) залежно від заданого значення параметра відмов, ймовірно, інтенсивності відмов або часу/навантаження.

На основі наведених даних можна зробити висновок, що система II демонструє стабільно вищу ймовірність безвідмовної роботи в усьому діапазоні змін параметра. Особливо значуща різниця спостерігається при збільшенні параметра до 0,1–0,2, де система I починає істотно втрачати надійність. Це свідчить про вищу стійкість архітектури системи II до зростаючого навантаження або відмов, що робить її більш придатною для застосування в критичних умовах експлуатації бортових систем авіоніки.

Таблиця 5.4

Результати порівняльного аналізу відмовостійкості серійного зразка позиційного модулів ІМА

$\lambda^* t$	$P(\lambda^* t)$	
	I	II
0	1	1
0,001	0,999996	0,999999
0,003	0,999964	0,999999
0,005	0,999900	0,999991
0,01	0,996000	0,999900
0,05	0,990000	0,999800
0,1	0,967230	0,973010
0,2	0,891100	0,908290

Таблиця 5.5

Значення показників продуктивності та відмовостійкості для серійного зразка позиційних модулів

Показник відмовостійкості	АЛП в ПСЧ	АЛП в МА		
		m_i=1	m_i=2	m_i=3
Користувацька продуктивність [операцій/сек]	25	800	3500	950
P (λ*t))	0,999964	0,999999	0,999999	0,999999

Це пояснюється тим, що в аналізованих моделях відмовостійкості арифметико-логічних пристроїв (АЛП) застосовуються базові властивості модульної архітектури (МА), розглянуті вище. Завдяки цьому досягається підвищення загальної надійності системи за рахунок оптимального розподілу функціональних навантажень, динамічної реконфігурації та адаптивного резервування ресурсів**.** У результаті модуль демонструє вищу стійкість до

відмов**,** навіть за умови менших апаратурних витрат порівняно з традиційним методом мажоритарного троювання АЛП, який широко застосовується у складі систем первинної цифрової частоти (ПСЧ). Це вказує на ефективність використання модульних підходів у сучасних IMA-системах**,** де критично важливо забезпечити баланс між надійністю, продуктивністю та ресурсною економічністю.

5.2. Процес розробки програмного забезпечення та автоматизована інформаційна підтримка IMA.

Процес розробки програмного забезпечення авіоніки складається з чотирьох детальних процесів:

– Процес вимог до програмного забезпечення, який призводить до розробки вимог високого рівня (HLR);

– Процес проектування програмного забезпечення, який розробляє вимоги низького рівня (LLR) і архітектуру програмного забезпечення на основі HLR;

– Процес кодування, який призводить до створення вихідного коду та неінтегрованого об'єктного коду;

– Процес інтеграції програмного забезпечення, який передбачає консолідацію програмного забезпечення у формі виконуваних програм та його інтеграцію із зовнішніми пристроями.

Вимоги високого рівня (HLR) реалізуються на основі архітектури системи та системних вимог. Вони включають часові сигнали, керування пам'яттю, заплановані зв'язки із зовнішніми пристроями, методи реагування та виявлення помилок, моніторинг роботи системи та розділення програмного забезпечення [90-92].

HLR є основою для розробки вимог низького рівня, які використовуються в процесі проектування програмного забезпечення, які включають описи зв’язків із зовнішніми пристроями, визначення та спосіб потоку даних, механізми зв’язку та компоненти програмного забезпечення.

Процес кодування включає переклад LLR у вихідний код і попередньо скомпільований об’єктний код. Це пов'язано з процесом верифікації, оскільки на цьому етапі відбувається попереднє виконання частково розробленого коду. Процес інтеграції передбачає компіляцію та об’єднання скомпільованого коду у виконувані програми (одну або декілька) і вбудовування цього програмного забезпечення на цільову платформу (бортовий пристрій).

Процеси розробки програмного забезпечення визначають один або багато рівнів системних вимог. Вимоги високого рівня визначаються безпосередньо на основі архітектури системи та системних вимог. Вони розробляються в процесі проектування програмного забезпечення, таким чином створюючи взаємопов’язані вимоги низького рівня.

Однак, коли вихідний код створюється безпосередньо на основі вимог високого рівня, він також відповідає вимогам низького рівня та підлягає рекомендаціям щодо вимог низького рівня. Процеси вимог до програмного забезпечення використовують вихідні дані процесу життєвого циклу програмного забезпечення для створення вимог високого рівня. Основним результатом цього процесу є дані про вимоги до програмного забезпечення

Дані про вимоги до програмного забезпечення визначають вимоги високого рівня, включаючи вимоги, надані замовником. Дані повинні включати опис розподілу вимог до програмної системи, враховуючи вимоги безпеки та потенційні умови помилок, функціональні та робочі вимоги для кожного режиму роботи, критерії продуктивності, вимоги та обмеження, пов’язані з часом, обмеження розміру пам’яті, апаратні та програмні інтерфейси, виявлення помилок, моніторинг безпеки, а також вимоги до розділення програмного забезпечення рівні програмного забезпечення для кожного компонента.

Вхідні дані в процес розробки програмного забезпечення авіоніки включають вимоги до програмного забезпечення, план розробки програмного забезпечення та стандарти розробки програмного забезпечення. Якщо

заплановані критерії переходу виконуються, вимоги високого рівня використовуються в процесі виробництва для створення архітектури програмного забезпечення та вимоги низького рівня. Вони можуть включати один або декілька рівнів вимог.

Основним результатом цих процесів є опис проекту, який містить архітектуру програмного забезпечення та вимоги низького рівня. Дані повинні включати детальний опис того, як програмне забезпечення задовольняє вимоги високого рівня, включаючи алгоритми та структуру даних, і як вимоги до програмного забезпечення відповідають процесам і завданням. Він також повинен надавати опис архітектури програмного забезпечення, що визначає структуру програмного забезпечення, із реалізованими вимогами, описами вводу/виводу (наприклад, словник даних, дані та потік керування в рамках проекту), обмеження ресурсів, стратегію управління ресурсами та їх обмеженнями, а також запаси як а також методи вимірювання цих запасів (наприклад, час і пам’ять, процедури послідовності). Опис має включати внутрішньопроцесорні та внутрішньозадачні механізми зв’язку, включаючи фіксовані часові послідовності переривання, методи проектування та деталі їх реалізації (наприклад, завантаження програмного забезпечення). Важливим елементом опису є програмне забезпечення, модифіковане користувачем, методи розділення та заходи щодо запобігання злому розділу, а також опис компонентів програмного забезпечення (незалежно від того, чи є вони новими чи раніше виготовленими) і посилання на базову версію, з якої вони були створені. завантажено. Цей опис має також включати похідні вимоги, що випливають із процесу розробки програмного забезпечення. Якщо система містить неактивний код, опис заходів безпеки для активації коду на цільовому комп’ютері та обґрунтування проектних рішень безпосередньо включені до системних вимог, пов’язаних із її безпекою.

У процесі кодування програмного забезпечення вихідний код реалізується на основі архітектури програмного забезпечення та вимог

низького рівня. Вхідні дані процесу кодування – це вимоги низького рівня та архітектура програмного забезпечення з процесів проектування програмного забезпечення, плану розробки програмного забезпечення та стандартів кодування програмного забезпечення [97]. Процес програмного кодування може бути розпочато, коли будуть виконані заплановані критерії переходу. Вихідний код розробляється в ході цього процесу і базується на системній архітектурі та вимогах низького рівня. Цільовий комп'ютер і вихідний код процесів кодування програмного забезпечення використовуються для компіляції, об'єднання та завантаження даних у процесі інтеграції; це спрямовано на інтеграцію системи авіоніки або її складових обладнання [98].

Розробили алгоритми роботи контролю обчислювальної системи інтегрованої модульної авіоніки впроцесі проектування та експлуатаціїї, рисунок 5.6

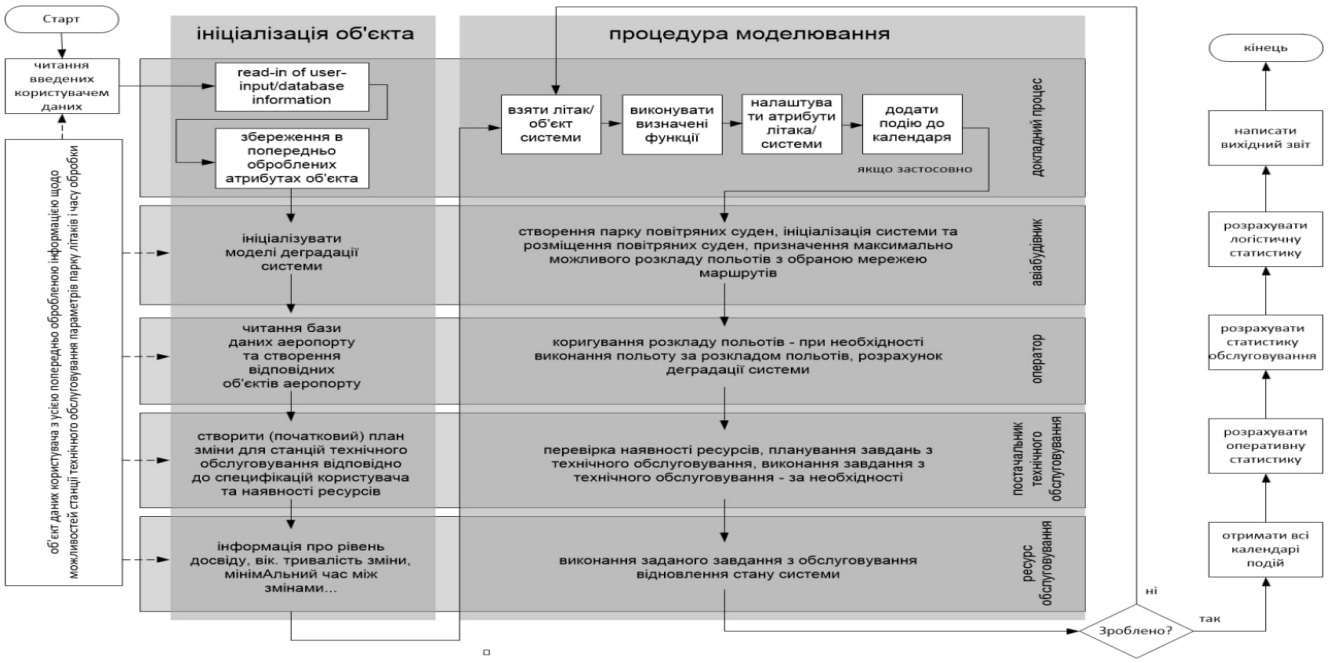


Рис.5.6 Алгоритми роботи контролю обчислювальної системи інтегрованої модульної авіоніки впроцесі проектування та експлуатаціїї

Процес інтеграції програмного забезпечення авіоніки складається з чотирьох детальних процесів:

– Комунікація в рамках сертифікації програмного забезпечення;
– Управління вимогами в рамках сертифікації ПЗ;
– Перевірка в рамках сертифікації ПЗ;

– Оцінка якості в рамках сертифікації ПЗ.

Спілкування в рамках сертифікації є важливим процесом, завданням якого є успішне завершення сертифікації програмного забезпечення. Це передбачає постійну співпрацю та спілкування між заявником та органом сертифікації. Заявник є організацією, яка бажає отримати сертифікацію. Цей процес поширюється на весь життєвий цикл програмного забезпечення, який починається з планування та закінчується його утилізацією. Завдання заявника полягає у визначенні заходів відповідності, які визначають спосіб, у який програмне забезпечення буде задовольняти основним вимогам сертифікації.

Процес управління вимогами, як і комунікація в рамках сертифікації, поширюється на весь життєвий цикл програмного забезпечення. Він охоплює всі дані та документацію, що використовуються для розробки та перевірки програмного забезпечення. Управління вимогами — це «мистецтво» визначення, організації та контролю змін на етапі розробки програмного забезпечення. Основним завданням цього процесу є досягнення максимально можливої ефективності при мінімізації помилок. Метод управління вимогами пов'язаний з рівнем пошкодження бортового обладнання. Стандарт DO-178C визначає два рівні управління програмним забезпеченням: рівні CC1 і CC2. Рівень CC1 має відповідати всім вимогам DO-178C, тоді як рівень CC2 задовольняє лише деякі з них (пов’язані з рівнями гарантії C і D).

Процес верифікації програмного забезпечення передбачає виявлення та опис помилок, які виникли від етапу планування програмного забезпечення до етапу розробки. Стандарт DO-178C визначає не методи, які використовуються для верифікації, а скоріше цілі, які мають бути досягнуті.

Завдання процесу оцінки якості полягає в тому, щоб продемонструвати, що розроблене програмне забезпечення відповідає передбачуваним вимогам і стандартам, що, як наслідок, має призвести до того, що продукт відповідає очікуванням замовника (або демонструє розбіжності щодо прийнятих вимог). Оцінка якості програмного забезпечення – це безперервний процес, який

починається на етапі планування та продовжується через етапи розробки та тестування до кінцевого продукту [119, 118].

Початковою точкою розробленого методу визначено, згідно безпеки програмного забезпечення, можна визначити наступним чином:

$$R(t) = 1 - Q(t) \quad (5.5)$$

де R(t) – стан безпеки програмного забезпечення (уразливостей не виявлено); Q(t) – вразливий стан (уразливості, виявлені в досліджуваному ПЗ); λ(t) – інтенсивність переходу зі стану R(t) у Q(t).

Інтенсивність появи вразливості програмного забезпечення визначається як щільність ймовірності появи вразливості в часі за умови, що протягом цього часу не виявлено жодної уразливості програмного забезпечення [120–122].

Інтенсивність появи вразливості програмного забезпечення :

$$\lambda(t) = lim_{\Delta t \to 0} \frac{P\{t < T \le t + \Delta t | t < T\}}{\Delta t} \quad (5.6)$$

Імовірність виявлення вразливості в одному випадку програмного забезпечення виражається формулою: $q = 1 - e^{-\lambda t}$ (5.7)

де q – ймовірність виявлення вразливості в програмному забезпеченні; t – час роботи в інтервалі часу (0, t).

Подальші перетворення дозволили визначити оцінку ($\lambda\hat{}$) невідомої інтенсивності виявлених вразливостей — параметр λ. Для цього використовуємо метод максимальної правдоподібності, який дає результат, який можна записати так: $\lambda\hat{} = \frac{n1+n2+\cdots+ni}{T_1+T_2+\cdots+T_i}$ (5.8)

Запропонований метод передбачає розрахунок ймовірності усунення вразливостей програмного забезпечення та середнього значення усунутих уразливостей (у відповідний момент часу). Запропонований підхід базується на використанні розгалужених процесів [123].

Отже, розглядаємо сценарій із застосуванням заходів захисту від уразливості. У такому випадку передбачається, що після одного періоду тестування всі вразливості програмного забезпечення виявлені та виправлені.

Як наслідок, ці попередні вразливості або не існують у наступний період, або їхній вплив падає до безпечного рівня; однак нові вразливості з'являються через виправлення попередніх. Крім того, завжди існує принаймні одна вразливість програмного забезпечення, тому для спрощення позначення передбачається, що нульовий період має принаймні одну вразливість.:

$$Q(n+1) = \sum_{k=Q(1)}^{Q(n)} Y_k^{(n+1)} \quad (5.9)$$

де $(Y_k)_{k\geq 1}$ утворює послідовність незалежних, однаково розподілених, невід'ємних випадкових величин.

Пошук вразливостей програмного забезпечення безпеки авіоніки, які призводять до помилок або хакерських атак, вважається корисною діяльністю, яка може сильно вплинути на безпеку та надійність польотів. Здатність передбачити появу вразливостей програмного забезпечення або кількісно виміряти їх вплив дає змогу прогнозувати тенденції безпеки програмного забезпечення та планувати широко зрозумілий процес управління його безпекою [124].

Розроблений метод спрямований на покращення можливості прогнозування вразливостей у тестованому програмному забезпеченні. Перевірка та подальше підвищення точності запропонованого методу вимагає подальших досліджень з подальшим емпіричним аналізом з використанням даних із баз даних уразливостей або інших типів ресурсів щодо вразливостей.

Отже, одним із рішень, запроваджених на AFIT у сфері обмеження помилок у розробленому програмному забезпеченні авіоніки, є автоматизована система керування відповідно до вимог стандарту DO-178C та впровадження цих вимог у формі процедури в ISO -9001 Система забезпечення якості. Створена комп'ютерна система дозволяє здійснювати перевірки та створювати документи, необхідні за стандартом DO-178C (тобто плани, звіти, звіти) безпосередньо з IT-мережі [125].

Завданням, прийнятим для реалізації, було створення засобу

автоматизованого керування розробкою програмного забезпечення для параметрів польоту, яка розроблятиметься та створюватиметься в AFIT за стандартом DO-178C.

Основні функції побудованої IT системи комп'ютерної підтримки включали:

– Налаштування нового проекту, що передбачає введення інформації про назву проекту, персональні дані керівника проекту та окремих підрядників, їх повноваження та рівні доступу системи до системи;

– Внесення даних до бази знань щодо реалізації проекту (деталі, фінанси, обмеження);

– Автоматична генерація шаблонів документів, необхідних у стандарті DO-178C (тобто планів, стандартів, процедур і методів перевірки, звітів та інших записів);

– Автоматична генерація тестів для розробленого програмного забезпечення та архівація результатів тестування;

– Архівування листування між виконавцями проекту, програмних файлів і результатів їх тестування;

– Автоматичне резервне копіювання файлів на сервер, розташований в іншій будівлі на території АФІТ (захист від втрати даних);

– Надання даних про реалізацію проекту згідно введеної авторизації користувачів системи;

– Звітування про статус проекту для цілей аудиту чи перевірки відповідно до введених інструкцій.

Комп'ютерна система, що підтримує управління керуванням програмним забезпеченням авіоніки, після інсталяції спеціалізованого програмного забезпечення (включаючи статичні та динамічні аналізи, адаптовані до вразливості програмного забезпечення та виявлення помилок, для визначення його вразливості до структурних пошкоджень системи авіоніки та хакерських атак), забезпечує прямі, електронні співпраця між учасниками проекту та її контроль з боку керівника проекту, який відповідає

за правильне виконання проекту.

Основним структурним елементом комп’ютерної системи, що забезпечує управління розробкою програмного забезпечення авіоніки, є спеціальний сервер, вбудований у мережу AFIT IT. Сервер взаємодіє з робочими станціями окремих користувачів системи. Файловий сервер, який називається резервним, використовується для захисту зібраної інформації. Його завдання – архівувати файли. Поточний стан виконання проекту зберігається в його пам'яті після кожного «робочого дня». Він також співпрацює з аварійним сервером, який включається в разі збою основного сервера (захист поточного прогресу проекту) [126].

Операційне програмне забезпечення основного сервера включає операційну систему Windows Server, базу даних Windows SQL Server і пакет редагування та розрахунків MS Office як спеціалізоване програмне забезпечення для управління проектами з використанням інструкцій згідно стандарту DO-178C.

Модель комп’ютерної системи, що підтримує розробку програмного забезпечення авіоніки, критичного для безпеки польотів; (1) діаграма архітектури комп’ютерної системи, інтегрованої з мережею IT; (2) діаграма архітектури файлового сервера [127].

Спеціалізоване програмне забезпечення, встановлене на сервері, використовує обчислювальні модулі, до складу яких входять:

– Базовий аналіз для попереднього тестування та верифікації розробленого програмного забезпечення (Static Analysis, Dynamic Analysis, TBvision, TBrun, TBmisra, TBsafe) [128];

– Розширений аналіз для попереднього тестування та верифікації розробленого програмного забезпечення (Modified Condition/Decision Coverage, Information Flow Analysis, Dynamic Data Flow Coverage, Extract Semantic Analysis) [127];

– Додатковий аналіз для попереднього тестування та верифікації розробленого програмного забезпечення (Test Vector Generation, TBeXtreme, TBmanager, Support for Target Testing, Tool Qualification);

– Додаткове програмне забезпечення для тестування, яке забезпечує постійну підтримку процесу розробки окремих програмних компонентів.

Представлене спеціалізоване програмне забезпечення, інтегроване з комп'ютерною системою підтримки процесу розробки програмного забезпечення авіоніки [128].

У технічній реалізації комп'ютерної системи підтримки процесу управління розробкою програмного забезпечення авіоніки використовувалися спеціалізовані модульні комп'ютери, які функціонували як сервери та вбудовані в IT-шафу.

У мережу AFIT IT також входять модулі живлення, комутатори, з'єднувальні панелі, кабельна розводка та додаткові елементи. Модульна структура серверів і застосування панелей підключення дозволяє вибрати конфігурацію IT-мережі, оптимальну для системного адміністратора та користувачів.

Вимоги до безпеки польотів є результатом оцінки рівня безпеки, який містить функціональні вимоги, вимоги до інтеграції та надійності для даної системи. Вимоги до рівня помилок визначаються в ході процесу оцінки безпеки, щоб гарантувати цілісність системи шляхом визначення засобів захисту системи та відповідей у разі таких помилок. Ці вимоги визначені для програмного та апаратного забезпечення з метою усунення або обмеження ефектів помилок, а також забезпечення виявлення помилок, допуску, видалення та уникнення. Системні процеси, відповідальні за вдосконалення та призначення системних вимог апаратному та/або програмному забезпеченню, призводять до розробки відповідної архітектури для системи відображення параметрів польоту.

Конфігурація BIOS розуміється як процес налаштування параметрів BIOS (двійкова система введення/виведення), доступних програмісту, що

призводить до покращеної взаємодії між апаратним і програмним рівнями.

Програмне забезпечення, нестандартна поведінка якого, як зазначено в процесі оцінки безпеки, може призвести або сприяти суттєвій помилці, що призведе до умов, що обмежують функціональні можливості вертольота або додаткове навантаження для пілота, підлягає спеціальному аналізу [129].

Процес планування із залученням програмного забезпечення для встановлення системи відображення параметрів польоту визначається таким чином, щоб вимоги були виконані, а рівень достовірності був адекватним рівню надійності прийнятого програмного забезпечення.

Відповідно до вимог стандарту DO-178C процеси розробки програмного забезпечення, відображення параметрів польоту містяться в процесі планування програмного забезпечення та процесі розробки програмного забезпечення. Процеси, пов’язані з розробкою програмного забезпечення, включають процеси визначення вимог до програмного забезпечення; процеси, пов’язані з кодуванням програмного забезпечення, і процеси, пов’язані з інтеграцією [130].

Процес інтеграції програмного забезпечення для системи відображення параметрів польоту, включає інтеграцію програмного забезпечення та інтеграцію апаратного/програмного забезпечення. Процеси інтеграції можуть бути виконані, коли виконано заплановані вимоги переходу. Входами процесу інтеграції є архітектура програмного забезпечення з процесів проектування програмного забезпечення та вихідний код із процесів кодування програмного забезпечення, тоді як виходами процесу інтеграції є файли об’єктного коду з компіляцією. Інтеграційні процеси є завершеними, коли досягнуті їхні цілі та цілі пов’язаних інтегральних процесів. Об’єктний код має бути згенерований із вихідного коду, а потім скомпільований.

Усі файли з параметрами даних мають бути згенеровані, а програмне забезпечення має бути інтегровано в головний комп’ютер, емулятор цільового пристрою або цільовий пристрій. Програмне забезпечення має бути реалізовано на цільовому комп’ютері з метою інтеграції

апаратного/програмного забезпечення. Невідповідні або помилкові вхідні дані, виявлені під час процесу інтеграції, мають бути направлені до процесів вимог до програмного забезпечення, процесів розробки програмного забезпечення, процесів кодування або процесів планування програмного забезпечення як зворотний зв’язок, який потребує перевірки.

Графічний комп'ютер системи відображення параметрів польоту передає інформацію про параметри польоту на денний дисплей або нічний дисплей. Інформація, отримана від системи узгодження сигналів, приймача супутникової навігації GPS і блоку аеродинамічних даних ADU, подається у вигляді графічних символів або в цифровому вигляді.

Структура графічного програмного забезпечення складається з таких елементів:

– Безпосереднє налаштування операційної системи процесора BIOS;

– Конфігурація вбудованої операційної системи WINDOWS XP;

– Конфігурація програмного забезпечення користувача графічного комп’ютера;

– Архітектура програмного забезпечення користувача графічного комп’ютера

Графічне програмне забезпечення комп'ютера вбудовано в постійну пам'ять материнської плати ЦП. Результати калібрування для окремих каналів вимірювання зберігаються у зовнішній пам’яті комп’ютера у FLASH-пакеті.

Графічна ідентифікація комп’ютерного програмного забезпечення містить таку інформацію, як назва програмного забезпечення, ідентифікаційний номер програмного забезпечення, ідентифікатор версії програмного забезпечення, модулі компонентів програмного забезпечення та надану ліцензію.Системне програмне забезпечення проходить тестування з метою підтвердження його відповідності встановленим вимогам стандартів AQAP 2210 та DO-178C [103]. Водночас забезпечується високий рівень упевненості у тому, що потенційні помилки, здатні спричинити неприйнятні

умови відмови, ідентифіковані в процесі оцінки безпеки відповідно до стандарту ARP 4761, своєчасно виявляються та усуваються.

5.3. Уніфіковане автоматизоване робоче місце з перевірки конструктивно-фунціонального модуля ІМА КБО

В процесі виробництва КБО виробники стикаються з проблемою контролю якості продукції, що випускається. Традиційно контроль якості продукції на авіаприборобудівльному підприємстві здійснюється за допомогою перевірки кожного виготовленого виробу в складі автоматизованого робочого місця [102].

АРМ з перевірки КБО має забезпечувати:

– імітацію процесів інформаційного обміну по інтерфейсів SpaceWire для всіх типів модулів, по ПК, РК, МКІО для модулів введення-виведення, по інтерфейсу Fibre Channel для модуля графічного і модуля-комутатора;

– контроль процесу тестування;

– завантаження програмного забезпечення, тестування і ведення файлів звіту по тестуванню кожного модуля.

– Функціональна схема АРМ представлена на рисунку 5.7 АРМ з перевірки КБО містить:

– інструментальну електронно-обчислювальну машину, що забезпечує установку інструментальних і програмних засобів;

– технологічне обладнання (технологічні модулі, які забезпечують сполучення інтерфейсів КБО і інтерфейсів ІЕВМ, комплект з'єднай-них джгутів для з'єднання рами з встановленим КБО з технологічним обладнанням і джерелом живлення);

– програмне забезпечення для тестування кожного модуля, принтер для виведення на друк результатів тестування;

– джерело живлення для подачі напруги на КБО;

– комплект експлуатаційної документації, що містить інструкції для перевірки і завантаження тестового програмного забезпечення.

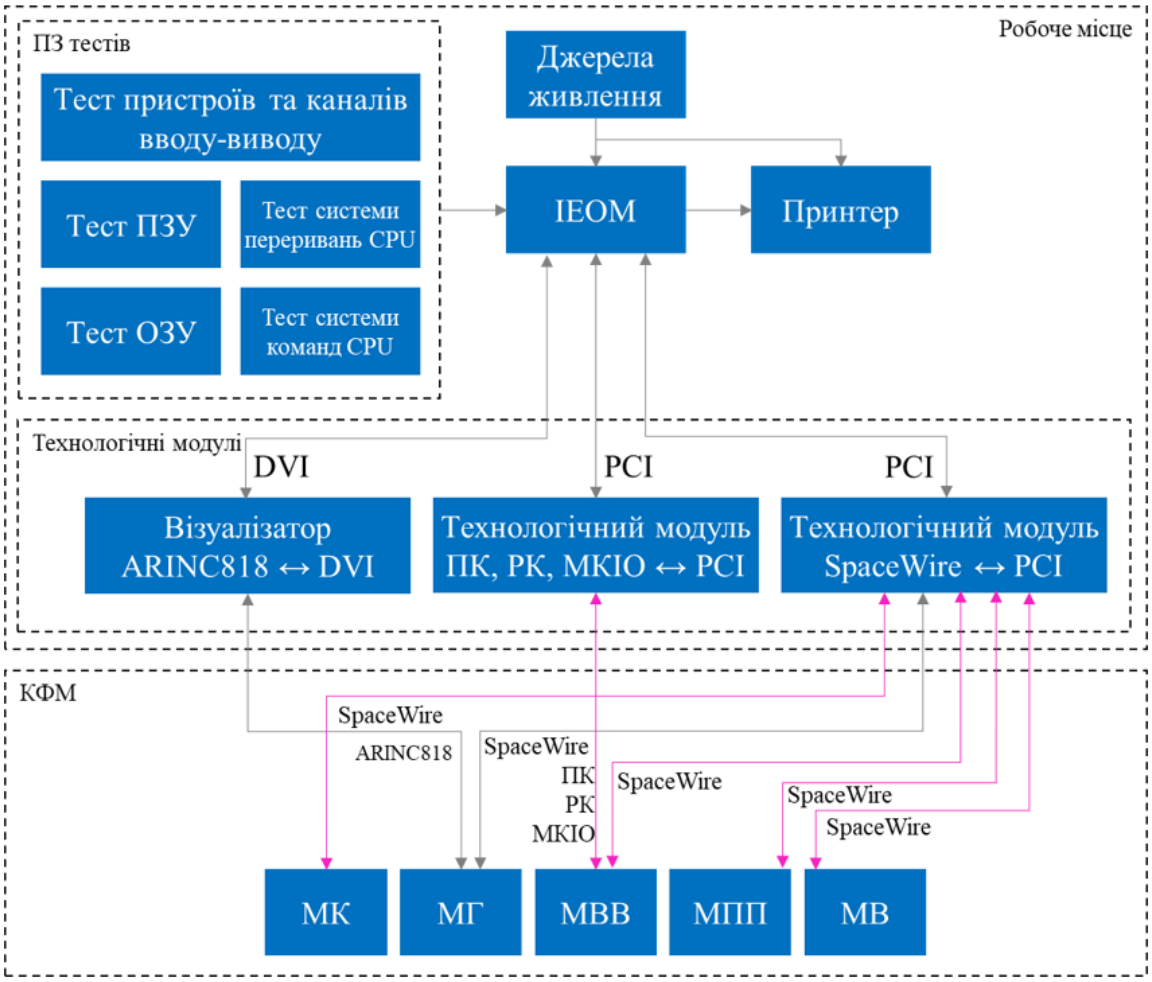


Рис. 5.7. Модель уніфікованого автоматизованого робочого місця (DVI – Digital Visual Interface)

На рисунку представлено структурну схему уніфікованого автоматизованого робочого місця перевірки конструктивно-функціонального модуля інтегрованої модульної авіоніки (IMA), яке включає програмне забезпечення тестів для перевірки пристроїв і каналів введення-виведення, ПЗУ, ОЗУ, систем переривань та команд центрального процесора; джерела живлення, що забезпечують електроживлення інструментальної EOM (IEOM) та периферійних пристроїв, зокрема принтера; технологічні модулі, серед яких візуалізатор ARINC818 ↔ DVI, модулі ПК, РК, МКІО ↔ PCI та SpaceWire ↔ PCI; а також об'єкти перевірки – конструктивно-функціональні модулі (КФМ), серед яких модулі керування (МК), генерації (МГ), вводу-виводу (МВВ), попередньої обробки (МПП) та відображення (МВ). Взаємодія між технологічними модулями та КФМ здійснюється за

допомогою цифрових інтерфейсів SpaceWire, ARINC818, ПК/РК/МКІО, що дозволяє забезпечити повноцінну перевірку функціональності та відмовостійкості модулів в умовах, наближених до реального середовища експлуатації, рисунок 5.8.

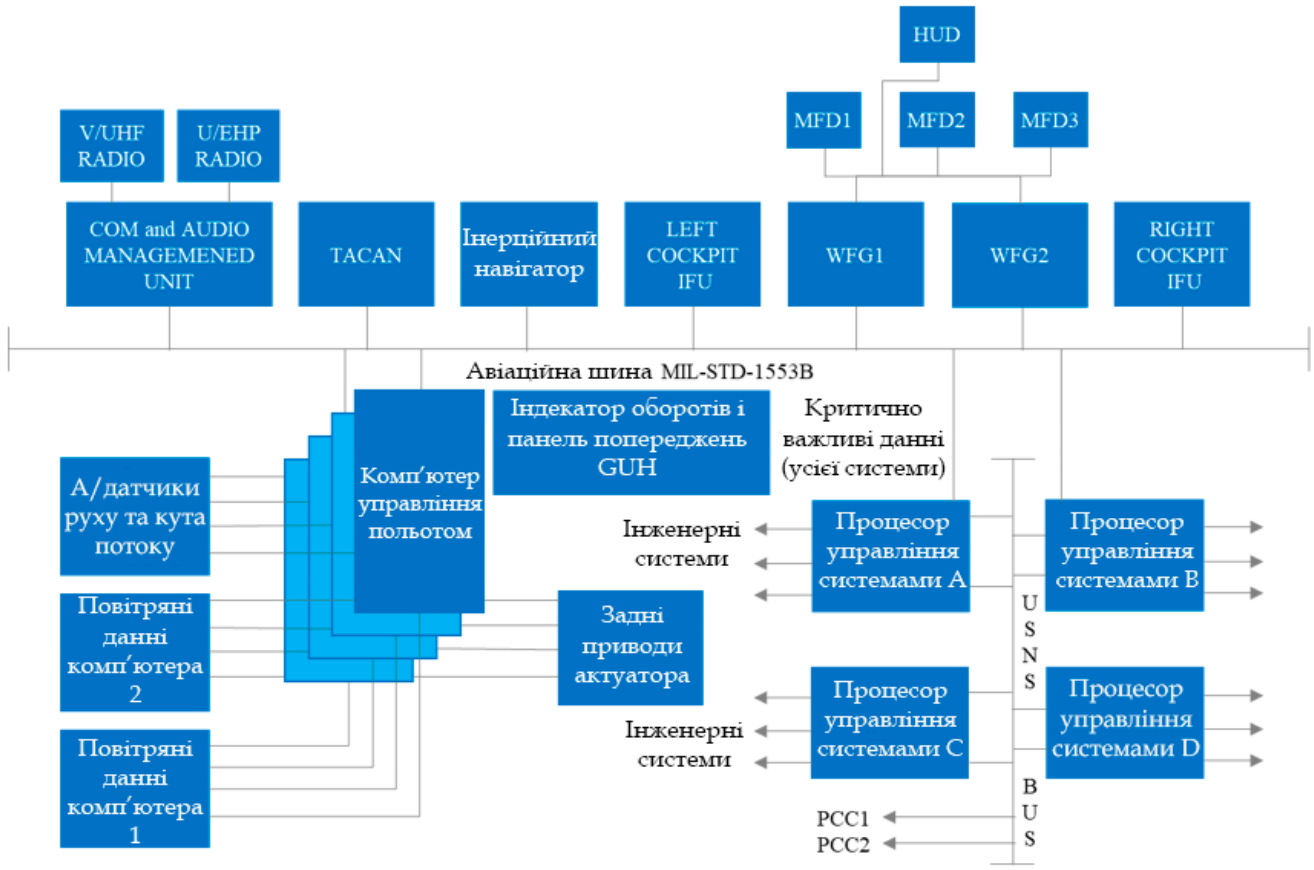


Рис. 5.8 Функціональна схема взаємодії елементів системи управління польотом через авіаційну шину MIL-STD-1553B у складі IMA

В потрібно тільки технологічний модуль, що забезпечує сполучення інтерфейсу PCI і інтерфейсу SpaceWire. Для МВВ додатково необхідний технологічний модуль для сполучення каналів ПК, РК і МКІО з інтерфейсом IEBM. Для МГ необхідний технологічний модуль для сполучення SpaceWire і PCI, а також визуализатор для сполучення інтерфейсу Fibre Channel і DVI. Визуализатор підключається безпосередньо до монітора з виходом DVI [100].

На рисунку 5.9 представлено два варіанти алгоритмічних схем автоматизованого тестування працездатності модуля IMA. Підваріант (а) реалізує послідовну логіку виконання тестів – оперативної пам'яті (ОЗУ), постійної пам'яті (ПЗУ), регістрів керування (РК), каналів введення-

виведення (КВВ) та центрального процесора (CPU), після чого виконується перевірка результатів: у разі успішного проходження всіх тестів модуль вважається справним, в іншому випадку — несправним. Підваріант (б) ілюструє паралельну структуру виконання тестів, де всі перевірки відбуваються одночасно з подальшим прийняттям рішення щодо справності модуля за результатами агрегованої оцінки. Такий підхід дозволяє порівняти ефективність послідовного та паралельного підходів до верифікації функціонального стану модуля в умовах автоматизованої діагностики.

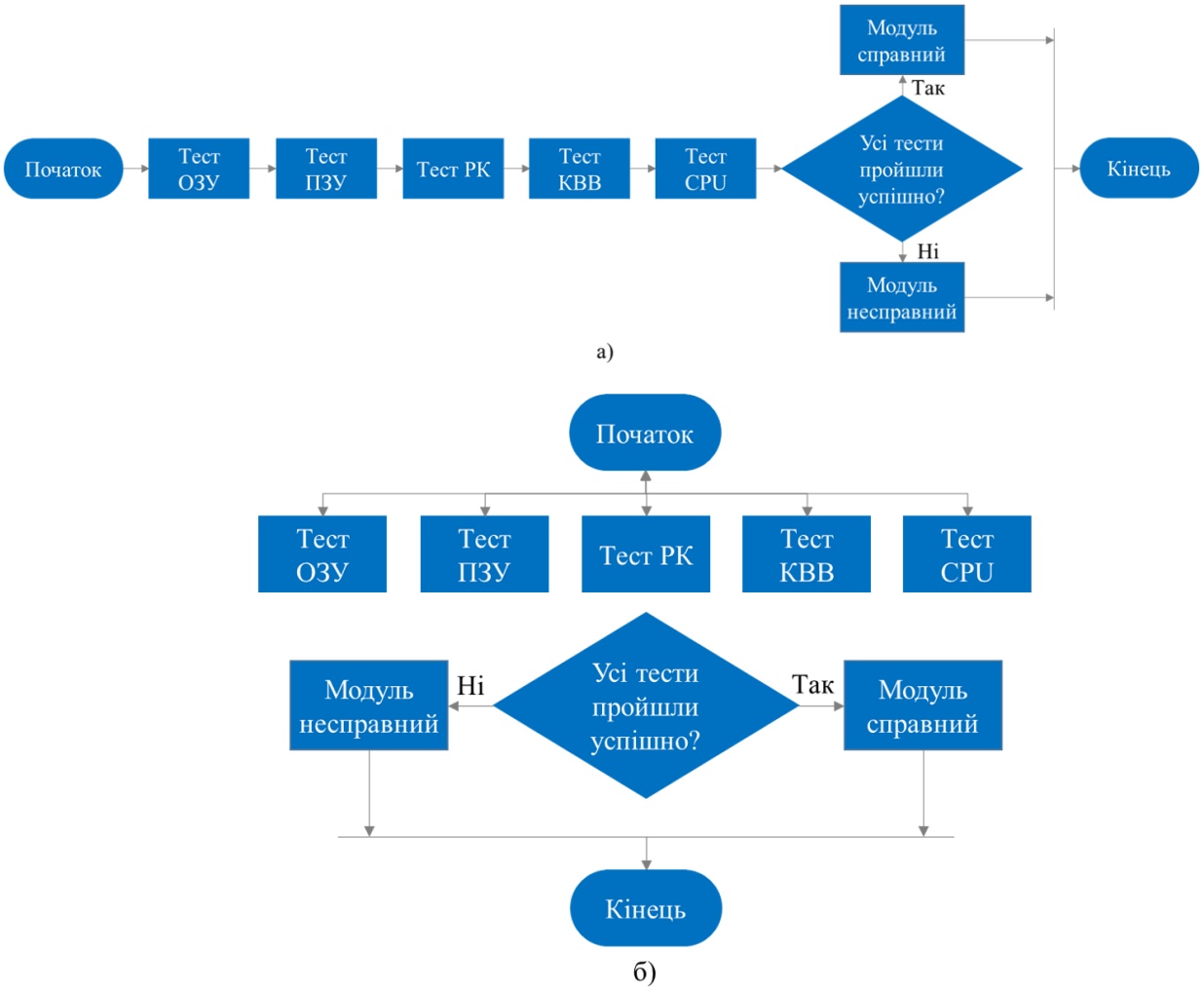


Рис. 5.9. Алгоритми роботи автоматизованого робочого місця перевірки КБО: а) в послідовному режимі, б) в паралельному режимі

На рисунку 5.9 представлено узагальнену структурно-функціональну модель процесу формування інформаційної підтримки проєктування та експлуатації бортового програмного забезпечення в системах інтегрованої модульної авіоніки (IMA) [113-114]. Модель охоплює етапи розробки,

верифікації, інтеграції, експлуатації та обслуговування програмного забезпечення ПС, із урахуванням факторів надійності, обслуговуваності, часу технічного супроводу та результатів технічного аудиту. Центральне місце займає діаграма Гантта (GSE), яка відображає послідовність процедур аналізу відмов, оцінки ефективності, ідентифікації впливових факторів, моделювання інформаційних потоків, аналізу обсягів даних та формування технічних рішень. Модель також включає взаємодію з модулями ПС КСАО, базами даних, засобами відображення результатів та механізмами верифікації, що забезпечує інтеграцію процесів оцінювання якості та підтримки прийняття рішень у рамках життєвого циклу ПЗ авіоніки.

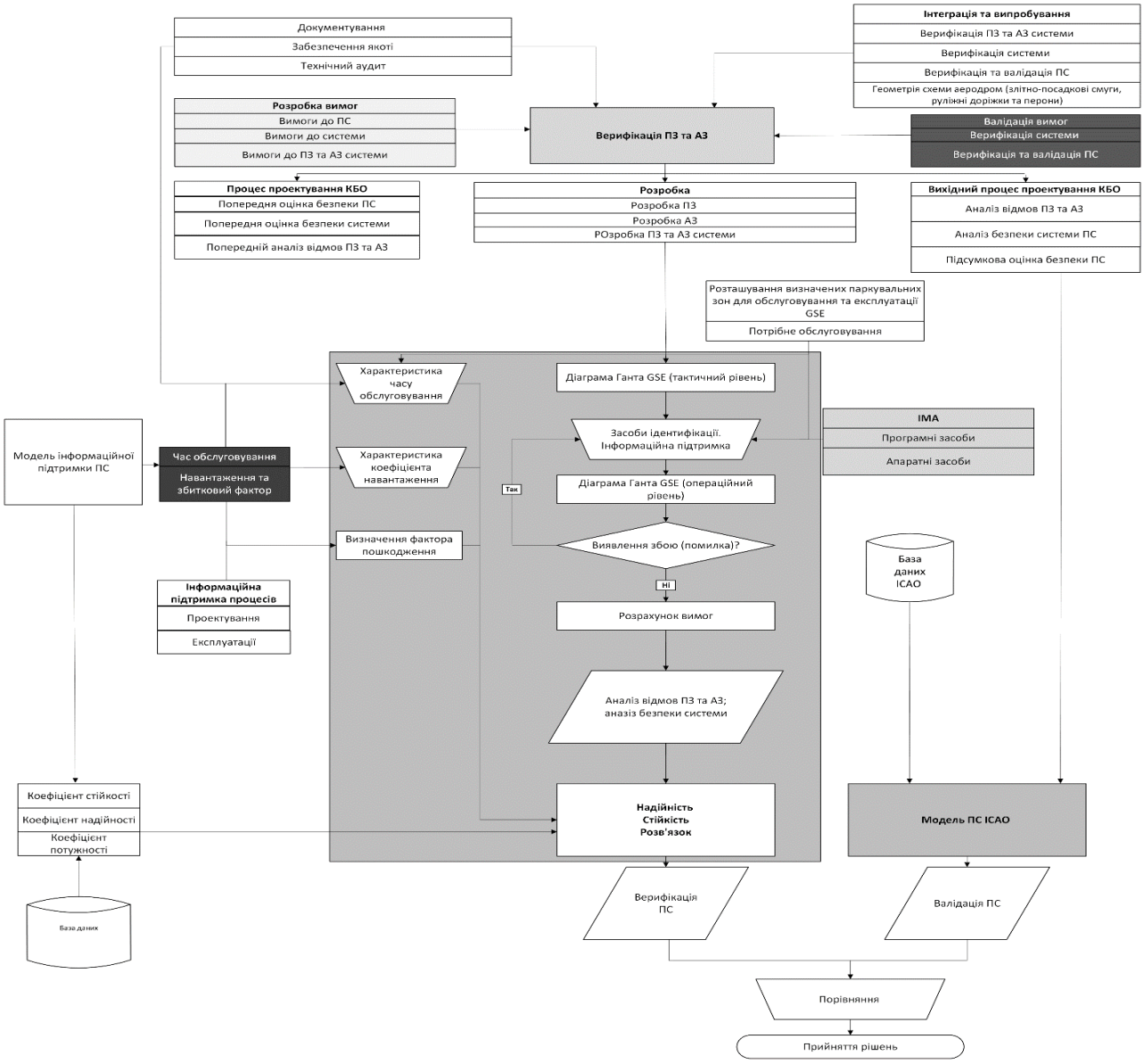
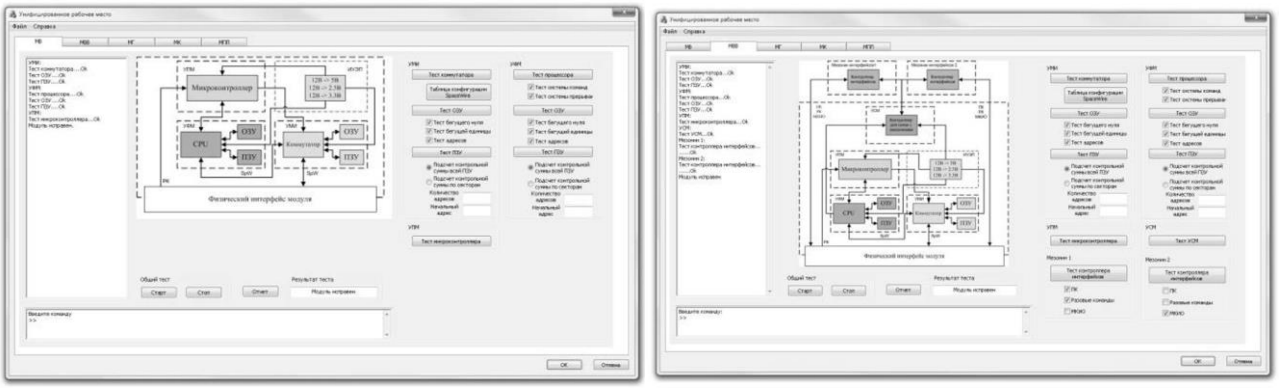


Рис. 5.9. Алгоритм роботи бортового обладнання під час польоту.

Тестування починається з подачі на КБО напруги харчування. Потім завантажується і запускається необхідне програмне забезпечення для тестування модуля. Під час тестування модуля перевіряються всі вузли КБО: осередки пам'яті ОЗУ, осередки пам'яті ПЗУ, система команд процесора, система переривань процесора, пристрої та канали введення-виведення. Всі результати тестування видаються на екран, а також заносяться в файл звіту[115]. Розрізняють 2 види алгоритмів перевірки модулів в складі АРМ: в послідовному режимі (рис. 5.10 а) і в паралельному режимі (рис. 5.10 б).

Інструментальна програма перевірки КБО забезпечує занесення в КБО тестового ПО за технологічним каналу і обмін інформацією між ІЕВМ і КБО. Зовнішній вигляд робочих вікон програм САПР з перевірки КБО представлені на рисунку 5.10 – 5.11 .



а) б)

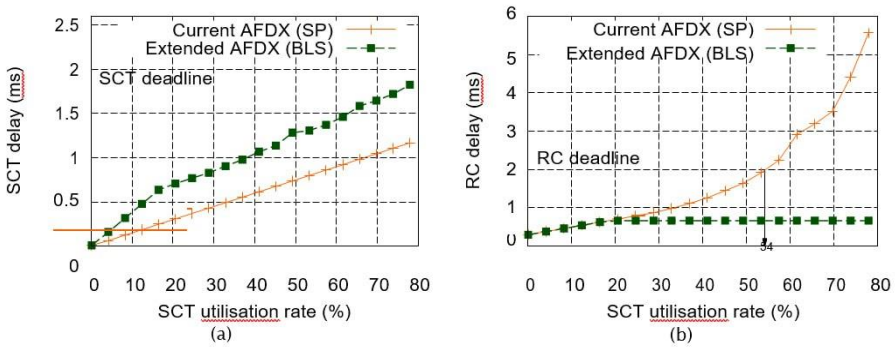
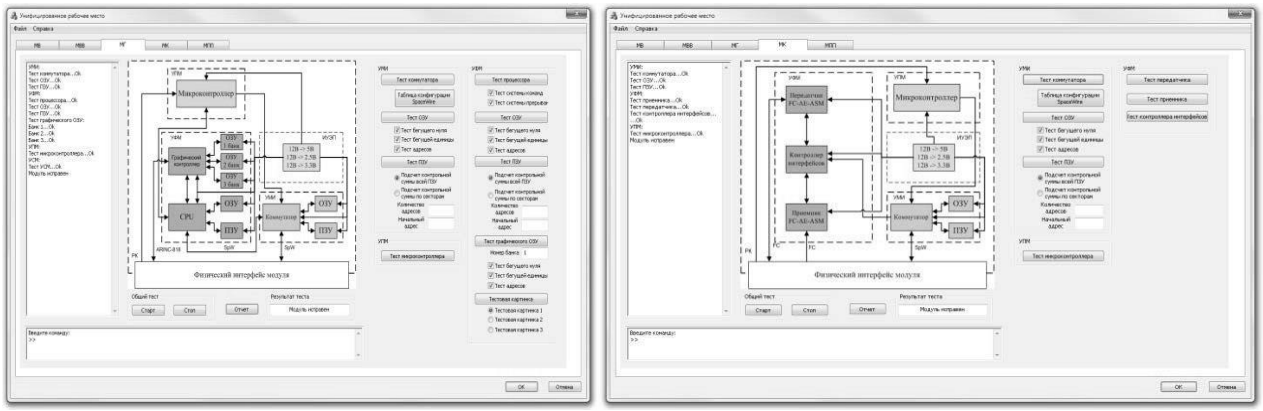


Рис. 5.10. Робоче вікно програми САПР для перевірки та усунення несправностей: а) модуль обчислювальний ;б) модуль вводу-виводу



а) б)

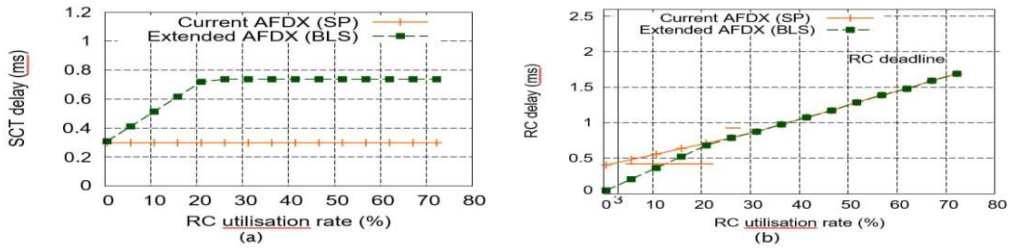
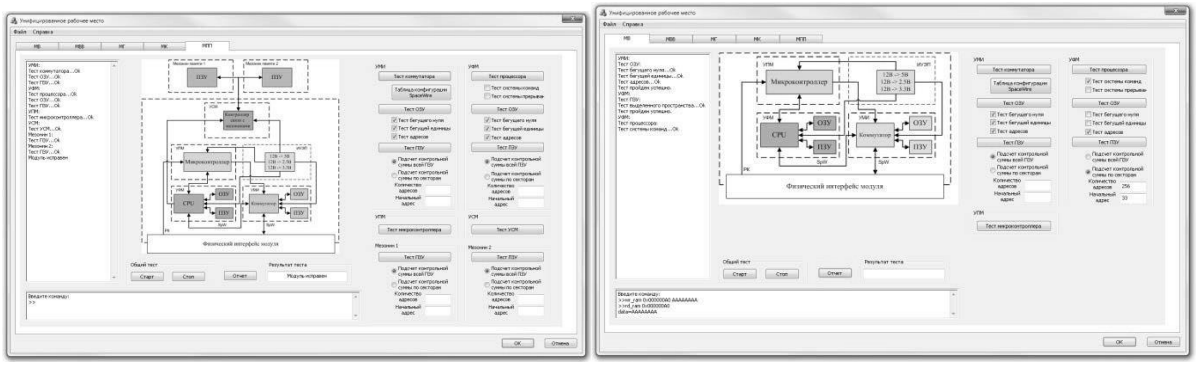


Рис. 5.11 Робоче вікно програми САПР для перевірки та усунення несправностей: а) модуль графічний б) модуль комутатор

Робоче вікно інструментальної програми розділене на кілька фрагментів. Зліва знаходить лог проведених тестів. Знизу командний рядок для проведення окремих операцій, наприклад, читання або запис в ОЗУ / ПЗУ. У центрі робочого вікна знаходить структурна схема модуля, що тестується [116]. Під структурною схемою знаходяться кнопки початку тесту і зупинки тесту, а також кнопка для отримання звіту, а в додаткове вікно виводиться інформація про результат повного тесту: «модуль справний» або «модуль відмовив». Справа в робочому вікні знаходяться кнопки для проведення тестів окремих вузлів модуля з додатковими настройками.



а) б)

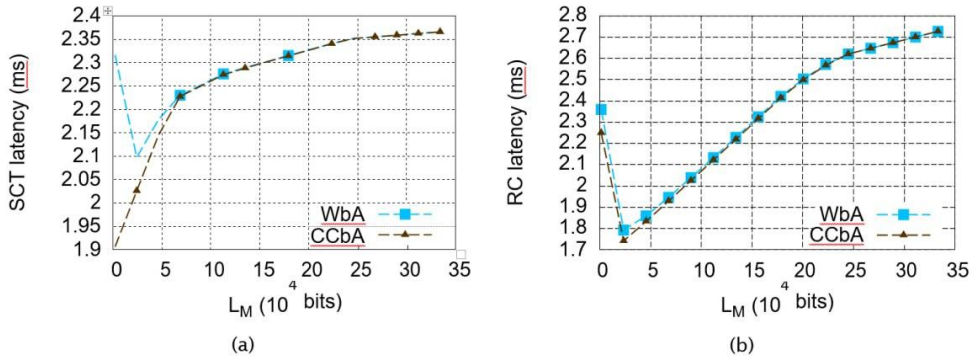


Рис. 5.12. Робоче вікно програми САПР для перевірки та усунення несправностей: а) модуль постійної пам'яті; б) модуль обчислювальний.

Таким чином, інструментальна програма перевірки конструктивно-функціональних модулів (КБО) інтегрованої модульної авіоніки забезпечує ефективну взаємодію з ІЕВМ через технологічний канал шляхом завантаження тестового програмного забезпечення, виконання перевірок та обміну діагностичними даними. Робоче середовище програми САПР має зручну інтерфейсну структуру, що включає лог тестування, структурну схему модуля, командний рядок і функціональні елементи керування тестами. Реалізована можливість адресного тестування окремих вузлів модуля з гнучкими параметрами налаштування дозволяє оперативно виявляти та усувати несправності. Візуалізація результатів тестування у формі статусного звіту підвищує інформативність процесу та сприяє прийняттю рішень щодо технічного стану модуля, що значно підвищує ефективність експлуатаційної підтримки ІМА

На основі порівняльного аналізу засобів інформаційної підтримки згідно з критеріями, поданими в таблиці 5.9, можна зробити висновок, що впровадження розробленого підходу до інформаційної підтримки процесів проєктування та експлуатації КБО ІМА дозволило істотно підвищити відповідність більшості критеріям ефективності. Зокрема, спостерігається розширення охоплення за такими критеріями, як універсальність, сумісність з експлуатаційними функціями, можливість роботи в режимі реального часу, інтеграція з модулями КСАО, адаптивність до різних категорій даних,

гнучкість у побудові інформаційних структур **та** підтримка прийняття рішень. У порівнянні з іншими засобами (методичними, програмними, інформаційними тощо), розроблений засіб охоплює ширший спектр критеріїв, що свідчить про його вищу комплексність**,** функціональну повноту та практичну придатність для реалізації в сучасних умовах експлуатації ІМА. Отже, запропонована система демонструє помітне покращення за якісними показниками, що підтверджує доцільність її інтеграції в контур авіаційного життєвого циклу КБО.

Таблиця 5.9

Ефект інформаційної підтримки процесу проектування та експлуатації комплексу бортового обладнання

№	Засоби	Критерії																												
		За ступенем автоматизації		По Універсальності		За ступенем уніфікованості		За видами каналів зв'язку з об'єктом		За видами обробки інформації		За способом обробки інформації		По виду на об'єкт		За видами контролю			За способом контролю параметрів		За місцем розташування		За кількістю різних класів станів в об'єкті		За способом представлення результатів контролю		За способом прийому інформації		По виду програмного пристрою		ІМА
		Неавтоматичні	Автоматичні	Універсальні	Спеціалізовані	Уніфіковані	Неуніфіковані	Провідні	Безпровідні	Інформаційно-логічні	Інформаційно-обчислювальні	З первинною Реєстрацією інформації	Без первинної реєстрації інформації	Активні	Пасивні	Прогнозуючі	Діагностичні	Працездатності	Цифрові	Аналогові	Вбудовані	Зовнішні	Багатоальтернативні	Двоальтернативні	З кількісної оцінкою	З допусковою оцінкою	Паралельно	Послідовно	З зовнішнім програмуванням	З внутрішнім програмуванням	
1	Методичні засоби	-	+ -	-	+	+	-	+	-	-	-	—	-	+	-	—	+	-	-	-	+	+	-	—	—	—	+	+	-	-	+
2	Лінгвістичні засоби	-	+	-	+	-	+	+	-	-	-	+	-	-	+			-	-	—	-	+	+			—	—	+	—	—	+
3	Математичні засоби	-	+	-	+	-	-	+	-	-	-		+	-	+		+	+	-	-	—	-	-	—	-	-	—	-	-	—	+
4	Програмні засоби	-	+	+	+	-	-	+	-	+	-	-	—	-	+		-	-	—	-	-	—	-	-	—	-	-	—	-	-	+
5	Інформаційні засоби	-	+	-	-	+	+	-	-	+	-	-	—	+	+							+	+		-	-		+	-	-	+
6	Організаційні засоби	+	+	-	+	+	-	-	-	-	-	+			+		-	-	—	-	-	—	-	+	+	+	-	-	—	-	+
7	Інформаційна підтримка ППтаЕ КБО ІМА	-	+	+	+	+	-	+	+	+	+	+	+	+	—	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+

5.4. Висновки до п’ятого розділу

Розроблена структурно-логічна модель уніфікованого автоматизованого робочого місця для підвищення достовірності контролю обчислювальної системи інтегрованої модульної авіоніки є важливим кроком у напрямку забезпечення безпеки та надійності авіаційних систем. Ця модель дозволяє систематизувати та візуалізувати робоче місце, забезпечуючи операторам зручний інтерфейс для контролю та аналізу обчислювальної системи. Вона також дозволяє виявляти та усувати потенційні проблеми ще до виробництва та експлуатації, що покращує якість та надійність авіаційних систем.

Результати застосування цієї моделі можуть бути корисними для

планування та оптимізації роботи бортових систем з урахуванням обмежень обчислювальної потужності та мережі передачі даних

Розроблений алгоритм контролю обчислювальної системи інтегрованої модульної авіоніки під час польоту повітряного судна для контролю функціональних елементів та подальшої розробки програмного забезпечення у складі уніфікованого робочого місця обчислювальної системи інтегрованої модульної авіоніки є важливим кроком у напрямку забезпечення безпеки та надійності авіаційних систем. Цей алгоритм дозволяє систематизувати та автоматизувати процес контролю функціональних елементів обчислювальної системи під час польоту, що сприяє підвищенню безпеки та ефективності польоту.

Розроблене алгоритмічне та програмне забезпечення інформаційної підтримки процесів проектування та експлуатації бортового обладнання інтегрованої модульної авіоніки для верифікації створених методів та моделей. Вони дозволяють виконувати аналіз та експерименти з використанням розроблених методів та моделей, що сприяє покращенню якості та надійності бортового обладнання.

ВИСНОВКИ

1. Проведений аналіз інформаційної підтримку процесу проектування та експлуатації комплексу бортового обладнання (аналіз сучасних інструментів, програмного забезпечення, баз даних, методів аналізу даних, систем моніторингу та інше) який показав їх обмежені можливості відносно функціоналу, що спрямоване на забезпечення доступності, актуальності, цілісності, комплексності, надійності, зручності використання та захищеності інформації, таким чином, дозволив визначити задачі дослідження, які орієнтовані на побудову ефективних систем інформаційних технологій для авіаційної промисловості та засобів інформаційної підтримки для оптимізації процесів проектування та експлуатації комплексу бортового обладнання.

2. Розроблено метод підтримки проектування інтегрованої модульної авіоніки який дозволяє вдосконалити процес проектування комплексів бортового обладнання нового покоління, а саме підвищенню якості та ефективності процесу проектування, скороченню часу розробки та впровадженню нових технологій у виробництво авіаційної техніки, сприяючи покращенню конкурентоспроможності та інноваційності авіаційної промисловості.

3. Розроблена модель процесу проектування комплексу бортового обладнання інтегрованої модульної авіоніки для побудови математичних методів підвищення надійності та відмовостійкості, забезпечення якості та безпеки авіаційних систем. (Ця модель дозволяє систематизувати та формалізувати процес врахування аспектів надійності та відмовостійкості на ранніх етапах проектування, що дозволяє виявляти та усувати потенційні проблеми ще до виробництва та експлуатації. Результати застосування цієї моделі можуть бути корисними для розробників, інженерів та дослідників, що працюють у сфері авіаційних технологій, сприяючи підвищенню якості та надійності авіаційних систем та забезпецюючи безпеку польотів).

4. На базі запропонованих методів підтримки процесів проектування інтегрованої модульної авіоніки та розробленої моделі процесу проектування

комплексу бортового обладнання інтегрованої модульної авіоніки, дозволило розробити графову модель структури функцій комплексу бортового обладнання інтегрованої модульної авіоніки для реалізації обчислювальної потужності та навантаження на мережу передачі даних є важливим інструментом для аналізу та оптимізації роботи авіонічних систем. Ця модель дозволяє систематизувати та візуалізувати взаємозв'язки між різними функціями та параметрами системи, що допомагає зрозуміти їх вплив на обчислювальні та комунікаційні процеси.

5. Результати застосування цієї моделі можуть бути корисними для інженерів та дослідників, що працюють у сфері авіаційних технологій, для планування та оптимізації роботи бортових систем з урахуванням обмежень обчислювальної потужності та мережі передачі даних.

6. Розроблено структурна модель процесу проектування складних автоматизованих систем управління комплексу бортового обладнання є важливим інструментом для вирішення проблеми виявлення несправностей автоматизованих систем. Ця модель дозволяє систематизувати та візуалізувати взаємозв'язок між етапами проектування, аналізувати ситуації виявлення аналогій та помилок, а також ефективно повідомляти та формулювати звіти щодо виявлених проблем.

7. Розроблено структурно-логічна модель уніфікованого автоматизованого робочого місця для підвищення достовірності контролю обчислювальної системи інтегрованої модульної авіоніки є важливим кроком у напрямку забезпечення безпеки та надійності авіаційних систем. Ця модель дозволяє систематизувати та візуалізувати робоче місце, забезпечуючи операторам зручний інтерфейс для контролю та аналізу обчислювальної системи. Вона також дозволяє виявляти та усувати потенційні проблеми ще до виробництва та експлуатації, що покращує якість та надійність авіаційних систем.

Результати застосування цієї моделі можуть бути корисними для планування та оптимізації роботи бортових систем з урахуванням обмежень обчислювальної потужності та мережі передачі даних.

8. Розроблені методи оцінки ймовірності безвідмовної роботи структурно-логічної моделі уніфікованого автоматизованого робочого місця для контролю функціональних елементів обчислювальної інтегрованої модульної авіоніки на виробництві є важливим інструментом для забезпечення надійності та безпеки авіаційних систем. Ці методи дозволяють оцінити ймовірність виявлення та усунення потенційних проблем у роботі функціональних елементів ще до виробництва та експлуатації, сприяючи підвищенню якості та надійності авіаційних систем.

9. Розроблена інформаційна технологія для підвищення відмовостійкості бортового обладнання, забезпечення безпеки та надійності авіаційних систем. Ця технологія включає компоненти, такі як системи моніторингу та діагностики, програмне забезпечення для аналізу та передбачення відмов, системи автоматичного управління та резервування, а також методи аналізу даних для виявлення патернів та тенденцій у відмовах обладнання.

10. Розроблений алгоритм контролю обчислювальної системи інтегрованої модульної авіоніки під час польоту повітряного судна для контролю функціональних елементів та подальшої розробки програмного забезпечення у складі уніфікованого робочого місця обчислювальної системи інтегрованої модульної авіоніки є важливим кроком у напрямку забезпечення безпеки та надійності авіаційних систем. Цей алгоритм дозволяє систематизувати та автоматизувати процес контролю функціональних елементів обчислювальної системи під час польоту, що сприяє підвищенню безпеки та ефективності польоту.

11. Розроблене алгоритмічне та програмне забезпечення інформаційної підтримки процесів проектування та експлуатації бортового обладнання інтегрованої модульної авіоніки для верифікації створених

методів та моделей. Вони дозволяють виконувати аналіз та експерименти з використанням розроблених методів та моделей, що сприяє покращенню якості та надійності бортового обладнання.

Експериментальні дослідження: Проведені експериментальні дослідження з використанням розробленого забезпечення дозволили підтвердити ефективність та коректність створених методів та моделей. Це дає підставу для впровадження їх в практику проектування та експлуатації бортового обладнання.

Впровадження в практику: Результати досліджень та розробок можуть бути використані для покращення процесів проектування та експлуатації бортового обладнання інтегрованої модульної авіоніки. Впровадження розроблених методів та моделей може сприяти збільшенню надійності, ефективності та безпеки авіаційних систем.

СПИСОК ВИКОРИСТАННИХ ДЖЕРЕЛ

1. ДСТУ 2860-94. Надійність техніки. Терміни та визначення. К.: Держстандарт України, 1995, 91 с.

2. ДСТУ 2861-94. Надійність техніки. Аналіз надійності. Основні положення. К.: Держстандарт України, 1995, 32 с.

3. ДСТУ 2862-94. Надійність техніки. Методи розрахунку показників надійності. К.: Держстандарт України, 1995, 39 с.20.

4. ДСТУ 3004-95. Надійність техніки. Методи оцінки показників надійності за експериментальними даними. – К.: Держстандарт України, 1995. – 123 с. 21.

5. ДСТУ 3433-96. Надійність техніки. Моделі відмов. Основні положення. – К.: Держстандарт України, 1998. – 42 с. 22.

6. ДСТУ 3524-97. Надійність техніки. Проектна оцінка надійності складних систем з урахуванням технічного і програмного забезпечення та оперативного персоналу. Основні положення. – К.: Держстандарт України, 1999. – 21 с. 23.

7. ДСТУ 3942-2000. Надійність техніки. Плани випробувань для контролю середнього наробітку до відмови (на відмову). – К.: Держстандарт України, 2000. – 30 с.

8. Durak U., Rausch A., et al. Digital twin of iron bird test rig with flight simulator for testing avionics and system integration // DLR (German Aerospace Center), 2022. URL: https://elib.dlr.de/195403/

9. Li L., Digital Twin in Aerospace Industry: A Gentle Introduction, IEEE Access, 2021. DOI: 10.1109/ACCESS.2021.3136458.

10. Zhang W., Liu J., Cheng L., Filho R. S., Gao F., A Survey of Optimal Hardware and Software Mapping for Distributed Integrated Modular Avionics Systems, Applied Sciences, 2020. DOI: 10.3390/app10082675.

11. Priem R., Gagnon H., Chittick I. et al., An efficient application of Bayesian optimization to an industrial MDO framework for aircraft design, arXiv:2006.08434, 2020. https://arxiv.org/abs/2006.08434.

12. Xu Z., Saleh J.H., Machine Learning for Reliability Engineering and Safety Applications: Review of Current Status and Future Opportunities, arXiv:2008.08221, 2020. https://arxiv.org/abs/2008.08221.

13. О. Кучер, П. Власенко, Контроль та аналіз стану надійності систем і агрегатів повітряних суден в експлуатації, Наукоємні технології. К.: НАУ, №1(15), С. 15-26, 2010.

14. Konakhovych, G. “Research of variants of construction of a system of technical maintenance and repair in aircraft engineering”, Bulletin of the Engineering Academy of Ukraine, Issue 1, pp. 52-56, 2012.

15. Ramamoorthy C. V. A structural theory of machine diagnosis // Proc. of afips conf. 1968. Vol. 30. P. 743{756.

16. Nakagawa, T. “Two-unit redundant models, Stochastic Models in Reliability and Maintenance”, N.Y.: Springer, Issue 1, pp. 165-185, 2002.

17. Li L., Aslam S., Wileman A., Perinpanayagam S. «Digital Twin in Aerospace Industry: A Gentle Introduction», IEEE Access, 2021, Vol. 9, pp. 167845–167859. DOI: 10.1109/ACCESS.2021.3136458.

18. Ghanjaoui Y., Fuchs M., Biedermann J., Nagel B.«Model-based design and multidisciplinary optimization of complex system architectures in the aircraft cabin», CEAS Aeronautical Journal, 2023, Vol. 14(4), pp. 887–905. DOI: 10.1007/s13272-023-00683-w.

19. Konakhovych, G. “Research of variants of construction of a system of technical maintenance and repair in aircraft engineering”, Bulletin of the Engineering Academy of Ukraine, Issue 1, pp. 52-56, 2012.

20. Міляєв Ю., Нечипоренко О. , Основи надійності технічних систем: навч. посіб. К.: Видавн.-полігр. центр Акад. муніцип. управління, 2008, 246 с

21. Kucher, O., Vlasenko, P., “Comparative Analysis of Reliability and Efficiency Indicators in Foreign and National Aviation”, Knowledge-based Technology, Issue 2, pp. 11-19, 2009.

22. Kucher, O., Vlasenko, P., “Management of reliability of airline

aircraft fleet", Aerospace Technology and Technology, Issue 4, pp. 88-94, 2009.

23. Liu, J.W.S.W., Real-Time Systems, Prentice Hall PTR, Upper Saddle River, NJ, USA, 1st edn. 2000.

24. Kermia, O., Sorel, Y. "Schedulability Analysis for Non-Preemptive Tasks under Strict Periodicity Constraints", Proceedings of the 2008 14th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications, pp. 25-32, 2008.

25. Yomsi, P., Sorel, Y. "Schedulability Analysis for non-Necessarily Harmonic Real-Time Systems with Precedence and Strict Periodicity Constraints using the Exact Number of Preemptions and no Idle Time", Proceedings of the 4th Multidisciplinary International Scheduling Conference, MISTA'09, Dublin, Ireland, August, 2009.

26. IEEE Standard 1149.1–2013. IEEE Standard for Test Access Port and Boundary-Scan Architecture. – New York: The Institute of Electrical and Electronics Engineers, Inc., 2013. – 546 p.

27. Wasouf S.S., Ahmed H.N., El-Mahlawy M.H., et al. "A Proposed Boundary Scan Testing Module for Automatic Testing of Digital Integrated Circuits", Aerospace Sciences & Aviation Technology (ASAT-13), 2009.

28. IEEE Std 1149.1-2001. IEEE Standard Test Access Port and Boundary-Scan Architecture / The Institute of Electrical and Electronics Engineers, Inc., New York, 2001.

29. Bleeker H., Eijnden P., Jong F. Boundary-Scan Test: A Practical Approach. {{ Dordrecht, The Netherlands : Kluwer academic publishers, 1993.

30. Watkins, C. B., and Walter, R. Transitioning from federated avionics architectures to integrated modular avionics. In 2007 IEEE/AIAA 26th Digital Avionics Systems Conference (DASC), Dallas, TX, Oct.21–25, 2007.

31. Gaska, T. Optimizing an incremental modular open system approach (MOSA) in avionics systems for balanced architecture decisions. In 2012 IEEE/AIAA 31st Digital Avionics Systems Conference (DASC),Williamsburg, VA, Oct. 14–18, 2012.

32. RTCA, DO-297 Integrated Modular Avionics (IMA) Development Guidance and Certification Considerations, 2005.

33. RTCA. DO-178C: Software Considerations in Airborne Systems and Equipment Certification. – Washington, D.C.: RTCA Inc., 2011. – 336 c.

34. RTCA. DO-178B: Software Considerations in Airborne Systems and Equipment Certification. – Washington, D.C.: RTCA Inc., 1992. – 228 c.

35. SAE, ARP4754, The Aerospace Recommended Practice (ARP) (Guidelines For Development Of Civil Aircraft and Systems), 2010.

36. Gaska, T. Lockheed Martin Common CockpitTM Avionics Suite A Modular Open Systems Approach (MOSA) Driven Architecture. In American Helicopter Society 63rd Annual Forum, Virginia Beach, VA, May 1–3, 2007.

37. Spitzer, C. The Avionic Handbook (1st ed.). Boca Raton, FL: CRC, 2001, ch. 29.ARINC Specification 664P7, Aircraft Data Network Part 7 Avionics Full Duplex Switched Ethernet (AFDX) Network, 2005.

38. Cui, Q., & Li, Y. (2015). The change trend and influencing factors of civil aviation safety efficiency: the case of Chinese airline companies. Safety science, 75, 56-63, https://doi.org/10.1016/j.ssci.2015.01.015.

39. 7. Denney, E., Pai, G., & Whiteside, I. (2019). The role of safety architectures in aviation safety cases. Reliability Engineering & System Safety, 191, 106502. https://doi.org/10.1016/j.ress.2019.106502.

40. Greenberg, R., Cook, S. C., & Harris, D. (2005). A civil aviation safety assessment model using a Bayesian belief network (BBN). The Aeronautical Journal, 109(1101), 557-568, https://doi.org/10.1017/S0001924000000907.

41. Janic, M. (2000). An assessment of risk and safety in civil aviation. Journal of Air Transport Management, 6(1), 43-50, https://doi.org/10.1016/S0969-6997(99)00021-6.

42. Goncharenko A. V.Development of a theoretical approach to the conditional optimization of aircraft maintenance preference uncertainty // Aviation, 2018. – Vol. 22, No 2. – P. 40–44. – DOI: 10.3846/aviation.2018.5929

43. Kasianov V. A., Goncharenko A. V. A multi-optional hybrid functions entropy as a tool for transportation means repair optimal periodicity determination // Aviation, 2018. – Vol. 22, No 2. – P. 60–66. – DOI: 10.3846/aviation.2018.5930

44. Łuczak K. Zarządzanie bezpieczeństwem w lotnictwie cywilnym. – Katowice: Uniwersytet Śląski, 2016. – 350 с. – [Електронний ресурс].: https://sbc.org.pl/Content/218989/zarzadzanie_bezpieczenstwem_w_lotnictwie.pdf

45. Menon, P. K., Dutta, P., Chen, O., Iyer, H., & Yang, B. J. (2019). A modeling environment for assessing aviation safety. In AIAA Aviation 2019 Forum (p. 2937), https://doi.org/10.2514/6.2019-2937.

46. Netjasov, F., & Janic, M. (2008). A review of research on risk and safety modelling in civil aviation. Journal of Air Transport Management, 14(4), 213-220, https://doi.org/10.1016/j.jairtraman.2008.04.008.

47. Annighöfer B., Reinhart J., Brunner M., Schulz B. «The Concept of an Autonomic Avionics Platform », arXiv:2103.12065, 2021. URL: https://arxiv.org/abs/2103.12065.

48. Zhao Y., Yang Z., Sanán D., Liu Y. «Event-based Formalization on ARINC 653 using Event-B», arXiv:1508.06479, 2015. URL: https://arxiv.org/abs/1508.06479.

49. Zhang W., Liu J., Cheng L., Filho R. S., Gao F. «A Survey of Optimal Hardware and Software Mapping for Distributed Integrated Modular Avionics Systems», Applied Sciences, 2020, Vol. 10(8):2675. DOI: 10.3390/app10082675

50. SAE AS6802, Time-Triggered Ethernet, Nov. 2011.

51. Jakovljevic, M. 2nd generation IMA: extended virtualization capabilities for optimized architectures. In 2013 IEEE/AIAA 32nd Digital Avionics Systems Conference (DASC), East Syracuse, NY, Oct. 5–10, 2013.

52. Priem R., Gagnon H., Chittick I., Dufresne S., Diouane Y., Bartoli N. «An efficient application of Bayesian optimization aircraft design», arXiv: 2006.08434, 2020. URL: https://arxiv.org/abs/2006.08434.

53. Han P., Zhai Z., Nielsen B., Nyman U.«A Modeling Framework for Schedulability Analysis of Distributed Avionics Systems», Proceedings UPPAAL Conf., 2018.

54. Watkins, C. B., and Walter, R. Design considerations for systems hosted on integrated modular avionics platforms. In 2008 IEEE/AIAA 27th Digital Avionics Systems Conference (DASC), St. Paul, MN, Oct. 26–30, 2008.

55. AIAA «The Digital Twin Paradigm for Aircraft: Review and Outlook», AIAA Scitech Forum, Paper 0553, 2020. DOI:10.2514/6.2020-0553.

56. Gaska, T. Selecting components, standards, and topologies for next generation integrated modular avionics (IMA) architectures. In American Helicopter Society 69th Annual Forum, Phoenix, AZ, May 21–23, 2013.

57. Gaska, T., Werner, B., and Flagg, D. Applying virtualization to avionics systems – the integration challenges. In 2010 IEEE/AIAA 29th Digital Avionics Systems Conference (DASC), Salt Lake City, UT, Oct. 3–7, 2010.

58. Hilbrich, R., and Dieudonné, L. Deploying safety-critical applications on complex avionics hardware architectures. Journal of Software Engineering and Applications, Vol. 6, 5 (2013), 229–235.

59. Konakhovych, G., Kozlyuk, I. “Optimization of Operation of Advanced Aircraft by Technical and Economic Criteria”, Problems of Informatization and Control, Kyiv: Press of the NAU, pp. 79-85, 2005.

60. Kovalenko, Yu., Rybalka, L., Burlaka, M. “Analysis of the efficient functioning of the software in information control systems”, Measuring and Computing Engineering in Technical Processes, Issue 1, pp. 95-100, 2016.

61. Y. Kovalenko, H. Konakhovych, I. Kozlyuk, «Specificity of optimization of performance indicators of technical operation and updating of radio electronic systems of aircraft», International Journal of Engineering Research and Applications (IJERA), vol. 10(09), pp. 48-58, 2020.

62. Ю. Коваленко, Л. Рибалка, М. Бурлака, «Вдосконалення автоматизованої системи управління технологічної підготовки виробництва в авіаційній галузі», Вісник хмельницького національного університету.

Технічні науки, № 4, С. 209-214, 2018.

63. Ю. Коваленко, Л. Рибалка, М. Бурлака, «Аналіз надійності функціонування програмного забезпечення інформаційно-управляючих систем», Вимірювальна та обчислювальна техніка в технологічних процесах, № 4, С. 74-81, 2017.

64. Ю. Коваленко, «Модель захищеної автоматизованої системи на підприємстві», Вимірювальна та обчислювальна техніка в технологічних процесах, № 2, С. 68-72, 2017.

65. Коваленко Ю., Козлюк І., «Проектування технологічних процесів інформаційної системи підтримки прийняття рішень», Збірник наукових праць Військового інституту Київського національного університету імені Тараса Шевченка, № 61, С. 14-23., 2018.

66. Коваленко Ю., Козлюк І., «Структурний аналіз технологічного проектування і управління технологічним проектом у авіаційній галузі», Збірник наукових праць Військового інституту Київського національного університету імені Тараса Шевченка, № 59, С. 22-31, 2018.

67. Committee, AE ARINC 664 Aircraft Data Networks, Part7: Avionics Full Duplex Switched Ethernet (AFDX) Network; Aeronautical Radio, Inc.: Annapolis,MD,USA,2005.doi:https://global.ihs.com/doc_detail.cfm?document_name=ARINC%20664% 20P7 & item_s_key = 00465045.

68. ISO/IEC 11801:2017 Information technology – Generic cabling for customer premises. – 6th ed. – Geneva: International Organization for Standardization, 2017. – 221 p.

69. ISO/IEC 13961:1995 Information technology – High-performance serial bus – Scalable Coherent Interface (SCI). – Geneva: ISO/IEC, 1995. – 119 p.

70. Y. Kovalenko, «Methods of developing integrated modular avionics systems», Scientific Journal of Polonia University, no. 43(6), pp. 324-334, 2020.

71. Y. Kovalenko, I. Kozlyuk, «Functional methods of developing integrated modular avionics systems», The Bulletin of Zaporizhzhia National University: Physical and mathematical Sciences, no. 1, pp101-116., 2021

72. Y. Kovalenko, I. Kozlyuk, «Implementation of the integrated modular avionics application development complex according to the ARINC653 standard», The Bulletin of Zaporizhzhia National University: Physical and mathematical Sciences, no. 2, pp28-36, 2020.

73. ARINC Specification 653. Avionics Application Software Standard Interface. URL: https://www.sae.org/standards/content/arinc653p3a-1/

74. DO-178B Software Considerations in Airborne Systems and Equipment Certification. Dept. of Measurement and Information Systems. URL: https://inf.mit.bme.hu/sites/default/files/materials/taxonomy/term/445/13/13_CES_DO-178B.pdf (дата ссылки: 07.03.2020)

75. RTCA DO-255 / EUROCAE ED-96. Requirements Specification for Avionics Computer Resource (ACR). URL: https: // standards. globalspec. com / std / 1968378 / RTCA% 20DO-255.

76. Y. Kovalenko, I. Kozlyuk «Методи оцінки показників надійності авіаційної техніки за результатами випробувань та технічної діагностики», Наукоємні технології, № 3(39), С. 367-375, 2018.

77. Ю. Коваленко, «Нейросетевые технологии принятия решений управления сложными социотехническими системами (ССТС) в реальном времени, Вимірювальна та обчислювальна техніка в технологічних процесах, № 1, С. 95-100, 2016.

78. Parkinson P., Kinnan L. Safety-Critical Software Development for Integrated Modular Avionics. Wind River. 2015. Vol. 11. No.2. URL: https://events.windriver.com/wrcd01/wrcm/2015/02/Safety-Critical-Software-Development-for-Integrated-Modular-Avionics-White-Paper-1.pdf

79. Kovalenko, Y., «A programmable logic controller (PLC); programming language structural analysis», Advances in Intelligent Systems and Computingthis link is disabled, С.234–242, 2017.

80. Hayley J., Reynolds R., Lokhande K., Kuffner M., Yenson S. (2012) Human-Systems Integration and Air Traffic Control. Lincoln laboratory journal, vol. 19, no. 1, pp. 34-49.

81. Parkinson P., Kinnan L. (2015) Safety-Critical Software Development for Integrated Modular Avionics. Wind River, vol. 11, no. 2.

82. Tiedeman H. and Parkinson P. (2019) Experiences of Civil Certification of Multi-Core Processing Systems in Commer-cial and Military Avionics, Integration Ac-tivities, and Analysis. SAE Int. J. Adv. & Curr. Work. in Mobility, no. 1(2), pp. 419-428. doi: https://doi.org/10.3182/20110828-6- it-1002.01501.

83. Ghannem A., Hamdi M.S., Kes-sentini M., Ammar H.H. (2017) Searchbased requirements traceability recovery: A multi-objective approach. Proc. IEEE Congress on Evolutionary Computation (CEC), pp. 1183-1190.

84. Ю. Коваленко, «Системний аналіз проблеми багатокритеріального вибору варіанту удосконалення системи захисту інформації», Захист інформації, Т. 17, № 1, С. 38-43, 2015.

85. Ю. Коваленко, «Комплексний підхід до управління корпоративним інформаційним контентом», Безпека інформації, Т. 21, № 1, С.70-76, 2015.

86. IEEE Std 1149.1-2001. IEEE Standard Test Access Port and Boundary-Scan Architecture / The Institute of Electrical and Electronics Engineers, Inc., New York, 2001.

87. Jiang Z., Zhao T, Wang S., Ju H.(2020) New Model-Based Analysis Meth-od with Multiple Constraints for Integrated Modular Avionics Dynamic Reconfiguration Process. Processes, no. 8, pp. 574. doi.org/10.3390/pr8050574.

88. Athavale J., Mariani R. and Pau-litsch M. (2019) Flight Safety Certification Implications for Complex Multi-Core Pro-cessor Based Avionics Systems, 2019 IEEE International Reliability Physics Symposium (IRPS), Monterey, CA, USA, pp. 1-6. doi: 10.1109 / IRPS.2019.8720422.

89. Montano G., McDermid J. (2008) Human Involvement in Dynamic Reconfigu-ration of Integrated Modular Avionics, Avi-onics. In Proceedings of the 27th Digital Avionics Systems Conference, St. Paul, MN, USA, 26–30 October 2008; IEEE: Pisca-taway, NJ, USA. doi: https://ieeexplore. ieee.org/abstract/ document/4702821

90. Bieber, P., Boniol, F., Boyer, M., Noulard, E., and Pagetti, C. New challenges for future avionics architectures. Journal Aerospace Labs, Vol. 4 (May 2012).

91. DuBois, T., Kinahan, W., and Dones, F. Joint common architecture (JCA) recommendations. In American Helicopter Society (AHS) 70th Forum, Montreal, Canada, May 20–22, 2014.

92. Gaska, T. An affordable IMA bridge for refreshing deployed avionics systems. In 2014 IEEE/AIAA 33rd Digital Avionics Systems Conference (DASC), Colorado Springs, CO, Oct. 5–9, 2014.

93. Ю. Коваленко, «Метод захисту інформації в комп'ютерних мережах промислових підприємств», Вісник Інженерної академії України, № 4, С. 93-97, 2016.

94. Системотехніка та основи проектування аеронавігаційних систем [Текст] : навч. посіб. для студ. вищих навч. закл. / М. І. Валуєв [и др.] ; Національний авіаційний ун-т. – К. : НАУ, 2003. - 115 с.

95. Принципи проектування програмного забезпечення оперативного контролю польотів повітряних суден за інформацією бортових параметричних реєстраторів [Текст] : навч. посібник / Ю. К. Зіатдінов [и др.]; Національний авіаційний ун-т. - К. : НАУ, 2008. - 116 с.

96. Goldberg D E, Genetic algorithms in search, optimization and machine learning, Addison-Wesley Publishing Company, 1989.

97. Rasheed K, Hirsh H & Gelsey A, A genetic algorithm for continuous design space search, Artif Intell Eng, 11 (1997) 295-305.

98. Bos A H W, Aircraft conceptual design by genetic/gradientguided optimization, Eng Appl Artif Intel, 11(3) (1998) 377-382.

99. Y. Kovalenko, H. Konakhovych, I. Kozlyuk, «Specificity of optimization of performance indicators of technical operation and updating of radio electronic systems of aircraft», International Journal of Engineering Research and Applications (IJERA), vol. 10(09), pp. 48-58, 2020.

100. Y. Kovalenko, I. Kozlyuk, «Designing of technological processes of decision-making support information system in integrated modular avionics». Problems of Informatization and Management, no. 64, pp. 4-15, 2020.

101. I. Kozlyuk, Y. Kovalenko, «Functional bases of the software development and operation in avionics». Problems of Informatization and Management, no. 63, pp. 49-63, 2020.2

102. Y. Kovalenko, «Experiment test of the reliability of computer systems of integrated modular avionics». Problems of Informatizati on and Management, no. 65, pp. 45-53, 2021.5

103. Y. Kovalenko, « Methodology For Testing Languages For Embedded Avionics Systems ». Problems of Informatizati on and Management, no. 63, pp. 44 – 52 , 2022.

104. H. Konakhovych, I. Kozlyuk, Y. Kovalenko, «Specificity of optimization of performance indicators of technical operation and updating of radio electronic systems of aircraft», System research and information technologies, no. 3, pp. 41-54, 2020.

105. I. Kozlyuk, Y. Kovalenko, «Reliability of computer structures of integrated modular aviation for hardware configurations», System research and information technologies, no. 3, pp. 84-94, 2021.

106. V. Lakhno, S. Kazmirchuk, Y. Kovalenko, L. Myrutenko, T. Zhmurko, «Design of adaptive system of detection of cyber-attacks, based on the model of logical procedures and the coverage matrices of features», Eastern-European Journal of Enterprise Technologies, no. 3(9), pp. 30-38, 2016.

107. Ю. Коваленко, «Модель захищеної автоматизованої системи управління на підприємстві», Вісник Інженерної академії України, № 2, С. 68-72, 2017.

108. Ю. Коваленко, «Система тестування на проникнення комп’ютерних мереж та систем», Вісник Інженерної академії України, № 1, С. 167-170, 2015.

109. В. Козачок, Ю. Коваленко, «Особливості побудови комплексних систем захисту інформації в розподілених корпоративних мережах», Сучасний захист інформації, № 1, С. 41-47, 2015.

110. Ю. Коваленко, «Захишенний застосунок на основі технології VOIP», Вісник Інженерної академії України, № 2, С. 27-30, 2015.

111. А. Корченко, С. Казмирчук, А. Гололобов, Ю. Коваленко, «Метод инкрементирования порядка лингвистических переменных для систем анализа и оценивания риска», Захист інформації, Т. 17, №2, С. 101-111, 2015.

112. Ю. Коваленко, Н. Вишневська, І. Мішина, А. Дзюбаненко, «Структурний аналіз мов програмування захищених застосунків для програмованих логічних контролерів», Безпека інформації, Т. 22, № 1, С. 38-43, 2016.

113. Kovalenko Yu. B., Kozlyuk I. O. Method of developing automated information support for integrated modular avionics systems: monitoring, diagnostics and reliability improvement // Visnyk of the National Aviation University. – 2024. – Vol. 80. – [40-52pp.]. DOI: https://doi.org/10.18372/2073-4751.80.19768.

114. Kovalenko Yu., Shmatko O., Herasymov S., Milevskyi S., Balitskyi N., Pohasii S., Aleksiiev M., Vlasov I., Melenti Y., Peleshok Y. Development of a method for assessing the efficiency of technical systems' computer dynamic simulators // Eastern-European Journal of Enterprise Technologies. – 2025. – Vol. 2, No. 9 (134). – P. 50–64. – DOI: https://doi.org/10.15587/1729-4061.2025.327558.

115. Kovalenko Yu., Benzar A., Taranenko A., Balynska O., Balynskyi I. Contemporary strategies for managing organisational information security // Journal of Information Systems Engineering & Management. – 2024. – Vol. 10, No. 12(s). 505-513 pp. – DOI: https://doi.org/10.52783/jisem.v10i12s.1859.

116. Kovalenko Yuliia, Kovalchuk Olena. Zero Trust Security: Treats to Information Security in the Context of National Security // International and

National Security: Theoretical and Applied Aspects: Proceedings of the 9th International Scientific and Practical Conference (Dnipro, March 14, 2025). – Dnipro: Dnipropetrovsk State University of Internal Affairs, 2025. – [P. 41–43].

117. Fielding J P, Introduction to aircraft design. Cambridge University Press UK, 1999.

118. Gundlach J F, Multidisciplinary design optimization of subsonic fixed-wing unmanned aerial vehicles projected through 2025, PhD Dissertation, Virginia Polytechnic Institute and State University, Blacksburg, Virginia, 2004.

119. Sobester A & Keane A J, Multidisciplinary design optimization of UAV airframes, in 47th AIAA/ASME/ASCE/AHS/ASC Str Struct Dy Mat Conf, Newport, Rhode Island, USA, AIAA 2006- 1612, 1-4 May 2006, 13 pages.

120. Kroo I, Aircraft Design: synthesis and analysis, Desktop Aeronautics Inc, Stanford, CA, 2003. 8 Torenbeek E, Advanced Aircraft Design: Conceptual Design, Technology and Optimization of Subsonic Civil Airplanes, John Wiley & Sons, 2013

121. Bos A H W, Aircraft conceptual design by genetic/gradientguided optimization, Eng Appl Artif Intel, 11(3) (1998) 377-382.

122. Obayashi S, Multidisciplinary design optimization of aircraft wing planform based on evolutionary algorithms, in Proc of the 1998 IEEE Int Conf Syst Man Cyb, La Jolla, California, USA, 11-14 October 1998, 3148-3153.

123. Parmee I C & Watson A H, Preliminary airframe design using co-evolutionary multi objective genetic algorithms, in GECCO–99: Proc Gen Evol Computat Conf (Banzhaf W et al. eds), Orlando, Florida, USA, Morgan Kaufmann, July 1999, 1657-1665.

124. Jun C L & Song A H, Conceptual / Preliminary aircraft design using genetic algorithm and fuzzy mathematics, In 37th AIAA Aer Sci M&E, AIAA-1999-0113, 11-14 January 1999.

125. Ng T T H & Leng G S B, Application of genetic algorithms to conceptual design of a micro-air vehicle, Eng Appl Artif Intel, 15 (2002) 439-445.

126. Bandte O & Malinchik S, A broad and narrow approach to interactive evolutionary design – an aircraft design example, in Proc Gen Evol Computat Conf - GECCO 2004 (DebK et al. (eds.)), Lecture Notes in Computer Science Volume, 3103, 2004, 883-895.

127. Kroo I, Aeronautical applications of evolutionary design, in VKI Lecture Series on Optimization Methods & Tools for Multi-Criteria / Multi-Disciplinary Des, November 15-19, 2004, 11 pages.

128. Ghorbany A & Malek M B, Airplane Conceptual Design Based on Genetic Algorithm, Mech & Aerospace Eng J, 1(1) (2005) 101-114.

129. Marta A, Parametric study of a genetic algorithm: using an aircraft design optimization problem, Report Stanford University, Department of Aeronautics and Astronautics, 2008.

130. Davis L, Handbook of genetic algorithms, New York, Van Nostrand Reinhold, 1991.

ДОДАТОК А.

Акти впровадження наукових досліджень дисертаційної роботи

ЗАТВЕРДЖУЮ
Проректор з навчальної роботи
Національного авіаційного університету
_____ Анатолій ПОЛУХІН
«20» 05 2024 р.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ * НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ * 01132330 * МІСТО КИЇВ

АКТ ВПРОВАДЖЕННЯ

у навчальний процес Національного авіаційного університету результатів дисертаційної роботи Коваленко Юлії Борисівни «Інформаційна підтримка процесу проектування та експлуатації комплексу бортового обладнання інтегрованої модульної авіоніки» на здобуття наукового ступеня доктора технічних наук.

Комісія у складі: голови – завідувача кафедри телекомунікаційних та радіоелектронних систем (ТКРС) Гнатюка В.О., члени комісії – професор кафедри ТКРС Конахович Г.Ф., доцент кафедри ТКРС Чуприн В.М. та доцент кафедри ТКРС Зуєв О.В., склали цей акт про те, що результати дисертаційної роботи Коваленко Ю. Б. «Інформаційна підтримка процесу проектування та експлуатації комплексу бортового обладнання інтегрованої модульної авіоніки» впроваджені у навчальний процес і використовуються на кафедрі ТКРС при викладанні наступної дисципліни: «Методи забезпечення надійності та ефективності експлуатації сучасних телекомунікаційних та радіотехнічних систем», «Основи теорії надійності, експлуатації та ремонту інформаційно-телекомунікаційних та радіотехнічних систем». Дана дисципліна викладається при підготовці докторів філософії та бакалаврів відповідно за спеціальністю 172 «Електронні комунікації та радіотехніка».

№ з/п	Назва роботи, що впроваджується	Дисципліна, форма впровадження	Ефективність від впровадження
	1	2	3
1	Методи підвищення відмовостійкості та ефективності контролю і діагностики бортового радіоелектронного обладнання	«Основи теорії надійності, експлуатації та ремонту інформаційно-телекомунікаційних та радіотехнічних систем» Лекційні заняття	Ознайомлення студентів з методами та моделями підвищення відмовостійкості, та ефективності контролю і діагностики бортового радіоелектронного обладнання
2	Метод оцінки надійності та ефективності експлуатації обчислювальних структур бортового радіоелектронного обладнання.	«Методи забезпечення надійності та ефективності експлуатації сучасних телекомунікаційних та радіотехнічних систем» Практичні заняття	Ознайомлення студентів з методом та засобами реалізації систем підвищення відмовостійкості, контролю і діагностики радіоелектронного обладнання.

Голова комісії,
завідувач кафедри ТКРС — Віктор ГНАТЮК

Члени комісії:

професор кафедри ТКРС — Георгій КОНАХОВИЧ

доцент кафедри ТКРС — Володимир ЧУПРИН

доцент кафедри ТКРС — Олексій ЗУЄВ

ТОВ "СІТОН ДІДЖИТАЛ"
вул. Гарматна, 4, 4-й поверх, офіс 4,
Київ, Україна, 03067
+380 44 239 99 99
info@seeton.digital

ЗАТВЕРДЖУЮ
Директор товариства з обмеженою відповідальністю «СІТОН ДІДЖИТАЛ»

Дмитро НІКІТЮК
(підпис)

“ 15 ” 05 2025 року
М.П.

АКТ

Комісія у складі:
голови – директор, Нікітюк Дмитро Вікторович;
членів комісії: – заступник керівника відділу систем інформаційної безпеки, Коровайченко Юрій Юрійович;
– провідний фахівець з інформаційної безпеки, Костіна Єлизавета Василівна.

цим Актом засвідчує, що результати дисертаційного дослідження доцента кафедри Кібербезпеки Державного некомерційного підприємства «Державний університет «Київський авіаційний інститут».

Коваленко Юлії Борисівни

на тему: «Інформаційна підтримка процесу проектування та експлуатації комплексу бортового обладнання інтегрованої модульної авіоніки» на здобуття наукового ступеня доктора технічних наук за спеціальністю 05.13.06 «Інформаційні технології» впроваджені в діяльності товариства з обмеженою відповідальністю **«СІТОН ДІДЖИТАЛ».**

Зокрема, під час створення та впровадження апаратно-програмного комплексу для інформаційної підтримки процесу проєктування та експлуатації комплексу бортового обладнання інтегрованої модульної авіоніки, орієнтованого на потреби підприємств авіаційної галузі, використано аналітичні та прикладні результати дисертаційної роботи Коваленко Юлії Борисівни, які:

- реалізують інформаційну технологію, що включає алгоритмічні засоби обчислення прогнозованих похибок у системах авіоніки, запобігають виходу параметрів за межі стійкої роботи та сприяють підвищенню надійності й відмовостійкості функціональних елементів обладнання;
- забезпечують створення засобів контролю та діагностики обчислювальної системи інтегрованої модульної авіоніки, яка функціонує як у стандартних, так і в особливих умовах експлуатації повітряних суден, із реалізацією засобів програмного, інформаційно-вимірювального та алгоритмічного забезпечення.

Голова комісії Директор _____ Дмитро НІКІТЮК
(підпис)

Члени комісії Заступник керівника відділу систем інформаційної безпеки _____ Юрій КОРОВАЙЧЕНКО
(підпис)

Провідний фахівець З інформаційної безпеки _____ Єлизавета КОСТІНА
(підпис)

ДОДАТОК Б.

Список опублікованих праць за темою дисертації

1. Kovalenko Y. Methods of developing integrated modular avionics systems. Scientific Journal of Polonia University. 2020. No. 43(6). P. 324–334.

2. Kovalenko Y., Kozlyuk I. Designing of technological processes of decision-making support information system in integrated modular avionics. Problems of Informatization and Management. 2020. No. 64. P. 4–15.

3. Kozlyuk I., Kovalenko Y. Functional bases of the software development and operation in avionics. Problems of Informatization and Management. 2020. No. 63. P. 49–63.

4. Kovalenko Y. Experiment test of the reliability of computer systems of integrated modular avionics. Problems of Informatization and Management. 2021. No. 65. P. 45–53.

5. Kovalenko Y., Kudrenko S. Methodology for testing languages for embedded avionics systems. Problems of Informatization and Management. 2022. No. 63. P. 44–52.

6. Конахович Г. Ф., Козлюк І. О., Коваленко Ю. Б. Оптимізація показників ефективності організаційної системи технічної експлуатації та оновлення радіоелектронних систем повітряних суден. Системні дослідження та інформаційні технології. 2020. № 4. С. 115–131.

7. Козлюк І. О., Коваленко Ю. Б. Надійність обчислювальних структур інтегрованої модульної авіоніки для конфігурацій апаратних засобів. Системні дослідження та інформаційні технології. 2021. № 2. С. 84–93.

8. Kovalenko Y., Kozlyuk I. Implementation of the integrated modular avionics application development complex according to the ARINC 653 standard. The Bulletin of Zaporizhzhia National University: Physical and Mathematical Sciences. 2020. No. 2. P. 28–36.

9. Kovalenko Y., Kozlyuk I. Functional methods of developing integrated modular avionics systems. The Bulletin of Zaporizhzhia National University: Physical and Mathematical Sciences. 2021. No. 1. P. 101–116.

10. Kovalenko Y., Kozlyuk I. Методи оцінки показників надійності авіаційної техніки за результатами випробувань та технічної діагностики. Наукоємні технології. 2018. № 3(39). С. 367–375.

11. Lakhno V., Kazmirchuk S., Kovalenko Y., Myrutenko L., Zhmurko T. Design of adaptive system of detection of cyber-attacks, based on the model of logical procedures and the coverage matrices of features. Eastern-European Journal of Enterprise Technologies. 2016. No. 3(9). P. 30–38.

12. Коваленко Ю., Рибалка Л., Бурлака М. Модель захищеної автоматизованої системи управління на підприємстві. Вісник Інженерної академії України. 2017. № 2. С. 68–72.

13. Коваленко Ю. Система тестування на проникнення комп’ютерних мереж та систем. Вісник Інженерної академії України. 2015. № 1. С. 167–170.

14. Коваленко Ю. Системний аналіз проблеми багатокритеріального вибору варіанту удосконалення системи захисту інформації. Захист інформації. 2015. Т. 17. № 1. С. 38–43.

15. Коваленко Ю. Комплексний підхід до управління корпоративним інформаційним контентом. Безпека інформації. 2015. Т. 21. № 1. С. 70–76.

16. Козачок В., Коваленко Ю. Особливості побудови комплексних систем захисту інформації в розподілених корпоративних мережах. Сучасний захист інформації. 2015. № 1. С. 41–47.

17. Коваленко Ю. Захищений застосунок на основі технології VOIP. Вісник Інженерної академії України. 2015. № 2. С. 27–30.

18. Корченко А., Казмирчук С., Гололобов А., Коваленко Ю. Метод інкрементування порядку лінгвістичних змінних для систем аналізу та оцінювання ризику. Захист інформації. 2015. Т. 17. № 2. С. 101–111.

19. Коваленко Ю., Вишневська Н., Мішина І., Дзюбаненко А. Структурний аналіз мов програмування захищених застосунків для програмованих логічних контролерів. Безпека інформації. 2016. Т. 22. № 1. С. 38–43.

20. Коваленко Ю. Нейромережеві технології прийняття рішень управління складними соціотехнічними системами (ССТС) в реальному часі. Вимірювальна та обчислювальна техніка в технологічних процесах. 2016. № 1. С. 95–100.

21. Коваленко Ю. Комп'ютеризована система захищеного застосування на основі технології VOIP. Збірник наукових праць ОДАТРЯ. Одеса, 2016. С. 91–95.

22. Коваленко Ю. Аналіз оцінки надійності програмного забезпечення. Вісник Інженерної академії України. 2016. № 4. С. 93–97.

23. Kovalenko Y. A programmable logic controller (PLC); programming language structural analysis. Advances in Intelligent Systems and Computing. 2017. P. 234–242.

24. Коваленко Ю., Рибалка Л., Бурлака М. Вдосконалення автоматизованої системи управління технологічної підготовки виробництва в авіаційній галузі. Вісник Хмельницького національного університету. Технічні науки. 2018. № 4. С. 209–214.

25. Коваленко Ю., Рибалка Л., Бурлака М. Аналіз надійності функціонування програмного забезпечення інформаційно-управляючих систем. Вимірювальна та обчислювальна техніка в технологічних процесах. 2017. № 4. С. 74–81.

26. Козлюк І., Коваленко Ю. Проектування технологічних процесів інформаційної системи підтримки прийняття рішень. Збірник наукових праць Військового інституту Київського національного університету імені Тараса Шевченка. 2018. № 61. С. 14–23.

27. Козлюк І., Коваленко Ю. Структурний аналіз технологічного проектування і управління технологічним проектом у авіаційній галузі.

Збірник наукових праць Військового інституту Київського національного університету імені Тараса Шевченка. 2018. № 59. С. 22–31.

28. Kovalenko Yu. B., Kozlyuk I. O. Method of developing automated information support for integrated modular avionics systems: monitoring, diagnostics and reliability improvement. Visnyk of the National Aviation University. 2024. Vol. 80. P. 40–52. DOI: https://doi.org/10.18372/2073-4751.80.19768.

29. Shmatko O., Herasymov S., Milevskyi S., Balitskyi N., Pohasii S., Aleksiiev M., Vlasov I., Melenti Y., Kovalenko Yu., Peleshok Y. Development of a method for assessing the efficiency of technical systems' computer dynamic simulators. Eastern-European Journal of Enterprise Technologies. 2025. Vol. 2, No. 9(134). P. 50–64. DOI: https://doi.org/10.15587/1729-4061.2025.327558.

30. Benzar A., Kovalenko Yu., Taranenko A., Balynska O., Balynskyi I. Contemporary strategies for managing organisational information security. Journal of Information Systems Engineering & Management. 2024. Vol. 10, No. 12(s). P. 505–513. DOI: https://doi.org/10.52783/jisem.v10i12s.1859.

Відомості про апробацію результатів дисертації

1. Коваленко Ю. Б., Савчук М. М. Система інтелектуального фаззінгу. Безпека інформаційних технологій (ITSEC): тези доп. IV міжнар. наук. конф. (Київ, 20–23 трав. 2014 р.). – К., 2014. – С. 69.

2. Коваленко Ю. Б., Барлака М. А. Алгоритми планування, засновані на квантуванні в системах реального часу. Безпека інформаційних технологій (ITSEC) : тези доп. IV міжнар. наук. конф. (Київ, 17–19 трав. 2016 р.). – К., 2016. – С. 47.

3. Коваленко Ю. Б., Рибалка Л. П. Новітні технології управління корпоративним інформаційним контентом. АВІА-2015 : тези доп. XII міжнар. наук.-техн. конф. (Київ, 28–29 квіт. 2015 р.). – К., 2015. – Т. 1. – С. 121–122.

4. Коваленко Ю. Б., Мішина І. Ю. Проблема підбору варіантів удосконалення проектованої системи фаззінгу з використанням генетичних

алгоритмів. Інтегровані інтелектуальні робототехнічні комплекси (ІІРТК-2015): тези доп. VIII міжнар. наук.-практ. конф. (Київ, 18–19 трав. 2015 р.). – К., 2015. – С. 266–268.

5. Коваленко Ю. Б., Мішина І. Ю. Розробка альтернативної системи захисту інформації на основі існуючого проекту. Політ. Сучасні проблеми науки: тези доп. XV міжнар. наук.-практ. конф. молодих учених і студентів (Київ, 8–9 квіт. 2015 р.). – К., 2015. – С. 112.

6. Kovalenko Y. B. Specifics of the software for aircraft operation. Комплексне забезпечення якості технологічних процесів та систем: тези доп. X міжнар. наук.-практ. конф. (Чернігів, 29–30 квіт. 2020 р.). – Чернігів, 2020. – Т. 2. – С. 156–157.

7. Коваленко Ю. Б., Рибалка Л. П. Аналіз методів зберігання, пошуку та структурування інформації // АВІА-2017: тези доп. XII міжнар. наук.-техн. конф. (Київ, 19–21 квіт. 2017 р.). – К., 2017. – Т. 1. – С. 439–432.

8. Коваленко Ю. Б., Ігнатенко П.Л. Системний аналіз комбінованого аналізатора бінарного коду BitBlaze. Комплексне забезпечення якості технологічних процесів та систем : тези доп. VI міжнар. наук.-практ. конф. (Чернігів, 26–29 квіт. 2016 р.). – Чернігів, 2016. – С. 306.

9. Коваленко Ю. Б. Структурна схема системи Elite Keylogger. Приладобудування: стан і перспективи : тези доп. XV міжнар. наук.-техн. конф. (Київ, 17–19 трав. 2016 р.). – К., 2016. – С. 86–87.

10. Kovalenko Yu. B. Principles of organizing the architecture of perspective onboard digital computing systems in avionics // POLIT. Challenges of science today : theses of reports (Kyiv, 5–9 Apr. 2021). – Kyiv., 2021. – P. 44–46.

11. Kovalenko Yuliia, Kovalchuk Olena. Zero Trust Security: Threats to Information Security in the Context of National Security // International and National Security: Theoretical and Applied Aspects: proceedings of the 9th International Scientific and Practical Conference (Dnipro, 14 Mar. 2025). – Dnipro : Dnipropetrovsk State University of Internal Affairs, 2025. – P. 41–43