

**Святний В.А.**

ДВНЗ «Донецький національний технічний університет»

**Мірошкін О.М.**

Університет Ульма (Німеччина)

**Мінаков М.А.**

ДВНЗ «Донецький національний технічний університет»

**Маргієв Г.Е.**

ДВНЗ «Донецький національний технічний університет»

## ВИКОРИСТАННЯ РОЗПОДІЛЕНОГО ПАРАЛЕЛЬНОГО МОДЕЛЮЮЧОГО СЕРЕДОВИЩА ПІД ЧАС РОЗРОБЛЕННЯ СИСТЕМ РЕАЛЬНОГО ЧАСУ

Для вирішення певних прикладних задач комп'ютерні системи повинні приймати рішення в обмежені інтервали часу. Прикладом таких систем є системи реального часу, або вбудовані системи. Розробка вбудованих систем ускладнюється постійним зростанням кількості та складності вимог і системних функцій та ускладненням систем в цілому. У цій статті розглянуто підходи до розробки вбудованого програмного забезпечення як найважливішої частини вбудованої системи, зокрема використання V-моделі як методики розробки вбудованого програмного забезпечення. Описано її переваги та недоліки. Розглянуто модельну підтримку процесу розробки вбудованих систем та використання документо-базованого підходу, їх переваги та недоліки. Наведено модельно-базований підхід до розробки. Розглянуто особливості використання SPES-фреймворка як комплексної методики розробки вбудованих систем. Прیدілено особливу увагу опису кожної з чотирьох позицій, які визначаються SPES-фреймворком. Розглянуто використання підходу Design Space Exploration, що може бути застосований на останніх етапах моделювання вбудованої системи. Описано переваги використання модельно-базованого підходу та SPES-фреймворку. Наведено основний засіб для проведення верифікації та валідації моделі вбудованої системи. Розглянуто парадигми валідації та верифікації, такі як Model/Software/Hardware -in-the-loop simulation. Описано проблему використання послідовних симуляторів. Запропоновано використання високопродуктивного розподіленого паралельного моделюючого середовища, що може бути використане під час розробки, валідації та верифікації моделі вбудованої системи. Визначено подальшу роботу з використання розподіленого паралельного моделюючого середовища в процесах розробки та тестування вбудованих систем і вбудованого програмного забезпечення.

**Ключові слова:** вбудована система, модельно-базована розробка, SPES-фреймворк, V-модель, Model-in-the-loop, розподілене паралельне моделююче середовище.

**Постановка проблеми.** Більшість вбудованих систем належить до систем реального часу – класу комп'ютерних систем, які мають вирішувати спеціалізовані задачі за певні часові інтервали [1, с. 3]. При цьому характеристики часових інтервалів визначаються галуззю застосування вбудованої системи та можуть мати істотні відмінності для різних випадків застосування.

Постійні ускладнення систем, збільшення кількості виконуваних функцій, посилення вимог безпеки, скорочення витрат на розробку та доведення розробки до ринку призводять до того, що розробка сучасних вбудованих систем стає все

більш складною задачею. Нагальною проблемою є запровадження сучасних методик розробки надійних систем реального часу з одночасним зниженням кількості використаних для проектування часових та інших ресурсів.

Вбудовані системи складаються з програмного та апаратного забезпечення. Обчислювач найчастіше реалізується за допомогою мікроконтролерів, що приєднуються до різноманітної апаратури комунікаційних приладів, сенсорів, виконавчих механізмів (actuators) та панелі оператора чи користувача. Програмне забезпечення, що виконується на подібних мікроконтролерах, називається

вбудованим програмним забезпеченням [2] і є найважливішою частиною всієї системи, тому що реалізує бізнес-логіку, тобто функціонал системи. Отже, якість вбудованого програмного забезпечення визначає і якість всієї вбудованої системи взагалі.

**Аналіз останніх досліджень і публікацій.** Життєвий цикл програмного забезпечення (*Systems development life cycle*) – це проміжок часу з моменту прийняття рішення про необхідність створення програмного забезпечення до моменту проведення приймальних тестів (Acceptance testing) та пусконаладжувальних робіт [3]. Серед моделей, що використовуються під час розробки вбудованого програмного забезпечення, можна виділити такі: водоспадна модель, V-модель, спіральна модель, еволюційна модель, ітеративна модель [4]. Моделлю, що де-факто є стандартом під час розробки вбудованих систем, є V-модель ХТ 2005, яка також рекомендована для використання під час роботи над військовими та аерокосмічними проектами [5]. Розглянемо її основні положення, переваги та недоліки.

V-модель визначає результати, які будуть досягнуті в проєкті, і описує конкретні процедури. Крім того, вона визначає обов'язки окремих учасників проєкту. На рисунку 1 схематично зображено процес розробки.

Переваги V-моделі:

- мінімізація проєктних ризиків;
- обмеження загальної вартості протягом всього проєкту і життєвого циклу системи;
- покращення комунікації між усіма зацікавленими сторонами;

- створення прототипів у процесі розробки.

Недоліки V-моделі:

- перевірка, яка спрямована на демонстрацію того, що специфікація вимог чи продукт відповідають первинному наміру, проводиться після імплементації;
- надмірна різноманітність артефактів (наприклад, документів, ролей) і бюрократія;
- без відповідної інструментальної підтримки V-модель практично неможливо використовувати.

Серед недоліків V-моделі основним є те, що тести, верифікація та валідація проводяться після імплементації програмного забезпечення, що ускладнює виправлення помилок. Найбільш негативним сценарієм є наявність помилки в специфікації вимог. Тоді виправлення цієї помилки коштує дуже дорого, особливо на пізніх стадіях розробки.

**Постановка завдання.** Основною метою публікації є опис методів імплементації розподіленого паралельного моделюючого середовища в наявну методологію розробки вбудованих систем.

**Виклад основного матеріалу дослідження.** Можливим рішенням мінімізації помилок є використання модельно-базованого підходу для опису вимог та системи, а також використання уніфікованої мови для опису системи (наприклад, UML та SysML). Методика використання подібного підходу описана в фреймворку SPES Modeling Framework (Software Platform for Embedded Systems) [6].

SPES Modeling Framework був розроблений групою німецьких фахівців для підтримки безперервного моделювання вбудованих систем для

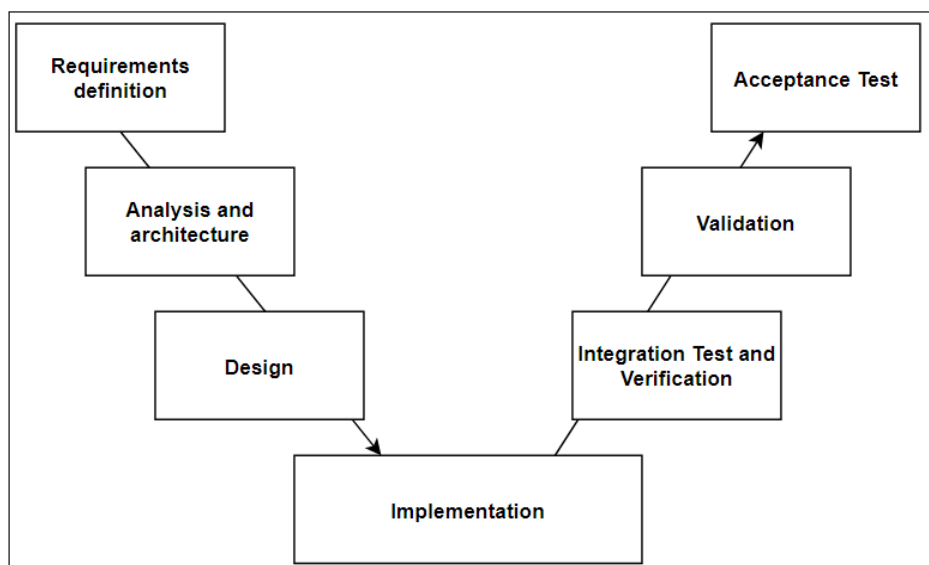


Рис. 1. Схематичне зображення V-моделі розробки вбудованого ПЗ

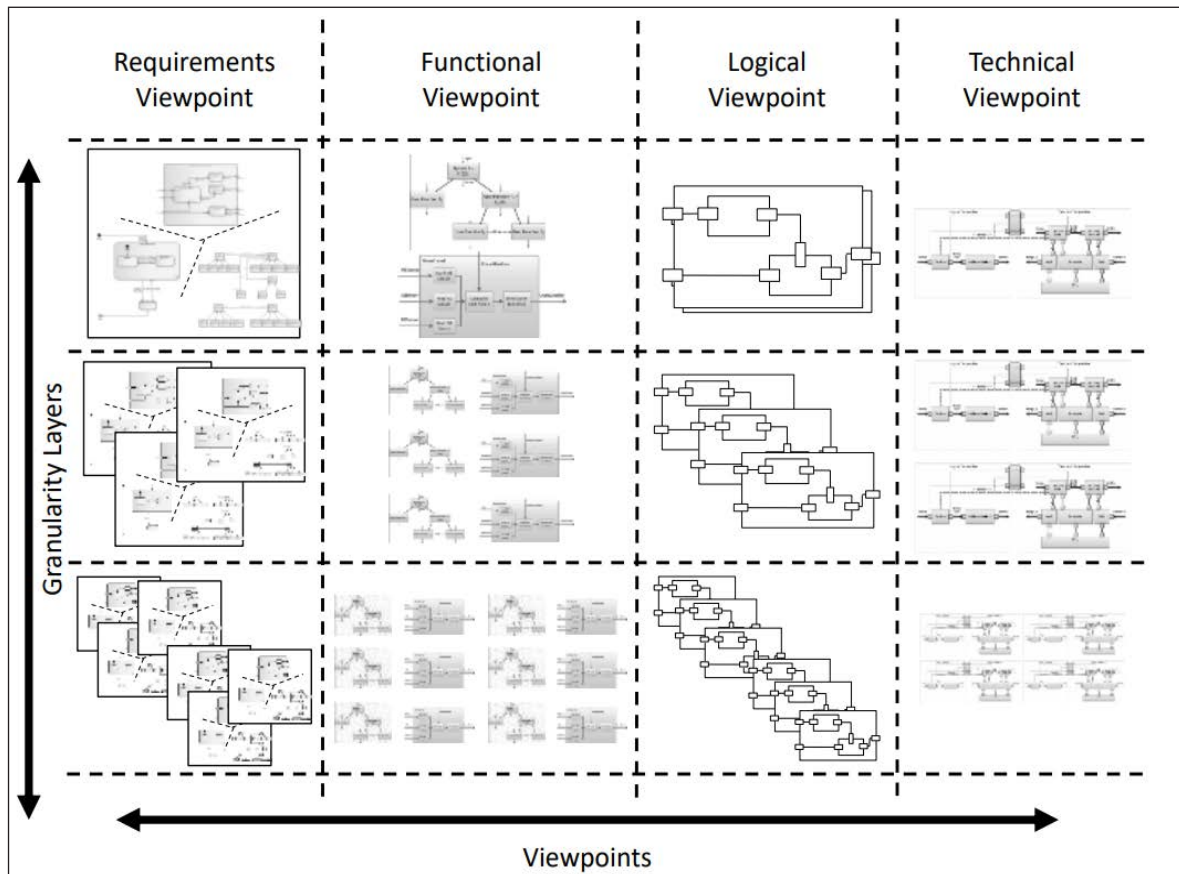


Рис. 2. Позиції з приводу розроблюваної системи за SPES Modeling Framework

різних галузей застосування [7]. Артефактно орієнтований характер дозволяє створювати моделі залежно від індивідуальних потреб без визначення процесу їх створення, що робить цей підхід більш адаптивним. Для цього фреймворк визначає чотири позиції:

- позицію вимог (Requirements Viewpoint);
- функціональну позицію (Functional Viewpoint);
- логічну позицію (Logical Viewpoint);
- технічну позицію (Technical Viewpoint).

Кожна позиція описує систему зі свого боку та має різні рівні деталізації (Granularity Layers) [8, с. 23–25].

Позиція вимог (Requirements Viewpoint) – це набір моделей, що описують фундаментальну поведінку майбутньої системи (System under Development) (далі – SUD), її взаємодію з середовищем, де буде функціонувати система (контекст). Це можуть бути інші вбудовані системи (наприклад, якщо SUD є частиною іншої системи чи так званої екосистеми вбудованих систем (Internet of Things)), кіберпростір, користувач, технічний чи фізичний процес. Моделі цієї позиції вимог описують цілі (goals), межі, а також сценарії

SUD, що є основою для моделей з інших позицій [9, с. 30–56].

Функціональна позиція описує функціональність SUD, при цьому розробляються моделі для опису окремих функцій та зв'язків між ними.

Логічна позиція зосереджена на декомпозиції SUD на логічні компоненти, які будуть функціонувати в межах одного електронного блоку керування. На цьому етапі приймаються найважливіші архітектурні рішення [9, с. 54–80].

Технічна позиція включає апаратні особливості SUD. Ця позиція фокусується на розміщенні логічних компонентів, визначених у логічній позиції, на апаратних компонентах (Deployment). На цьому етапі доступне використання підходу Design Space Exploration (DSE). Можна спрогнозувати, як сильно буде завантажена система, чи можливе доповнення новими функціями після релізу SUD шляхом оновлення програмного забезпечення, споживання ресурсів тощо. SPES Modeling Framework не вимагає використання шляху від позиції вимог до технічної послідовно, а пропонує розробникам обирати свій власний шлях через позиції та рівні деталізації залежно від особливостей системи чи компонентів

системи, контексту, вимог, компанії. Приклади наведені на рисунку 3.

Перевагами використання модельно-базованого підходу та SPES-фреймворку є побудова комплексної моделі вбудованої системи та застосування засобів для валідації та верифікації майбутньої системи вже на ранніх етапах розробки. Одним з таких засобів може бути симуляція.

Під час моделювання і симуляції вбудованої системи можна перевірити те, чи вірно розроблені вимоги до системи, чи вірно працюють алгоритми системи, а головне – чи це та система, яка потрібна замовнику. Етапами валідації та верифікації є проведення тестів за допомогою емуляторів контексту [10].

Розглянемо приклад, коли вбудована система повинна використовуватися для взаємодії з механічною чи фізичною системою за допомогою сенсору та виконавчого механізму (Рисунок 4). Для тестування такої системи необхідна модель механічної системи (імітаційна модель), яка повинна

емулювати її поведінку та інтерфейс. Таке тестування – Model-in-the-loop simulation – є найпростішим підходом для верифікації вбудованих систем, що не потребує додаткових ресурсів.

Така модель механічної системи може бути реалізована за допомогою комп'ютерної системи і під'єднаної напряду до моделі вбудованої системи (Рисунок 5). Такий підхід дозволить на ранніх етапах проводити тестування на рівні або за межами діапазону штатних параметрів, а також перевірку системи у разі збою. Пізніше імітаційну модель можна використовувати під час програмно-апаратного моделювання (Hardware-in-the-loop), де замість моделі використовується прототип вбудованої системи.

Для проведення Model-in-the-loop simulation необхідне створення високоточних імітаційних моделей, підвищення якості яких призводить до різкого росту навантаження на комп'ютерну систему. Отже, самого використання послідовних симуляторів може бути недостатньо для проведення подібних тестів. Альтернативою може стати використання паралельних та розподілених сис-

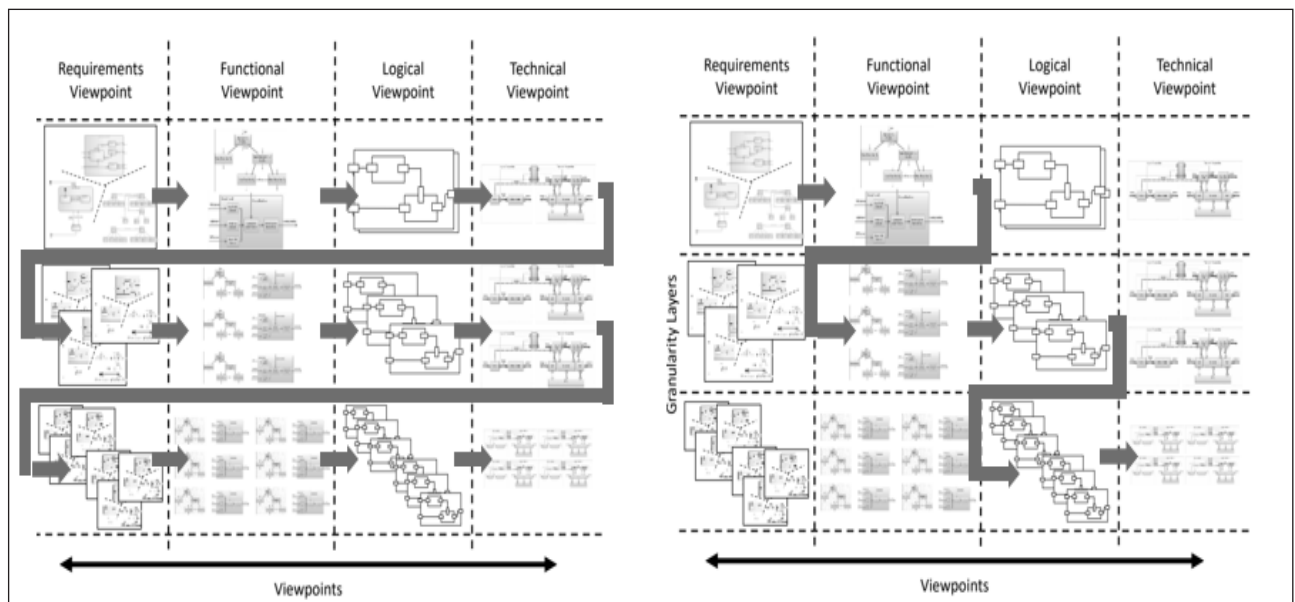


Рис. 3. Шляхи проходження по SPES-матриці

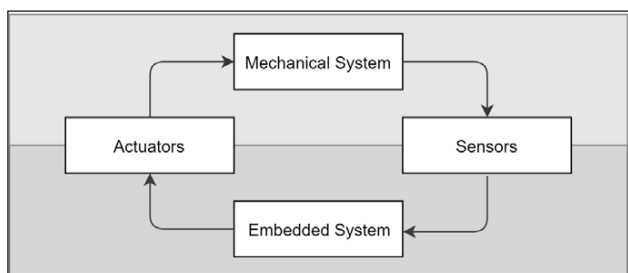


Рис. 4. Схема взаємодії між вбудованою та механічною системами

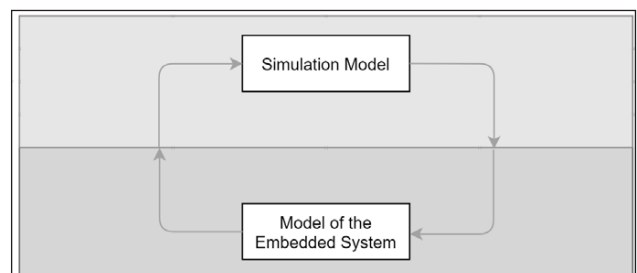


Рис. 5. Схема тестування “Model-in-the-loop simulation”

тем симуляції, таких як розподілене паралельне моделююче середовище (РПМС) [11, с. 83–90].

Використання високопродуктивної РПМС в Model/Software/Hardware-in-the-loop simulation під час проведення тестів має усунути можливі помилки, пов'язані з недостатньою продуктивністю послідовних обчислювальних систем для забезпечення взаємодії в режимі реального часу.

**Висновки.** Подальша робота з використання РПМС у процесах розробки та тестування вбудованих систем спрямована на вирішення таких питань: яку мову для опису поведінки імітаційної моделі й інтерфейсів необхідно використовувати, як адаптувати емуляційну модель для використання у паралельному середовищі, як налагодити взаємодію з моделлю вбудованої системи.

#### Список літератури:

1. П.Б. Вовк, А.П. Усійчук. Проблеми проектування вбудованих систем. *Комп'ютерно-інтегровані технології: освіта, наука, виробництво* : науковий журнал. Випуск № 11. Луцьк : Видавництво ЛНТУ, 2013. С. 82-87.
2. Michael Barr. Embedded Systems Glossary. Netrino Technical Library URL: <https://barrgroup.com/Embedded-Systems/Glossary> (дата звернення: 27.09.2019).
3. IEEE 610.12-1990 – IEEE Standard Glossary of Software Engineering Terminology (ДСТУ 2844-94 Програмні засоби ЕОМ. Забезпечення якості. Терміни та визначення).
4. V-Modell XT, FAQ. IT-Beauftragter der Bundesregierung URL: [https://www.cio.bund.de/Web/DE/Architekturen-und-Standards/V-Modell-XT/Haeufig-gestellte-Fragen/haeufig\\_gestellte\\_fragen\\_node.html](https://www.cio.bund.de/Web/DE/Architekturen-und-Standards/V-Modell-XT/Haeufig-gestellte-Fragen/haeufig_gestellte_fragen_node.html) (дата звернення: 27.09.2019).
5. Security, safety & human factors. IABG Safety URL: <https://www.iabg.de/en/services/schooling-training/security-safety-human-factors> (дата звернення: 27.09.2019).
6. M. Broy. Mission und Vision von SPES 2020. White Paper. 2010.
7. Wolfgang Böhm, Marian Daun, Vasileios Koutsoumpas, Andreas Vogelsang, Thorsten Weyer. Advanced Model-Based Engineering of Embedded Systems: Extensions of the SPES 2020 Methodology, 2016. Print ISBN: 978-3-319-48002-2, Electronic ISBN: 978-3-319-48003-9.
8. Weber R., Reinkemeier P., Henkler S., Stierand I.: Technical Viewpoint. In: K. Pohl, H. Hönniger, R. Achatz, M. Broy: Model-Based Engineering of Embedded Systems: The SPES 2020 Methodology. Springer, Heidelberg/New York, 2012. DOI: 10.1007/978-3-642-34614-9.
9. Andreas Jedlitschka, Ulrich Löwen (2012). Overview of the SPES Evaluation Strategy. In K.Pohl et al. (Eds.) Model-Based Engineering of Embedded Systems: The SPES 2020 Methodology, Springer Berlin Heidelberg, pp. 131–135.
10. Forsberg, K., Mooz, H. System Engineering for Faster, Cheaper, Better. San Jose, 1998.
11. Святний В.А. Паралельне моделювання складних динамічних систем. *Моделювання-2006* : Міжнародна конференція. Київ, 2006. С. 83–90.

#### **Svyatnyy V.A., Miroshkin O.M., Minakov M.A., Marhiiev H.E. DISTRIBUTED PARALLEL SIMULATION ENVIRONMENT USAGE FOR EMBEDDED SYSTEMS DEVELOPMENT**

*To solve some engineering tasks, computer systems should make decisions within the limited time intervals. Such systems are called real-time systems or embedded systems. The development of embedded systems becomes more and more complicated because of increasing of requirements complexity and system functions number, furthermore the complexity of whole system. In this paper we discuss embedded software development approaches, as the most important part of the embedded system, in particular the usage of V-model as a complex embedded software development approach. Its advantages and disadvantages are described. We considered a model support of embedded systems development process and usage of the document-based approach, their advantages and disadvantages. The model-based approach of development is also described. The peculiarities of the SPES-framework usage as a complex method of embedded systems development are considered. Special attention is paid to the description of four viewpoints defined by the SPES-framework. The usage of the Design Space Exploration approach, which can be applied at the final stages of embedded system modeling, is considered. The advantages of the model-based approach usage and in particular SPES-framework are described. The main method for verification and validation of the embedded system model is given. Paradigms of validation and verification such as Model/Software/Hardware-in-the-loop simulation are considered. Problems of sequential simulators application are described. It is proposed to use a high-performance distributed parallel simulation environment (DPSE) for development, validation and verification of the embedded system model. Further work on the use of distributed parallel simulation environment in the development and testing of embedded systems and embedded software is defined.*

**Key words:** embedded systems, model-based development, SPES-framework, V-Model, Model-in-the-loop, distributed parallel simulation environment.