

Міністерство освіти і науки України
Національний аерокосмічний університет
«Харківський авіаційний інститут»

Кафедра «Інженерії програмного забезпечення» (№ 603)

ЗАТВЕРДЖУЮ

Гарант освітньої програми



(підпис)

А.Г. Чухрай
(ініціали та прізвище)

« 29 » __ 08 ____ 2025 р.

СИЛАБУС ОBOB'ЯЗKОВОЇ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

**Інженерія програмного забезпечення наносупутників та
безпілотних повітряних літальних апаратів**

(назва навчальної дисципліни)

Галузь знань: F Інформаційні технології

(шифр і найменування галузі знань)

Спеціальність: F2 Інженерія програмного забезпечення

(код та найменування спеціальності)

Освітня програма: Інженерія програмного забезпечення (ОНП)

(найменування освітньої програми)

Рівень вищої освіти: другий (магістерський)

Силабус введено в дію з 01.09.2025 року

Харків – 2025 р.

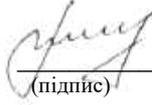
Розробник: Олександр ЛЮБИМОВ
(прізвище та ініціали, посада, науковий ступінь та вчене звання)


(підпис)

Силабус навчальної дисципліни розглянуто на засіданні кафедри інженерії програмного забезпечення (№ 603)

Протокол № 1 від « 29 » серпня 2025 р.

Завідувач кафедри д-р техн.наук., проф.
(науковий ступінь та вчене звання)


(підпис)

Ігор ТУРКІН
(ініціали та прізвище)

Погоджено з представником здобувачів освіти:
Представник студентського самоврядування


(підпис)

Діана ДИКУН
(ініціали та прізвище)

1. Загальна інформація про викладача



ПІБ: Любімов Олександр Вікторович
Посада: асистент кафедри інженерії програмного забезпечення

Науковий ступінь: -

Вчене звання: -

Перелік дисциплін, які викладає:

-
- Архітектура комп'ютерів
 - Програмування на Асемблері
 - Інженерія програмного забезпечення наносупутників та безпілотних повітряних літальних апаратів.
-

Напрями наукових досліджень: інженерія програмного забезпечення, програмне забезпечення бортових обчислювачів супутників та БПЛА, надійність ПЗ авіоніки, енергоефективні обчислення.

Контактна інформація:

E-mail: o.liubimov@khai.edu

Signal: OleksandrL.68

2. Опис навчальної дисципліни

Форма здобуття освіти	<i>Інституційна (очна (денна, вечірня), заочна, дистанційна, мережева)</i>
Семестр	2 семестр
Мова викладання	Українська
Тип дисципліни	<i>Обов'язкова</i>
Обсяг дисципліни: кредити ЄКТС/ кількість годин	<i>денна: 3 кредити ЄКТС / 90 годин (32 аудиторних, з яких: лекції – 16, практичні – 16; СРЗ – 58)</i>
Види навчальної діяльності	<i>Лекції, практичні заняття, самостійна робота</i>
Види контролю	<i>Поточний, модульний та підсумковий (семестровий) контроль (іспит).</i>
Пререквізити	<i>Системи реального часу.</i>
Кореквізити	немає
Постреквізити	<i>Надійність програмно-апаратних комплексів, науково-дослідницька робота магістра (КП).</i>

3. Мета та завдання навчальної дисципліни

Мета - формування у студентів системних знань та практичних навичок інженерії бортового програмного забезпечення наносупутників і безпілотних літальних апаратів з урахуванням вимог реального часу, відмовостійкості та обмежених ресурсів. Опанування принципів проектування компонентних архітектур авіоніки, зокрема на базі відкритих платформ NASA/JPL F' та PX4 Autopilot. Розвиток здатності формалізувати вимоги, здійснювати архітектурний аналіз та впроваджувати механізми виявлення й обробки відмов. Формування здатності визначати та обґрунтовувати процесні вимоги до верифікації та валідації бортового програмного забезпечення на основі положень авіаційних стандартів EUROCAE (зокрема ED-12C та ED-80) та космічних стандартів ECSS, з урахуванням рівня критичності систем і життєвого циклу розробки.

Завдання - Надати здобувачам знання щодо нормативної бази верифікації та валідації бортового ПЗ відповідно до стандартів EUROCAE та ECSS. Сформуванню вміння визначати рівні критичності програмного забезпечення та встановлювати відповідні V&V активності. Забезпечити практичний досвід застосування вимог стандартів до реальних або модельних бортових систем наносупутників і БПЛА. Надати практичні навички розробки сучасного бортового ПЗ наносупутників і БПЛА на базі відкритих фреймворків.

Компетентності, які набуваються:

Інтегральна компетентність:

Здатність продукувати нові ідеї, розв'язувати комплексні проблеми професійної та/або дослідницько-інноваційної діяльності у сфері інженерії програмного забезпечення та з дотичних до неї міждисциплінарних напрямках, застосовувати методологію наукової та педагогічної діяльності, проводити власне наукове дослідження, результати якого мають наукову новизну, теоретичне та практичне значення.

Загальні компетентності (ЗК):

Після закінчення цієї програми здобувач освіти буде здатен:

- ЗК01. Здатність до абстрактного мислення, аналізу та синтезу;
- ЗК02. Здатність застосовувати знання у практичних ситуаціях;
- ЗК03. Здатність проводити дослідження на відповідному рівні;
- ЗК05. Здатність генерувати нові ідеї (креативність);
- ЗК06. Здатність до пошуку, оброблення та аналізу інформації з різних джерел;

Фахові (спеціальні) компетентності:

Після закінчення цієї програми здобувач освіти буде здатен:

СК01. Здатність аналізувати предметні області, формувати, класифікувати вимоги до програмного забезпечення.

СК02. Здатність розробляти і реалізовувати наукові та/або прикладні проєкти у сфері інженерії програмного забезпечення.

СК03. Здатність проєктувати архітектуру програмного забезпечення, моделювати процеси функціонування окремих підсистем і модулів.

СК10. Здатність планувати і виконувати наукові дослідження з інженерії програмного забезпечення.

СК11. Здатність застосовувати і розвивати фундаментальні і міждисциплінарні знання для успішного розв'язання наукових проблем інженерії програмного забезпечення.

Програмні результати навчання:

ПРН01. Знати і застосовувати сучасні професійні стандарти і інші нормативно-правові документи з інженерії програмного забезпечення

ПРН05. Розробляти, аналізувати, обґрунтовувати та систематизувати вимоги до програмного забезпечення.

ПРН06. Розробляти і оцінювати стратегії проєктування програмних засобів; обґрунтовувати, аналізувати і оцінювати варіанти проєктних рішень з точки зору якості кінцевого програмного продукту, ресурсних обмежень та інших факторів.

ПРН07. Аналізувати, оцінювати і застосовувати на системному рівні сучасні програмні та апаратні платформи для розв'язання складних задач інженерії програмного забезпечення.

ПРН09. Обґрунтовано вибирати парадигми і мови програмування для розроблення програмного забезпечення; застосовувати на практиці сучасні засоби розроблення програмного забезпечення.

ПРН11. Забезпечувати якість на всіх стадіях життєвого циклу програмного забезпечення, у тому числі з використанням релевантних моделей та методів оцінювання, а також засобів автоматизованого тестування і верифікації програмного забезпечення.

ПРН16. Планувати, організовувати та здійснювати тестування, верифікацію та валідацію програмного забезпечення.

ПРН21. Вміти застосовувати на практиці теоретичні положення та стандарти з інженерії систем та програмних засобів

4. Зміст навчальної дисципліни

Змістовий модуль 1. Бортове програмне забезпечення наносупутників на базі F'

Тема 1. Введення в архітектуру побудови ПЗ наносупутників. Стандарти.

Анотація: Модуль присвячений вивченню архітектурних принципів побудови бортового програмного забезпечення наносупутників з урахуванням обмежених ресурсів, вимог надійності та відмовостійкості. Розглядаються підходи до системної інженерії космічних апаратів, нормативні вимоги стандартів ECSS до життєвого циклу програмного забезпечення, а також принципи формалізації вимог і трасування. Особлива увага приділяється компонентній архітектурі F' (F Prime) як сучасній платформі розробки flight software, її execution model, механізмам обміну даними та управління. У практичній частині модуль передбачає розробку базових компонентів, інтеграцію підсистем і реалізацію механізмів fault detection та Safe Mode з подальшим аналізом поведінки системи.

Тема лекції 1. Архітектура бортових обчислювальних систем наносупутників.

Тема лекції 2. Стандарти розроблення бортового ПЗ наносупутників.

Тема практичного завдання 1. Аналіз архітектури реального наносупутника та декомпозиція підсистем та вимог до них.

Тема практичного завдання 2. Проєктування вимог до ПЗ та оцінка рівня його критичності.

Самостійна робота здобувачів: опрацювання матеріалу лекцій, підготовка до практичних занять, практична робота з вимогами до бортового ПЗ, декомпозиція підсистем наносупутника на складові та аналіз відповідних ресурсних обмежень, формування питань до викладача.

Тема 2. Розробка ПЗ наносупутників на базі F'

Анотація: Тема присвячена практичному опануванню компонентної архітектури платформи F' (F Prime) та її моделі виконання, зокрема механізмів Ports, Commands, Events, Telemetry, deployment topology та rate groups у детермінованих flight software системах. Розглядаються принципи побудови програмної структури бортового комп'ютера наносупутника та інтеграції підсистем на рівні програмних компонентів. Особлива увага приділяється механізмам fault detection, isolation and recovery (FDIR), health monitoring та реалізації Safe Mode. У практичній частині передбачено встановлення F', створення власних компонентів, побудову тестової топології, реалізацію telemetry pipeline, інтеграцію mock-підсистем (ADCS/EPS) та проведення fault injection з аналізом реакції системи. Тема формує навички інженерного проектування, інтеграції та тестування бортового ПЗ в умовах обмежених ресурсів і підвищених вимог до надійності.

Тема лекції 3. Архітектура та модель виконання F'

Тема лекції 4. Аналіз та керування відмовами та інтеграція підсистем.

Тема практичного завдання 3. Практичний розбір використання F' та побудова типової топології ПЗ.

Тема практичного завдання 4. Побудова телеметрично-командних зв'язків компонентів, використання підсистем обробки відмов.

Самостійна робота здобувачів: опрацювання матеріалу лекцій, поглиблені практичні навички з розробки та проектування ПЗ, формування питань до викладача.

Модульний контроль 1

Форма занять: написання модульної роботи в аудиторії (за рішенням лектора допускається проведення у дистанційній формі) що складається з трьох складових – теоретичних питань, аналітичного та практичного завдань.

Змістовий модуль 2. Бортове програмне забезпечення БПЛА на базі PX4 Autopilot

Тема 3. Архітектура ПЗ БПЛА та авіаційні стандарти

Анотація: Тема присвячена вивченню архітектурних принципів побудови бортового програмного забезпечення безпілотних літальних апаратів та практичному освоєнню платформи PX4 Autopilot. Розглядаються особливості event-driven архітектури, middleware uORB, структура flight stack, механізми керування та відмовостійкості. Значна увага приділяється авіаційним процесним вимогам (EUROCAE ED-12C), класифікації критичності, verification & validation активностям та інтеграції програмних модулів. У практичній частині передбачено роботу з PX4 SITL, реалізацію власних модулів, інжекцію відмов та аналіз поведінки системи.

Тема лекції 5. Архітектура систем БПЛА та flight stack PX4

Тема лекції 6. Авіаційні стандарти та процесні вимоги.

Тема практичного завдання 5. Встановлення та конфігурація PX4, запуск SITL.

Тема практичного завдання 6. Формалізація вимог до ПЗ БПЛА з визначенням критичності за DAL.

Самостійна робота здобувачів: опрацювання матеріалу лекцій, підготовка до практичних занять, робота з авіаційними стандартами EUROCAE та їх порівняння з аналогічними стандартами до ПЗ наносупутників.

Тема 4. Розробка та тестування ПЗ БПЛА на базі PX4

Анотація: Тема присвячена практичному освоєнню процесів розробки, розширення та тестування бортового програмного забезпечення безпілотних літальних апаратів на базі платформи PX4 Autopilot. Розглядаються механізми створення власних програмних модулів, робота з middleware uORB, інтеграція сенсорних даних та аналіз часових характеристик виконання. Особлива увага приділяється реалізації та перевірці failsafe-механізмів, інжекції відмов у середовищі SITL та аналізу логів польоту. У межах теми формуються навички підготовки тест-планів, встановлення трасування між вимогами та тестами та оцінювання впливу змін у програмному коді на безпеку й детермінованість системи.

Тема лекції 7. Розширення функціональності PX4

Тема лекції 8. Failsafe механізми та V&V у PX4.

Тема практичного завдання 7. Реалізація користувачького програмного модуля у екосистемі ПЗ PX4.

Тема практичного завдання 8. Симуляція відмов у SITL (відмова сенсорів, втрата GPS).

Самостійна робота здобувачів: опрацювання матеріалу лекцій, підготовка до практичних занять, проєктування власного модуля наглядю за відмовами бортової авіоніки та ПЗ (схема + опис).

Модульний контроль 2

Форма занять: написання модульної роботи в аудиторії (за рішенням лектора допускається проведення у дистанційній формі) що складається з трьох складових – теоретичних питань, аналітичного та практичного завдань.

5. Індивідуальні завдання

Не передбачено навчальним планом

6. Методи навчання

Лекції з елементами інтерактиву (пояснення з використанням презентацій, прикладів, міні-опитувань). *Практичні заняття* – підготовка та захист звітних документів. Робота в малих групах – колективний аналіз наданих документів. *Використання системи онлайн-тестування.* *Самостійна робота* – індивідуальні завдання, робота з електронними матеріалами та онлайн-курсами. *Консультації* – індивідуальні та групові (очно або онлайн) для підтримки та корекції навчального процесу.

7. Методи контролю

Поточний контроль: опитування на лекціях та практичних заняттях; програмований контроль (тестування, онлайн-тести); оцінювання виконання індивідуальних і групових практичних завдань.

Модульний контроль: складання модульного контролю;

Підсумковий контроль: іспит.

8. Критерії оцінювання та розподіл балів, які отримують здобувачі

Таблиця 8.1 – Розподіл балів, які отримують здобувачі освіти

Складові навчальної роботи	Бали за одне заняття (завдання)	Кількість занять (завдань)	Сумарна кількість балів
Змістовний модуль 1			
Тестування після лекції	0..1	4	0..4

Виконання і захист практичних робіт	0...5	4	8...20
Модульний контроль (тестування)	0...22	1	0...26
Змістовний модуль 2			
Тестування після лекції	0..1	4	0..4
Виконання і захист практичних робіт	0...5	4	0...20
Модульний контроль (тестування)	0...22	1	0...26
Усього за семестр			0...100

Семестровий контроль (іспит) проводиться у разі відмови здобувача від балів поточного тестування й за наявності допуску до іспиту. Під час складання семестрового іспиту здобувач має можливість повторно пройти контроль знань та покращити свою оцінку з теорії. Тест складається з 30 питань та виконується з обмеженням часу – 30 хвилин.

Таблиця 8.2 – Шкали оцінювання: бальна і традиційна

Сума балів	Оцінка за традиційною шкалою	
	Іспит, диференційний залік	Залік
90 – 100	Відмінно	Зараховано
75 – 89	Добре	
60 – 74	Задовільно	
0 – 59	Незадовільно	Не зараховано

Критерії оцінювання роботи здобувача протягом семестру

Задовільно (60-74). Здобувач демонструє базове розуміння архітектури бортового програмного забезпечення наносупутників та БПЛА, принципів побудови компонентних систем на базі F' та загальної структури PX4 Autopilot. Виконує основні практичні завдання, але з окремими неточностями в аргументації або формалізації вимог. Розуміє загальні положення стандартів ECSS та EUROCAE ED-12C, може визначити критичність функцій та описати базові V&V активності. Міні-проект виконано на мінімально достатньому рівні з частковим обґрунтуванням архітектурних рішень.

Добре (75-89). Здобувач упевнено орієнтується в архітектурі F' та PX4, коректно пояснює принципи моделі виконання ПЗ, middleware uORB та механізми обробки помилок та збоїв. Здатний формалізувати вимоги, побудувати матрицю трасування, визначити DAL та сформулювати перелік V&V активностей відповідно до стандартів. Практичні роботи та міні-проект виконані повністю, з логічним

обґрунтуванням прийнятих інженерних рішень. Відповіді на теоретичні питання структуровані та аргументовані.

Відмінно (90-100). Здобувач демонструє системне та глибоке розуміння архітектури бортового ПЗ, здатність виконувати комплексний аналіз компонентної інтеграції, детермінізму та відмовостійкості. Аргументовано застосовує вимоги ECSS та EUROCAE ED-12C для формування процесних і технічних рішень, коректно визначає вплив змін у кодї на безпеку системи. Практичні роботи містять повноцінну архітектурну модель, FDIR-логіку, відстежуваність та елементи V&V аналізу. Відповіді характеризуються чіткою термінологією, глибиною та інженерною обґрунтованістю. Всі контрольні точки здані з оцінкою «відмінно». Досконально знає всі теми та уміти застосовувати їх.

9. Політика навчального курсу

Процедура відпрацювання пропущених занять: здобувач самостійно ознайомлюється з пропущеним матеріалом: лекційними конспектами, навчальними презентаціями, записами занять або додатковими матеріалами, наданими викладачем. Пропущене практичне заняття відпрацьовується шляхом виконання всіх завдань, передбачених для цього. За потреби здобувач може узгодити індивідуальну консультацію з викладачем для роз'яснення складних тем або перевірки виконаних завдань. Після виконання завдань здобувач надає результати викладачу для перевірки у форматі звіту про виконану роботу. Виконане заняття оцінюється за тими ж критеріями, що і основне заняття. Відпрацювання вважається успішним після схвалення викладачем результатів та підтвердження засвоєння матеріалу.

Дотримання вимог академічної доброчесності здобувачами освіти під час вивчення навчальної дисципліни. Під час вивчення навчальної дисципліни здобувачі освіти мають дотримуватися загальноприйнятих морально-етичних норм і правил поведінки, вимог академічної доброчесності, передбачених Положенням про академічну доброчесність Національного аерокосмічного університету «Харківський авіаційний інститут» (<https://khai.edu/assets/files/polozhennya/polozhennya-pro-akademichnu-dobrochesnist.pdf>). Очікується, що розроблені здобувачами програми та звіти до них будуть оригінальними. Відсутність посилань на використані джерела, фабрикування джерел, списування, втручання в роботу інших здобувачів освіти становлять, але не обмежують, приклади можливої академічної недоброчесності. Виявлення ознак академічної недоброчесності в письмовій роботі здобувача освіти є підставою для її незарахування викладачем незалежно від масштабів плагіату чи обману.

Вирішення конфліктів. Порядок і процедури врегулювання конфліктів, пов'язаних із корупційними діями, зіткненням інтересів, різними формами дискримінації, сексуальними домаганнями, міжособистісними стосунками та іншими ситуаціями, що можуть виникнути під час навчання, а також правила

етичної поведінки регламентуються Кодексом етичної поведінки в Національному аерокосмічному університеті «Харківський авіаційний інститут» (<https://khai.edu/ua/university/normativna-baza/ustanovchi-dokumenti/kodeks-etichnoi-povedinki/>).

10. Методичне забезпечення та інформаційні ресурси

Підручники, навчальні посібники, навчально-методичні посібники, конспекти лекцій, методичні рекомендації з проведення лабораторних робіт тощо, які видані в Університеті знаходяться за посиланням:

1. Дистанційний курс дисципліни розроблено у системі дистанційного навчання Mentor, яку впроваджено в Національному аерокосмічному університеті «ХАІ», доступ до курсу за посиланням: <https://mentor.khai.edu/course/view.php?id=9429>

11. Рекомендована література

Базова

1. DO-178C. Software Considerations in Airborne Systems and Equipment Certification. – Washington, DC : RTCA, Inc., 2011. – 156 p. – ISBN 978-1-56018-489-7.
2. ECSS-Q-ST-80C. Space product assurance – Software product assurance. – Noordwijk : European Cooperation for Space Standardization (ECSS), 2009. – 180 p. – ECSS Document No. ECSS-Q-ST-80C.
3. Kopetz H. Real-Time Systems: Design Principles for Distributed Embedded Applications. – 2nd ed. – New York : Springer, 2011. – 396 p. – ISBN 978-1-4419-8236-0. – DOI: <https://doi.org/10.1007/978-1-4419-8237-7>
4. Storey N. Safety-Critical Computer Systems. – Harlow : Addison-Wesley, 1996. – 384 p. – ISBN 978-0-201-42787-8.
5. Sommerville I. Software Engineering. – 10th ed. – Boston : Pearson Education, 2016. – 816 p. – ISBN 978-0-13-394303-0.
6. Burns A., Wellings A. Real-Time Systems and Programming Languages. – 4th ed. – Boston : Addison-Wesley, 2009. – 336 p. – ISBN 978-0-321-41745-9.
7. Leveson N. Engineering a Safer World: Systems Thinking Applied to Safety. – Cambridge, MA : MIT Press, 2011. – 560 p. – ISBN 978-0-262-01662-9.

Допоміжна

1. Knight J. C. Safety Critical Systems: Challenges and Directions. – Proceedings of the 24th International Conference on Software Engineering (ICSE 2002). – Orlando, FL : ACM, 2002. – P. 547–550. – DOI: <https://doi.org/10.1145/581339.581408> – ISBN 1-58113-472-X.

2. Rushby J. Theorem Proving for Verification. – In: D. Bjorner et al. (eds.) Formal Methods for Industrial Applications. – Berlin : Springer, 1999. – P. 15–29. – DOI: https://doi.org/10.1007/3-540-48119-2_2 – ISBN 978-3-540-65898-1.
3. Sha L., Rajkumar R., Lehoczky J. Real-Time Scheduling Theory: A Historical Perspective // Real-Time Systems. – 2004. – Vol. 28, No. 2–3. – P. 101–155. – DOI: <https://doi.org/10.1023/B:TIME.0000045310.43241.d9> – ISSN 0922-6443.
4. Koopman P. Better Embedded System Software. – Pittsburgh : Drumnadrochit Education LLC, 2010. – 386 p. – ISBN 978-0-9837901-0-5.
5. MISRA. MISRA C:2012 Guidelines for the Use of the C Language in Critical Systems. – MISRA Consortium, 2012. – ISBN 978-1-906400-10-1.
6. Laprie J.-C. Dependability: Basic Concepts and Terminology. – Vienna : Springer, 1992. – 342 p. – ISBN 978-3-7091-9124-3. – DOI: <https://doi.org/10.1007/978-3-7091-9124-3>

12. Інформаційні ресурси

1. ESA. CubeSat Design Specification (CDS) [Електронний ресурс]. – Noordwijk: ESA, 2014. – Режим доступу: <https://ecss.nl> (дата звернення: 10.05.2025).
2. NASA. NASA Software Engineering Handbook (NASA-HDBK-2203) [Електронний ресурс]. – Washington, DC : NASA, 2019. – Режим доступу: <https://www.nasa.gov> (дата звернення: 10.05.2025).