

1. Мікропроцесорна схема для організації доступу до записаних в запам'ятовувачі (2) даних чи програм, що містить щонайменше один мікропроцесор (1), запам'ятовувач (21) для операційної системи і щонайменше один запам'ятовувач (22, 23, 24) для вільного програмування з індивідуальними зовнішніми програмами, причому в запам'ятовувачі (22, 23, 24) для вільного програмування передбачено кілька областей пам'яті з відповідними адресними просторами, причому кожному адресному простору присвоєний кваліфікатор (програмний ідентифікатор), а також засоби (7, 8), виконані з можливістю завантажування присвоєної області пам'яті кваліфікатора (програмного ідентифікатора) до першого допоміжного регістра (72) кожного разу при звертанні до відповідної області пам'яті перед адресацією області пам'яті, завантажування кваліфікатора (програмного ідентифікатора) адресованої області пам'яті до другого допоміжного регістра (82) і порівняння вмістів першого і другого допоміжних регістрів (72, 82), яка **відрізняється** тим, що кожному адресному простору області пам'яті поставлена у відповідність біт-послідовність (ACR), що містить щонайменше одне право доступу.
2. Мікропроцесорна схема за п. 1, яка **відрізняється** тим, що кожній адресі адресного простору поставлена у відповідність біт-послідовність (ACR), що містить права доступу.
3. Мікропроцесорна схема за п. 1 або 2, яка **відрізняється** тим, що біт-послідовності (ACR), що містять права доступу, разом з адресами чи блоками адрес і кваліфікаторами (програмними ідентифікаторами) занесені в таблицю, записану в пам'яті.
4. Мікропроцесорна схема за п. 3, яка **відрізняється** тим, що адресний блок позначений початковою адресою чи кінцевою адресою і довжиною адресного блока.
5. Мікропроцесорна схема за п. 3, яка **відрізняється** тим, що адресний блок позначений нижньою і верхньою адресами.
6. Мікропроцесорна схема за п. 1 або 2, яка **відрізняється** тим, що кожний адресний простір чи кожна адреса містить біт-послідовність (ACR), що містить права доступу.
7. Мікропроцесорна схема за одним із пп. 1-6, яка **відрізняється** тим, що біт-послідовність (ACR), яка містить права доступу, вміщує перше і друге право доступу, причому перше право доступу регулює доступи між двома областями (22, 23, 24) пам'яті, а друге право доступу регулює доступи всередині однієї області (22, 23, 24) пам'яті.
8. Мікропроцесорна схема за п. 7, яка **відрізняється** тим, що в біт-послідовності передбачено третє право доступу, яке регулює доступи всередині однієї області (22, 23, 24) пам'яті.
9. Мікропроцесорна схема за одним із пп. 1-8, яка **відрізняється** тим, що передбачений проміжний запам'ятовувач прав, що містить інші права доступу між двома областями пам'яті, виконаний з можливістю зчитування і запису прав доступу операційною системою.
10. Мікропроцесорна схема за п. 9, яка **відрізняється** тим, що проміжний запам'ятовувач вміщує дозволені присвоєння (доступи) між двома областями пам'яті.
11. Мікропроцесорна схема за п. 9 або 10, яка **відрізняється** тим, що присвоєння в таблиці реалізовані у вигляді кваліфікаторів (програмних ідентифікаторів) або у вигляді таблиці відповідності, за допомогою якої дані означення кваліфікаторів.
12. Мікропроцесорна схема за одним із пп. 1-11, яка **відрізняється** тим, що вона містить область (25) пам'яті, керовану виключно операційною системою.
13. Мікропроцесорна схема за одним із пп. 1-12, яка **відрізняється** тим, що вона містить поставлені у відповідність кожній області (22, 23, 24) пам'яті адресні області (27, 28, 29) для проміжного запису даних, виконані з можливістю керування лише від програм, записаних у відповідних областях (22, 23, 24) пам'яті.
14. Спосіб організації доступу до записаних в запам'ятовувачі (2) даних чи програм у мікропроцесорній схемі, що містить щонайменше один мікропроцесор (1), запам'ятовувач (21) для операційної системи і щонайменше один запам'ятовувач (22, 23, 24) для вільного програмування з індивідуальними зовнішніми програмами, причому в запам'ятовувачі (22, 23, 24) для вільного програмування передбачено кілька областей пам'яті з відповідними адресними просторами, причому кожному адресному простору присвоєний кваліфікатор (програмний ідентифікатор), і причому в кожній області пам'яті (22, 23, 24) записана одна зовнішня програма, який включає такі етапи:
 - a) визначення першого програмного ідентифікатора (PID_{pc}) тільки що виконаної кодової команди зовнішньої програми за допомогою кваліфікатора поточної адреси,
 - b) визначення другого програмного ідентифікатора (PID_{adr}) адресованої області пам'яті,
 - c) порівняння першого і другого програмних ідентифікаторів,
 - d) вибір першого чи другого права доступу (ACR) в залежності від результату порівняння на етапі c),
 - e) оцінка права доступу,
 - f1) продовження виконання програмного коду, якщо виконання кодової команди чи доступ до адресованої області пам'яті дозволено,
 - f2) виклик програми обробки помилок, якщо виконання кодової команди чи доступ до адресованої області пам'яті не дозволено.
15. Спосіб за п. 14, який **відрізняється** тим, що при однакових першому і другому програмних ідентифікаторах - в залежності від права доступу тільки що виконаної кодової команди - вибирають друге або третє право доступу.
16. Спосіб за п. 14 або 15, який **відрізняється** тим, що після етапу f2) виконують такі кроки:
 - g) перевірка проміжного запам'ятовувача прав на наявність запису, який представляє дозволений доступ області пам'яті з першим програмним ідентифікатором (PID_{pc}) до області пам'яті, позначеної другим програмним ідентифікатором (PID_{adr}),
 - h1) продовження виконання програмного коду, якщо такий запис у проміжному запам'ятовувачі прав є,
 - h2) виклик програми обробки помилок, якщо такий запис відсутній.
17. Спосіб за одним із пп. 14-16, який **відрізняється** тим, що кодова команда етапу a) є командою переходу, причому після цього виконують такі етапи:

- b) визначення другого програмного ідентифікатора (PID_{ADR}) адресованої області пам'яті,
 - c) порівняння першого і другого програмних ідентифікаторів,
 - d) перехід на викликану адресу,
 - e1) продовження виконання програмного коду в адресованій області пам'яті, якщо перший і другий програмні ідентифікатори однакові,
 - e2) зчитування адресного вмісту адресованої області пам'яті, якщо перший і другий програмні ідентифікатори не однакові,
 - aa) виклик програми обробки помилок, якщо зчитаний адресний вміст не є командою входу,
 - bb) продовження виконання програмного коду, якщо зчитаний адресний вміст є командою входу.
18. Спосіб за п. 17, який **відрізняється** тим, що перед етапом d) виконують етап запису адреси щойно виконаної команди переходу до проміжного запам'ятовувача (25...29).
19. Спосіб за п. 17 або 18, який **відрізняється** тим, що команда входу є сталою попередньо заданою біт-послідовністю.
20. Спосіб за одним із пп. 14-19, який **відрізняється** тим, що проміжним запам'ятовувачем (25) керують виключно зі сторони операційної системи.
21. Спосіб за одним із пп. 14-20, який **відрізняється** тим, що як проміжний запам'ятовувач для кодових команд, яким присвоєно перше право виконання, використовують загальнодоступний проміжний запам'ятовувач (26).
22. Спосіб за одним із пп. 14-21, який **відрізняється** тим, що для кодових команд, яким присвоєно друге право виконання, використовують поставлений у відповідність областям пам'яті проміжний запам'ятовувач (27, 28, 29), керований лише програмою, записаною у відповідній області пам'яті, і операційною системою.