

АВТОМАТИЗАЦІЯ І КОМП'ЮТЕРНО-ІНТЕГРОВАНІ ТЕХНОЛОГІЇ НЕРУЙНІВНОГО КОНТРОЛЮ

УДК 004.451.9

DOI: 10.31471/1993-9981-2022-1(48)-66-77

ВИБІР ОПЕРАЦІЙНИХ СИСТЕМ РЕАЛЬНОГО ЧАСУ ПРИ РОЗРОБЦІ ПРИСТРОЇВ ДЛЯ ТЕХНІЧНОЇ ДІАГНОСТИКИ

**В.А. Ровінський¹, О.В. Євчук², Ю.В. Ровінський¹*

¹Прикарпатський національний університет ім. В. Стефаника, м. Івано-Франківськ, вул. Шевченка, 54, <mailto:v.rovinsky@comp-sc.if.ua>

²Івано-Франківський національний технічний університет нафти і газу, м. Івано-Франківськ, вул. Карпатська, 15, itts@nung.edu.ua

Проведено аналіз характеристик найбільш розповсюджених операційних систем реального часу, придатних для створення систем і пристроїв технічної діагностики. Для аналізу вибрані найбільш популярні серед них. Визначено, що визначення типу операційної системи повинне здійснюватися на основі таких параметрів систем технічної діагностики, як: 1) необхідна швидкодія обробки вхідних даних і пов'язана з нею обчислювальна потужність мікропроцесора системи діагностування; 2) мобільність і прямо пов'язана із споживаною електричною потужністю потреба в наявності чи відсутності автономного живлення пристрою, що входить до складу системи технічного діагностування; 3) ціна пристрою, від якої залежить вибір елементної бази для побудови обчислювачів реального часу дії – дешеві мікропроцесорні системи будуть мати невелику швидкодію, невеликий об'єм пам'яті, і тому при розробці системи технічної діагностики не буде можливим використати довільну операційну систему; 4) галузь застосування, яка накладає додаткові обмеження на надійність функціонування операційної системи (авіація, атомна енергетика, застосування військового спрямування) і вимагає наявності сертифікації безпеки, які визначаються спеціальними стандартами; 5) потреба використання типових давачів та вимірювальних перетворювачів визначає вибір операційної системи з будованою підтримкою таких давачів. Крім того, до важливих критеріїв вибору операційних систем реального часу належить частотна область вхідних інформаційних сигналів, які потребують обробки. Визначено, що у разі, якщо необхідно обробляти високочастотні сигнали, притаманні ультразвуковій та радіочастотній діагностиці, або радіолокації, слід вибирати операційні системи, які підтримують програмовані логічні матриці або цифрові сигнальні процесори. Наявність відкритого коду операційної системи зумовлює можливість його аналізу або модифікації під вузькоспеціалізовані потреби при розробці систем технічної діагностики.

Ключові слова: операційна система, операційна система реального часу, ОСРЧ, мікроконтролер, мікропроцесор, технічна діагностика.

There was an analysis of characteristics of the most wide-spread real-time operating systems, which are suitable for creating technical diagnostics systems and devices. The most popular among them were selected for the analysis. It was determined that the type definition of the operating system should be based on such parameters as: 1) the required input data processing speed and the associated computing power of the system diagnostic microprocessor; 2) mobility and the need for the presence or absence of autonomous power supply of the device that is part of the technical diagnostic system and is directly related to the consumed electric power; 3) the price of device, which determines the element base selection for building real-time computers - cheap microprocessor systems will have slow speed, a small amount of memory and therefore it's not possible to use any operating system during a technical diagnostic system developing; 4) scope, which imposes additional restrictions on the reliability of the operating system functionality (aviation, nuclear power, military use) and requires safety certification, determined by special standards; 5) the need to use typical sensors and measuring transducers determines the pick of an operating system with built-in support for such sensors. In addition, one of the important criteria for choosing real-time operating systems is the frequency range of input data signals that need to be processed. It was determined that if it is necessary to process high-frequency signals that inherent in ultrasound and radio frequency diagnostics or radar, chosen

operating systems should be supported by programmable logic arrays or digital signal processors. The open-source code of the operating system can be analyzed or modified for highly specialized needs in the development of technical diagnostic systems.

Key words: operating system, real-time operating system, RTOS, microcontroller, microprocessor, technical diagnostics.

Вступ. Операційні системи реального часу (ОСРЧ, англ: Real Time Operating System - RTOS) - це тип операційних систем, основне призначення яких - надання необхідного та достатнього набору функцій для проектування, розробки та функціонування систем, які мають здатність забезпечити необхідний рівень сервісу у певний проміжок часу на конкретному апаратному устаткуванні. Серед основних галузей застосування ОСРЧ – засоби технічного діагностування, літальні апарати та системи оборони, системи котирування цін, мережеві та мультимедійні системи, антиблокувальні гальмівні системи, системи управління повітряним рухом, програми для онлайн-комунікацій (аудіо та відео). Ідеальна ОСРЧ має передбачувану поведінку при всіх сценаріях навантаження, включаючи одночасні переривання та виконання потоків.

До переваг операційних систем реального часу можна віднести, те, що ОСРЧ управляється подіями без затримки часу обробки. ОС реального часу пропонують розробку API на основі окремих задач. У більшості випадків, це допомагає розробникам або тестувальникам працювати незалежно над своїми частинами проекту, що зменшує взаємозалежності між модулями за рахунок абстрагування часових залежностей і проектування на основі задач. Можливості ОС для планування на основі системи пріоритетів дозволяють ефективно розподіляти загальний час виконання на час аналітичної обробки та час критичної обробки.

Особливостям виділення пам'яті в ОСРЧ приділяється більше уваги, ніж в операційних системах загального призначення. Перша особливість - швидкість виділення пам'яті. Стандартна схема виділення пам'яті передбачає сканування списку значної довжини для визначення вільної області пам'яті заданого розміру, а це неприйнятно, оскільки в ОСРЧ виділення пам'яті відбувається за фіксований час. Друга особливість - пам'ять може стати фрагментованою в разі розділення вільних її частин вже запущеними процесами. Це може призвести до зупинки програми через її

неможливість задіяти новий фрагмент пам'яті. Алгоритм виділення пам'яті, поступово збільшуючи фрагментарність пам'яті, може успішно працювати на настільних системах, якщо вони перезавантажуються не рідше одного разу протягом місяця, але є неприйнятним для вбудованих систем, які можуть працювати роками без перезавантаження.

На теперішній час існує близько 205 операційних систем, які можна віднести до систем реального часу дії [1] і далеко не всі з них можна використати при проектуванні систем технічної діагностики. Тому успішний вибір ОСРЧ для проектування діагностичної системи потребує всебічного аналізу переваг і недоліків кожної із екземплярів ОСРЧ.

Метою дослідження є виявлення особливостей операційних систем реального часу для розробки технічних засобів діагностування на основі їх порівняльного аналізу.

Практична значущість результатів дослідження полягає в тому, що результати досліджень можуть бути використані при створенні мобільних діагностичних систем широкого спектру застосування.

Матеріали та методи дослідження. Дослідження виконано з використанням даних офіційних виробників програмного забезпечення.

Аналіз літературних джерел. Не дивлячись на велику кількість існуючих операційних систем реального часу, найбільш популярними є наступні [2]:

- Deos (DDC-I)
- embOS (SEGGER)
- FreeRTOS (Amazon)
- Integrity (Green Hills Software)
- Keil RTX (ARM)
- LynxOS (Lynx Software Technologies)
- MQX (Philips NXP / Freescale)
- Nucleus (Mentor Graphics)
- Neutrino (BlackBerry)
- PikeOS (Sysgo)
- ThreadX (Microsoft Express Logic)
- µC/OS (Micrium)

- VxWorks (Wind River)
- Zephyr (Linux Foundation)

Тому є причина розглянути ці операційні системи і виявити їх особливості при використанні в системах технічної діагностики.

Deos™, операційна система реального часу з критичною безпекою DDC-I, яка була перевірена на відповідність вказівкам DO-178C/ED-12C Design Assurance Level A (DAL A) для додатків авіоники, підтримує ARINC 653 APEX, Rate Monotonic Scheduling (RMS). Deos є першою ОСРЧ, яка отримала сертифікат відповідності OSS для технічного стандарту FACE версії 3.1 як базового, так і розширеного профілів безпеки. Розширений профіль безпеки, який додає підтримку протоколу TCP/IP, підтримку багатьох процесів і розширені можливості POSIX (80 додаткових функцій), є надмножиною функціональних можливостей, які вимагаються базовим профілем.

Deos зарекомендувала себе як критично важлива для безпеки ОСРЧ з моменту її першої перевірки та аудиту і була сертифікована і вже «літає» в 10 тисячах літаків. З моменту первинної перевірки вона постійно розвивалась протягом останніх двох десятиліть в наступних базових лініях з новими процесорами та функціями, і врешті була успішно перевірена різними державними органами з сертифікації (FAA, ENAC, JAA, EASA, CAAC та інші).

Технічні особливості:

- Засоби розробки можуть запускатись на базі: Linux і Windows XP, 7, 8 і 10
- Підтримувані процесори: Одно- та багатоядерні процесори x86, PowerPC, ARM і MIPS

FreeRTOS — багатозадачна операційна система реального часу для вбудованих систем. Портована на 35 мікропроцесорних архітектур. Поширюється під ліцензією MIT з 2017 року. FreeRTOS призначена для роботи на масових мікроконтролерах, які можуть мати низьку швидкодію, малий об'єм оперативної пам'яті, відсутність пристроїв управління пам'яттю та реалізованих на апаратному рівні механізмів підтримки багатозадачності, в тому числі швидкого переключення контексту[5].

Диспетчер системи дуже простий і компактний (займає, в залежності від платформи та налаштування ядра, 4-9 кілобайт),

- Існуючі компілятори розробки C, C++ (підмножина), і Ada95 [3]

embOS — це пріоритетно керована ОСРЧ, розроблена як основа для розробки вбудованих додатків і постійно розвивається з 1992 року. Вона доступна для всіх популярних ядер, компіляторів і інструментів розробки і була розгорнута на кількох мільярдах пристроїв у різних областях застосування. Різні механізми кваліфікують embOS як якісний програмний продукт найвищої продуктивності та надійності, завойовуючи своє місце як провідна в галузі операційна система реального часу.

Основні особливості ОС:

- Потужний і простий у використанні API
- Висока продуктивність з низьким використанням пам'яті
- Доступні плагіни на рівні ядра
- Нульова затримка переривань
- Сумісність з MISRA-C:2012
- Функціональна безпека сертифікована відповідно до IEC 61508 SIL 3 і IEC 62304 Class C [4]

Технічні особливості:

- Засоби розробки можуть запускатись на базі: MacOS 10.5+(Intel and Arm M1), Linux і Windows Vista, 7, 8 і 10+ (x86, x64 and Arm),
- Підтримувані процесори: ARM7/9/11, ARM Cortex-A/R/M, AVR, AVR32, C16x, CR16C, ColdFire, H8, HCS12, M16C, M32C, MSP430, NIOS2, PIC18/24/32, R32C, R8C, RISC-V, RL78, RH850, RX100/200/600/700, RZ, SH2A, STM8, ST7, V850, 78K0, 8051
- Існуючі компілятори розробки: C, C++ однак підтримує пріоритетні процеси, витісняючи та оперативну багатозадачність, семафори та черги.

Версія Amazon FreeRTOS відрізняється функціями додаткової бібліотеки, яка забезпечує організацію безпечних з'єднань на базі мікроконтролерів із хмарними сервісами AWS IoT Core або потужнішими пристроями, що працюють під керуванням AWS Greengrass. У Amazon FreeRTOS забезпечується підтримка TLS 1.2, PKCS#11, шифрування даних, перевірки цілісності прошивки за допомогою цифрового підпису. Пропонуються засоби для оновлення прошивки в режимі OTA

(безпроводні). Мережевий стек включає підтримку MQTT, TCP/IP і Wi-Fi. Для вибору та установки програмних компонентів прошивки пропонується керуючий інтерфейс Amazon FreeRTOS Console. Специфічні компоненти для Amazon FreeRTOS, як і ядро FreeRTOS, поширюються під ліцензією MIT.[5]

Технічні особливості:

– Засоби розробки можуть запускатись на базі: GNU/Linux, macOS, Microsoft Windows и Solaris

– Підтримувані процесори: ARM (ARM7, ARM9, Cortex-M0, Cortex-M3, Cortex-M4, Cortex-A), Atmel AVR, AVR32, HCS12, MicroBlaze, Cortus (APS1, APS3, APS3R, APS5, FPF3, FPS6, FPS8), MSP430, PIC, Renesas H8/S, SuperH, RX, x86, 8052, Coldfire, V850, 78K0R, Fujitsu MB91460 series, Fujitsu MB96340 series, Nios II, Cortex-R4, TMS570, RM4x

– Існуючі компілятори розробки: C, C++, micro-python.

Integrity – ОСРЧ, орієнтована на вбудовані авіаційні системи (авіоніку). Integrity розроблена каліфорнійською компанією Green Hills Software. Сертифікована на відповідність POSIX. Орієнтована на однопроцесорні вбудовані системи, в центральному процесорі яких є блок управління пам'яттю. Система базується на власному розробленому в компанії мікроядрі μ -velocity, орієнтованому на пристрої із значними ресурсними обмеженнями. Середовища засобів розробки — інтегроване середовище для розробників вбудованої системи GHS Multi, серія компіляторів для мов Ада, C, C++, які підтримують засоби відладки і крос-компіляцію на різних апаратних платформах,.

Основна особливість системи — відмовостійкість (якщо виникне відмова в якій-небудь програмі, запусненій в цій операційній системі, система в цілому буде продовжувати працювати в штатному режимі, а перезапуск «впавших» додатків буде намагатись провести з наданням йому тих областей пам'яті даних, які були виділені додатком до його «падіння»).

Варіант системи Integrity-178В — операційна система реального часу, «підвид» Integrity, сертифікована за рівнем А стандарту DO-178В. Повністю підтримує вимоги ARINC 653-1-Application/Executive (APEX) API.

Використовується в американських військових літаках (наприклад, F-16, F-22, F-35) і гелікоптерах, а також у цивільних літаках Airbus A380, і Boeing 787.[6]

Технічні особливості:

– Засоби розробки можуть запускатись на базі: GNU/Linux, macOS, Microsoft Windows и Solaris

– Підтримувані процесори і платформи: Altera Cyclone, AMD x86, ARM Ltd, BAE RAD750, Broadcom, Fujitsu ARM, IBM 970, Intel Architecture (various), Intel SoC FPGA (Altera), Marvell (Cavium), Microchip PolarFire SoC, NVIDIA, NXP i.MX,NXP Layerscape, NXP MPC5xxx, NXP QorIQ, NXP S32, Qualcomm Snapdragon, Renesas R-Car, ST Telemaco, Texas Instruments Jacinto, Texas Instruments Sitara, Xilinx Zynq-7000 Soc, Xilinx UltraScale+ MPSoC

– Існуючі компілятори розробки: Ада, C, C++.

Keil RTX — це детермінована операційна система реального часу з відкритим вихідним кодом, що реалізує API CMSIS-RTOS v2, загальний інтерфейс RTOS для пристроїв на основі процесора Cortex-M. Сучасні програми для мікроконтролера часто повинні обслуговувати кілька дій одночасно, тому RTX дає можливість ефективно керувати перемиканням між видами діяльності. Кожна діяльність отримує окремий потік, який виконує конкретне завдання і це суттєво спрощує структуру користувацької програми. Потіки мають пріоритети, що дозволяє швидше виконувати критично важливі за часом частини користувацької програми. Детальніше про ОС:

– ОСРЧ з відкритим вихідним кодом. Keil RTX5 (включаючи вихідний код) постачається разом із Arm Keil MDK, повністю налаштовується і не має обмежень. Вона поставляється під ліцензією Apache 2.0 і на платформі GitHub.

– Проста у використанні - Зменшена потреба в навчанні, швидша розробка продукту. Keil RTX5 добре підтримується в середовищі μ Vision IDE/Debugger, яке містить інструменти, що підтримують RTX5, і дозволяють швидко відлагоджувати програми RTX5.

– Безпека та захист - Keil RTX5 сертифікована PSA і, природно, інтегрується з API архітектури безпеки платформи. Існує сертифікований у безпеці варіант Keil RTX5,

доступний як частина Arm Run-Time System для додатків функціональної безпеки (т.зв.:FuSa RTS).

– Гнучка мультизадачність – можна вибрати найкращий спосіб реалізації мультизадачності для прикладної програми. Keil RTX5 пропонує різні варіанти планування ядра, що дозволяє використовувати найбільш підходящий для вирішуваної задачі. Pre-emptive – кожен потік має інший пріоритет і працюватиме до тих пір, поки не буде готовий до запуску потік вищого пріоритету. Це зазвичай використовується в інтерактивних системах, де пристрій може перебувати в режимі очікування або у фоновому режимі, доки йому не буде надіслано вхідне повідомлення. Round-Robin – кожен потік буде працювати протягом фіксованого періоду часу роботи ЦП (кванту часу). Реєстратори даних/системні монітори зазвичай використовують циклове планування, щоб по черзі перебирати всі датчики або джерела даних без визначення пріоритетів. Co-operative – кожен потік працюватиме до тих пір, поки не буде сказано передати керування іншому потоку або не досягне блокуючого виклику ОС. Спільну багатозадачність можна побачити в програмах, які вимагають фіксованого порядку виконання.

– Детермінована поведінка – Keil RTX5 забезпечує повністю детерміновану поведінку, що означає, що події та переривання обробляються протягом заздалегідь визначеного часу (дедлайну). Програма користувача може будуватись на основі визначених послідовнісних і часових процесів.

– ОС призначена для вбудованих систем – використовує мінімальний обсяг системної пам'яті. Keil RTX5 спеціально написаний для програм, що працюють на мікроконтролерах ARM Cortex-M. Він працює швидко і вимагає лише мінімальних ресурсів MCU з розміром постійної пам'яті всього 5 КБ.[7]

Технічні особливості:

– Засоби розробки можуть запускатись на базі: Microsoft Windows Vista, 7, 8 і 10+

– Підтримувані процесори і платформи: ARM

– Існуючі компілятори розробки: C, C++

LynxOS — Unix-подібна операційна система реального часу, розроблена для вбудовуваних систем, сумісна зі стандартами POSIX і, останнім часом, з операційною системою Linux. LynxOS використовується переважно в авіації, системах управління промисловими процесами та в галузі телекомунікацій. Це вбудована операційна система POSIX із жорстким розділенням у реальному часі, розроблена та сертифікована відповідно до стандартів безпеки FAA DO-178B/C DAL A.

LynxOS-178 є основним хостом для програм POSIX і FACE™ реального часу в рамках розробки та інтеграції LYNX MOSA.ic™. Внутрішня реалізація POSIX LynxOS-178 задовольняє профілі PSE як для спеціалізованих, так і багатопільових додатків реального часу, а також FACE™.

LynxOS і LynxOS-178 були розгорнуті в мільйонах критично важливих для безпеки додатків по всьому світу, включаючи численні військові та аерокосмічні системи. Використання LynxOS забезпечує скорочену сертифікацію та повторну сертифікацію розробленого програмного забезпечення. Поетапна сертифікація підтримує довгострокові програми з численними технічними оновленнями.

Компоненти LynxOS розраховані на абсолютний детермінізм (жорстка продуктивність у реальному часі), що означає, що вони реагують протягом певного періоду часу. Передбачуваний час відгуку гарантується навіть при інтенсивному введенні-виводі завдяки унікальній моделі багатопоточності ядра, що сприяє надзвичайно коротким та швидким програмам обробки переривань.[8]

Технічні особливості:

– Засоби розробки можуть запускатись на базі: Microsoft Windows Vista, 7, 8 і 10+, Linux, Mac OS X

– Підтримувані процесори і платформи: Motorola 68010, x86/IA-32, ARM, Freescale PowerPC, PowerPC 970, LEON

– Існуючі компілятори розробки: C, C++, ObjectAda.

MQX (Message Queue eXecutive) — це операційна система реального часу, розроблена Precise Software Technologies, Inc., і в даний час продається Synopsys, Embedded Access, Inc. і

NXP Semiconductors. Як і більшість ОСРЧ, MQX включає багатозадачне ядро з випереджувальним плануванням і швидкою відповіддю на переривання, широкі можливості для міжпроцесного зв'язку та синхронізації, а також файлової системи.

Розмір системи налаштовується, це економить простір у пам'яті комп'ютера, використовуючи мінімум 6 КБ ПЗП, включаючи ядро, переривання, семафори, черги та менеджер пам'яті.

MQX включає набір протоколів Інтернету TCP/IP, вбудовану файловою системою DOS на основі таблиці розподілу файлів (MFS), стек хоста/пристрою універсальної послідовної шини (USB), а також проектування, налагодження та інструменти аналізу продуктивності. Він підтримується популярними бібліотеками, таких як wolfSSL, для забезпечення посиленних заходів безпеки, які працюють на рівні безпеки сокетів (SSL) і безпеки транспортного рівня (TLS).

MQX зазвичай використовується у вбудованих системах. Розробка MQX відбувається на хост-машині під керуванням Unix або Windows і крос-компілює цільове програмне забезпечення для роботи на різних архітектурах цільового центрального процесора.

MQX був портований на багато платформ і зараз працює на більшості сучасних процесорів, що використовуються на ринку вбудованих систем, включаючи Kinetis, ColdFire, PowerPC, ARC, ARM, StrongARM, xScale.

Усі нові пристрої Kinetis (ARM Cortex-M4), i.MX RT (ARM Cortex-M7) і ColdFire будуть включені в безкоштовну Freescale MQX RTOS. [9]

Технічні особливості:

- Засоби розробки можуть запускатись на базі: Microsoft Windows Vista, 7, 8 і 10+, Unix
- Підтримувані процесори і платформи: Freescale Power, ColdFire, Kinetis Cortex ARM, List of Freescale products
- Існуючі компілятори розробки: C, ASM.

Nucleus — операційна система реального часу, створена Accelerated Systems, підрозділом компанії Mentor Graphics (в даний час придбана компанією Siemens), для вбудовуваних систем і для різних процесорних платформ. Набула поширення в телевізійних декодерах, мобільних

телефонах, та інших переносних та кишенькових пристроях. Nucleus використовується Garmin International у GPS-модулях, призначених для цивільної авіації, а також практично у всіх китайських телефонах на базі масово використовуваного SOC MTK, у телефонах Siemens та Benq-Siemens 6x, 7x та 8x серій, Apple iPhone як baseband прошивки, та телефони на основі платформи Samsung Bada. Також використовується в годиннику фірми UWatch.

Nucleus підтримує режим асиметричної багато процесорної обробки (AMP) і режим симетричної багато процесорної обробки (SMP) для провідних 32- і 64-розрядних гетерогенних багатоядерних SoC. Nucleus також може працювати як GOS із вбудованим гіпервізором Mentor.

Під час роботи в режимі AMP Nucleus RTOS може співіснувати з іншими екземплярами Nucleus, Linux та/або відкритими машинними програмами, поширеними на інших процесорах. У цьому режимі кожен процесор працює незалежно і веде себе як окрема система всередині SoC. Mentor Embedded Multicore Framework забезпечує міжпроцесний зв'язок між операційними системами на різних ядрах та управління життєвим циклом процесора. Робота SMP передбачає, що екземпляр Nucleus RTOS керує кількома ядрами одночасно. Nucleus може розподіляти свої операції на всі ядра багатоядерного пристрою або будь-яку підмножину ядер. Для цієї мети Nucleus пропонує підтримку API під час виконання для прив'язаної обчислювальної області, а також контрольні задачі та спорідненість переривань для призначення ядра. [10]

Технічні особливості:

- Засоби розробки можуть запускатись на базі: Microsoft Windows Vista, 7, 8 і 10+, Linux, Mac OS X
- Підтримувані процесори і платформи: ARM (Cortex-M3-M4-R4-R4F-A8-A9, ARM7-9-11), PowerPC, MIPS32-16e, microMIPS, ColdFire, SuperH
- Існуючі компілятори розробки: C++.

PikeOS поєднує операційну систему реального часу (RTOS) з платформою віртуалізації та інтегрованим середовищем розробки на основі Eclipse IDE

для вбудованих систем. Це комерційний клон сімейства мікроядра L4. PikeOS розроблено для критичних додатків безпеки та безпечних додатків з потребами в сертифікації в галузі аерокосмічної, оборонної, автомобільної, транспортної, промислової автоматизації, медицини, мережевої інфраструктури та побутової електроніки.

Ключовою особливістю PikeOS є можливість безпечного виконання додатків з різними рівнями безпеки одночасно на одній комп'ютерній платформі. Це досягається шляхом суворого просторового та часового відокремлення цих програм. Програмне розділення додатків можна розглядати як контейнер із попередньо призначеними привілеями, який може мати доступ до пам'яті, часового кванту центрального процесора, пристроїв вводу/виводу та попередньо визначеного списку служб ОС. У PikeOS термін «додаток» відноситься до виконуваного файлу, пов'язаного з бібліотекою програмного інтерфейсу програм PikeOS (API) і виконується як процес всередині програмного розділу. Реалізація програмного інтерфейсу програм PikeOS (API) дозволяє додаткам варіюватися від простих циклів керування до повноцінних паравіртуалізованих гостьових операційних систем, таких як Linux або апаратно віртуалізованого програмного забезпечення [11].

Розділи програмного забезпечення також називають віртуальними машинами (VM), оскільки всередині розділу можна реалізувати повну гостьову операційну систему, яка виконується незалежно від інших розділів і, таким чином, може забезпечувати виконання задач із змішаною критичністю. PikeOS можна розглядати як гіпервізор типу 1.

Технічні особливості:

– Засоби розробки можуть запускатись на базі: Microsoft Windows Vista, 7, 8 і 10+, Linux, Mac OS X

– Підтримувані процесори і платформи: ARM, PowerPC, x86, SPARC

– Існуючі компілятори розробки: Python, C++, C, Perl.

ThreadX спочатку була розроблена та продана фірмою Express Logic, зараз підтримується Microsoft. ThreadX забезпечує випереджувальне планування на основі

пріоритетів, швидку відповідь на переривання, керування пам'яттю, міжпотоківу взаємодію, сповіщення про події та функції синхронізації потоків. Основні технологічні характеристики ThreadX включають поріг випередження, успадкування пріоритетів, ефективне керування таймерами, забезпечують реалізацію швидких програмних таймерів, якісний дизайн пікоядра, швидкий ланцюжок подій і малий їх розмір: мінімальний розмір ядра для процесора архітектури ARM становить близько 2КБ.

ThreadX підтримує багатоядерні процесори за допомогою асиметричної багатопроцесорної обробки (AMP) або симетричної багатопроцесорної обробки (SMP). Ізоляція потоків програм за допомогою модуля керування пам'яттю (MMU) або захисту пам'яті блоку захисту пам'яті (MPU) можлива з модулями ThreadX.

ThreadX має сертифікати безпеки від Technischer Überwachungsverein (TÜV), і UL (Underwriters Laboratories) і відповідає вимогам MISRA C Асоціації надійності програмного забезпечення автомобільної промисловості.

ThreadX є основою платформи X-Ware Internet of things (IoT) Express Logic, яка також включає підтримку вбудованої файлової системи (FileX), підтримку вбудованого інтерфейсу користувача (GUIX), набір вбудованих протоколів Інтернету (TCP/IP) і хмарне підключення (NetX) та підтримку універсальної послідовної шини (USB). ThreadX отримав високу оцінку розробників і є дуже популярною ОСРЧ. Станом на 2022 рік, за даними компанії з маркетингових досліджень VDC Research, ОСРЧ ThreadX стала однією з найпопулярніших ОСРЧ у світі, будучи розгорнута в понад 6,2 мільярда пристроїв, включаючи побутову електроніку, медичні пристрої, програми для передачі даних і SoC [12].

ThreadX поширюється з використанням маркетингової моделі, в якій надається вихідний код, а самі ліцензії безкоштовні. Дистрибутив програмного забезпечення містить підготовлені конфігурації та демонстрації для кожного порту та компілятора, що дозволяє швидко проектувати додаток.

Технічні особливості:

– Засоби розробки можуть запускатись на базі: Microsoft Windows Vista, 7, 8 і 10+

- Підтримувані процесори і платформи: ARC, ARM, Blackfin, CEVA, C6x, MIPS, NXP, PIC, PowerPC, RISC-V, RX, SH, SHARC, TI, V850, Xtensa, x86, Coldfire, others

- Існуючі компілятори розробки: C++, C.

μC/OS – Операційна система для мікроконтролерів (MicroC/OS, або скорочено μC/OS) — це операційна система реального часу (RTOS), розроблена у 1991 році. Вона містить ядро реального часу з використанням пріоритетної обробки переривань та механізму передбачення навантаження, написане переважно мовою програмування C. Призначена для використання у вбудованих системах.

MicroC/OS дозволяє визначити кілька функцій мовою C, кожна з яких може виконуватися як незалежний потік або завдання. Кожне завдання виконується з своїм пріоритетом і виконується так, ніби воно повністю володіє центральним процесором. Завдання з нижчим пріоритетом можуть бути витіснені завданнями з вищим пріоритетом у будь-який час. Завдання з вищим пріоритетом використовують служби операційної системи (наприклад, затримка або подія), щоб дозволити виконувати завдання з нижчим пріоритетом. Служби ОС надаються для керування завданнями та пам'яттю, обміну даними між завданнями та визначенням часу.

Особливості операційної системи [13]:

- Портативність. Пропонуючи максимально можливу простоту використання, ядра μC/OS постачаються з повним вихідним кодом і детальною документацією. Ядра μC/OS працюють на значній кількості процесорних архітектур.

- Масштабованість. Ядра μC/OS дозволяють виконувати значну кількість завдань і об'єктів ядра. Обсяг пам'яті ядер можна зменшити, щоб містити лише функції, необхідні для вашої програми, як правило, 6–24 КБ простору коду та 2–5 КБ простору даних.

- Надійність. Ядра μC/OS можуть включати функцію відладки, яка скорочує час розробки. Ядра забезпечують широку перевірку діапазону вхідних даних, включаючи перевірку показників, переданих при викликах API, перевірку параметрів управління і перевірку на рівні завдань із ISR, а також аргументів в межах допустимого діапазону.

- Ефективність. Ядра Micrium також містять цінну статистику часу виконання, що робить внутрішні елементи програми доступними для спостереження. За допомогою цієї інформації можна визначити вузькі місця в продуктивності та оптимізувати використання процесора ще на початку циклу розробки.

Технічні особливості:

- Засоби розробки можуть запускатися на базі: Microsoft Windows Vista, 7, 8 і 10+, Linux, Mac OS X

- Підтримувані процесори і платформи: Nios II SoC FPGA (Cortex-A), Blackfin ADSP-CM4xx (Cortex-M), ARM7, ARM9, ARM11, Cortex-A5, Cortex-A7, Cortex-A8, Cortex-A9, Cortex-A15, Cortex-A17, Cortex-A53, Cortex-A57, Cortex-R4, Cortex-R5, Cortex-R7, Cortex-M0, Cortex-M1, Cortex-M3, Cortex-M4(F), Cortex-M7, PSoC 4, PSoC 5, eSi-RISC, M14K, XMC4000, PIC24, PIC32, SmartFusion2, STM32F, STR9, MSP430, MSP432, Hercules RM, Hercules TMS570, Zynq-7000, Zynq Ultrascale+ MPSoC

Існуючі компілятори розробки: C++, C

VxWorks — операційна система реального часу, що розробляється компанією Wind River Systems (США), орієнтована на використання у комп'ютерах, що працюють в системах жорсткого реального часу. VxWorks є системою із крос-засобами розробки прикладного програмного забезпечення.

Як і більшість інших ОС реального часу, VxWorks включає багатозадачне ядро з витісняючим планувальником і швидким відгуком на переривання, засоби міжпроцесної взаємодії та синхронізації, а також файлову систему і мережеву підсистему (стек протоколів TCP/IP). У комплект поставки входять засоби для крос-компіляції, моніторингу продуктивності (WindView), віддаленого символічного налагодження, а також емуляції різних процесорів. Додатково постачається значна кількість різних стеків протоколів, графічних підсистем та ін. як від самої Wind River Systems, так і від третіх фірм. Кількість підтримуваних VxWorks вбудовуваних платформ є однією з найбільших серед операційних систем реального часу [14].

VxWorks має архітектуру клієнт-сервер і, як і більшість ОС жорсткого реального часу, побудована за технологією мікроядра. На нижньому рівні ядра (WIND Microkernel)

виконуються лише базові функції планування задач та управління комунікацією/синхронізацією між завданнями. Всі інші функції ОСРЧ вищого рівня - управління пам'яттю, мережеві засоби і т. д. - реалізуються через найпростіші функції нижнього рівня. За рахунок такої ієрархічної організації досягається швидкодія та детермінованість ядра системи, також це дозволяє легко будувати необхідну конфігурацію операційної системи.

VxWorks можна скомпонувати як для систем із жорсткими апаратними обмеженнями, так і для систем із розвинутою функціональністю. Окремі модулі системи можуть бути масштабованими. При складанні системи можна відключати певні функції системи, не потрібні в даний момент, також можна прибирати специфічні об'єкти синхронізації ядра, якщо в них немає потреби. Але, незважаючи на те, що система конфігурується, не можна сказати, що в ній реалізований компонентний підхід, тому що всі модулі побудовані над базовим ядром і не можуть бути використані в інших середовищах.

Ядро VxWorks має такі властивості:

- кількість завдань обмежується лише пам'яттю;
- має 256 рівнів пріоритетів задач;
- планування завдань організовано двома способами: витіснення за пріоритетами та циклічне;
- взаємодія завдань відбувається через черги повідомлень, семафори, події та канали (для взаємодії між процесами), сокети та віддалені виклики процедур (для мережевої взаємодії), поділ пам'яті (використовується для поділу даних) та сигнали (для управління винятковими ситуаціями);
- забезпечується кілька видів семафорів для управління критичними системними ресурсами
- двійкові, обчислювальні та взаємовиключні з пріоритетним успадкуванням; можливе детерміноване перемикання контексту.

Технічні особливості:

- Засоби розробки можуть запускатись на базі: Microsoft Windows 7, 8 і 10+, Linux, Mac OS X

- Підтримувані процесори і платформи: Motorola 680x0 и CPU32, PowerPC, Intel 386 / 486/ Pentium, Intel 960, Sparc, Mips R3000/4000, AMD 29K, Motorola 88110, HP PA-RISC, Hitachi SuperH, DEC Alpha

Існуючі компілятори розробки: C11 і C++17, Boost C++ libraries, Rust, Python

Zephyr – ОС Zephyr походить від Virtuoso RTOS для цифрових сигнальних процесорів (DSP). Операційна система Zephyr використовує підтримку дерева пристроїв (DTS) лише під час компіляції коду. Потужні інструменти конфігурації дозволяють гнучко включати лише ті функції, які дійсно потрібні в конкретних програмах. Об'єм пам'яті може становити всього 8 Кб. Інша особливість операційної системи Zephyr полягає в тому, що вона має лише один адресний простір, що означає, що код програми та ядро об'єднані в одній двійковій компіляції.

Створення ланцюжка інструментів: Будучи по суті кросплатформним проектом, операційна система Zephyr, природно, використовує систему збірки CMake. Він також широко використовує інструмент командного рядка під назвою «West», який використовується для створення, перепрограмування та налагодження програм, а також менеджер репозиторію Zephyr.

Підтримка давачів: операційна система Zephyr має розширену систему підтримки сенсорів з високим рівнем абстракції. Процес проходить успішно під час ввімкнення датчиків, які вже підтримуються операційною системою Zephyr. При цьому підтримується ряд плат і давачів, однак процес ускладнюється, якщо новий давач, якого немає у списку, потрібно ввімкнути на платі користувача.[15]

Технічні особливості:

- Засоби розробки можуть запускатись на базі: Microsoft Windows 7, 8 і 10+, Linux, Mac OS X

- Підтримувані процесори і платформи: ARM (Cortex-M0, -M1, -M3, -M4, -M7, -M23, -M33, -R4, -R5, -R82, -A9, -A53), ARC, MIPS, Nios II, RISC-V, Xtensa, SPARC, x86, x86-64

Існуючі компілятори розробки: C і C++, Python

Основні характеристики операційних систем реального часу зведені в таблиці 1.

Таблиця 1 Основні характеристики основних ОСРЧ

Назва	Ліцензія	Код	Платформа	Примітки
Deos	Proprietary	closed	x86, PowerPC, MIPS	Придатна для авіаційних застосунків, підтримка мови Ada
embOS	Proprietary	closed, available to customers, free object code for non-commercial use	ARM7/9/11, ARM Cortex-A/R/M, AVR, AVR32, C16x, CR16C, ColdFire, H8, HCS12, M16C, M32C, MSP430, NIOS2, PIC18/24/32, R32C, R8C, RISC-V, RL78, RH850, RX100/200/600/700, RZ, SH2A, STM8, ST7, V850, 78K0, 8051	Підтримка малопотужних процесорів
FreeRTOS	MIT	open source	ARM, AVR, AVR32, ColdFire, ESP32, HCS12, IA-32, Cortex-M3-M4-M7, Infineon XMC4000, MicroBlaze, MSP430, PIC, PIC32, Renesas H8/S, RISC-V, RX100-200-600-700, 8052, STM32, TriCore, EFM32	Підтримка малопотужних процесорів
Integrity	Proprietary	closed	ARM, XScale, Blackfin, ColdFire, MIPS, PowerPC, x86	Придатна для авіаційних застосунків, підтримка мови Ada, підтримка FPGA, цифрових сигнальних процесорів
Keil RTX	Apache 2.0	open source	ARM	
LynxOS	Proprietary	source available	Motorola 68010, x86/IA-32, ARM, Freescale PowerPC, PowerPC 970, LEON	Придатна для авіаційних застосунків, підтримка мови ObjectAda
MQX	Proprietary	complimentary source available	Freescale Power, ColdFire, Kinetis Cortex ARM, List of Freescale products	
Nucleus	Proprietary	source provided	ARM (Cortex-M3-M4-R4-R4F-A8-A9, ARM7-9-11), PowerPC, MIPS32-16e, microMIPS, ColdFire, SuperH	
Neutrino	Proprietary	some source provided	ARM, MIPS, PPC, SH, x86, XScale	
PikeOS	Proprietary	available to customers	PPC, x86, ARM, MIPS, SPARC-LEON	Підтримка мови Perl, розширені можливості віртуалізації
ThreadX	Proprietary	partial source available	ARC, ARM/Thumb, AVR32, BlackFin, 680x0-ColdFire, H8-300H, Luminary Micro Stellaris, M-CORE, MicroBlaze, PIC24dsPIC, PIC32, MIPS, V8xx, Nios II, PowerPC, Renesas RX100, RX200, RX600, RX700, Synergy, SH, SHARC, StarCore, STM32, StrongARM, TMS320C54x, TMS320C6x, x86/x386, XScale, Xtensa/Diamond, ZSP	Підтримка цифрових сигнальних процесорів
μC/OS	GNU GPL	open source	AVR, ARM, MIPS32, MSP430, Intel i386, Linux386	Підтримка малопотужних процесорів, підтримка FPGA, цифрових сигнальних процесорів

VxWorks	Proprietary	open source	ARM, IA-32, Intel 64, MIPS, PowerPC, SH-4, StrongARM, xScale	Значна кількість мов програмування, якими можна здійснювати розробку
Zephyr	Apache 2.0	open source	ARM (Cortex-M0, -M3, -M4, -M23, -M33, -R4, -R5, -A53), x86, ARC, RISC-V, Nios II, Xtensa, SPARC	Підтримка списку давачів

Результати досліджень характеристик найбільш розповсюджених ОСРЧ, придатних для створення систем і пристроїв технічної діагностики показали, що визначення типу операційної системи повинне здійснюватися на основі потреби забезпечення необхідної швидкодія обробки вхідних даних і обчислювальної потужності мікропроцесора системи діагностування. Крім того, значну роль грає мобільність і автономність системи, що прямо пов'язана із споживаною електричною потужністю, і визначає потребу в наявності автономного живлення. Також грає роль собівартість пристрою, що буде включати також вартість ліцензії ОСРЧ. Від цього залежить також вибір елементної бази, з високою або невеликою швидкодією, об'ємом пам'яті і т.д. У спеціальних випадках визначальною є галузь застосування, яка накладає додаткові обмеження на надійність функціонування операційної системи і вимагає наявності сертифікації безпеки, які визначаються спеціальними стандартами. Потреба використання типових давачів та вимірювальних перетворювачів і накладені обмеження на час розробки системи, визначає вибір операційної системи з будованою підтримкою таких давачів.

Висновки. Таким чином, у випадку потреби розробки системи діагностування для нафтогазового комплексу за умови обмеженого часу доцільно використовувати ОСРЧ Zephyr, яка має підтримку широкого списку вимірювальних сенсорів та перетворювачів. Для випадку об'єктів ядерної енергетики доцільно використання ОСРЧ типу LynxOS, Integrity, Deos, які мають потрібну сертифікацію, і засоби які забезпечують безпечну поведінку у випадку несприятливого середовища виконання. У разі, якщо розроблювана система не містить високочастотних сигналів, (наприклад не є приладом ультразвукової та радіочастотної

діагностики), повинна бути виконана мінімально дешево, слід використовувати ОСРЧ μ C/OS, FreeRTOS, embOS. У випадку потреби проектування системи яка вимагає широкої масштабованості, корпоративну підтримку, із значними обчислювальними ресурсами, і проект забезпечений значними фінансовими ресурсами, слід використати операційну систему ThreadX. Цей вибір доцільний, також, у випадку потреби обробляти середньо-швидкісну вхідну інформацію (сигнали до частот 300кГц). У випадку, якщо слід працювати із високочастотними сигналами, доцільно використовувати операційні системи Integrity, μ C/OS, які підтримують роботу з програмованими логічними матрицями FPGA.

Література / References

1. Comparison of real-time operating systems – Режим доступу: https://en.wikipedia.org/wiki/Comparison_of_real-time_operating_systems (дата доступу 08.05.2022)
2. I. Ferguson. What Are the Most Popular Real-Time Operating Systems? – Режим доступу: <https://www lynx.com/embedded-systems-learning-center/most-popular-real-time-operating-systems-rtos> (дата доступу 08.05.2022)
3. Deos, a Time & Space Partitioned, Multi-core Enabled, RTOS Verified to DO-178C/ED-12C DAL A. – Режим доступу: https://www.ddci.com/products_deos_do_178c_ari_nc_653/ (дата доступу 08.05.2022)
4. embOS — The leading RTOS (real-time operating system) – Режим доступу: <https://www.segger.com/products/rtos/embos/> (дата доступу 08.05.2022)
5. FreeRTOS. Real-time operating system for microcontrollers. – Режим доступу: <https://www.freertos.org/> (дата доступу 08.05.2022)

6. INTEGRITY RTOS. – Режим доступу:
<https://www.ghs.com/products/rtos/integrity.html>
(дата доступу 08.05.2022)

7. RTX Real-Time Operating System. –
Режим доступу: <https://www.keil.com/arm/rl-arm/kernel.asp> (дата доступу 08.05.2022)

8. LynxOS. – Режим доступу:
<https://www.lynx.com/products/lynxos-posix-real-time-operating-system-rtos> (дата доступу 08.05.2022)

9. MQX Real-Time Operating System (RTOS). – Режим доступу:
<https://www.nxp.com/design/software/embedded-software/mqx-software-solutions/mqx-real-time-operating-system-rtos:MQXRTOS> (дата доступу 08.05.2022)

10. Nucleus RTOS. The Real-time Operating System for Today's Advanced Designs. – Режим доступу:
<https://www.plm.automation.siemens.com/global/ru/products/embedded/nucleus-rtos.html> (дата доступу 08.05.2022)

11. PikeOS RTOS & Hypervisor. – Режим доступу: <https://www.sysgo.com/pikeos> (дата доступу 08.05.2022)

12. Azure RTOS ThreadX documentation. – Режим доступу: <https://docs.microsoft.com/en-us/azure/rtos/threadx/> (дата доступу 08.05.2022)

13. Micrium Real-Time Kernel. – Режим доступу:
<https://www.silabs.com/developers/micrium-os/micrium-real-time-kernel> (дата доступу 08.05.2022)

14. VxWorks. The Leading RTOS for the Intelligent Edge. – Режим доступу:
<https://www.windriver.com/products/vxworks> (дата доступу 08.05.2022)