Підвищення захисту від НСД до програмних кодів мікроконтролерів систем безпеки

КЕРІВНИК: Д. Т. Н. ПРОФ. КАФ. МБІС

ЯРЕМЧУК Ю. Є

РОЗРОБИВ: СТ. ГР. УБ-18М

СКУТЕЛЬНИК І. В

Актуальність і мета роботи

Мікроконтролери є основою функціонування будь-якого пристрою, в коло їх задач входить функція керування різноманітними периферійними пристроями об’єднуючи в складні системи які здатні вирішувати широке коло задач з різних галузей людської діяльності.

Важливим і відкритим лишається питання безпека систем, що побудовані на базі мікроконтролерів на апаратному рівні. Оскільки мікроконтролер є основною керуючою ланкою. Збиток який може нанести несанкціонований доступ до програмного достатньо очевидний. Так як отримавши контроль над системою зловмисник може отримати доступ важливих відомостей, починаючи з конфіденційних і закінчуючи відомостями особливої важливості. А це може призвести до непередбачуваних наслідків.

Мета роботи: Удосконалення методів захисту від несанкціонованого доступу програмних кодів мікроконтролерів систем безпеки.

Задачі

- проаналізувати атаки, що створюють загрозу програмним кодам мікроконтролерів;
- проаналізувати існуючі методи захисту мікроконтролерів;
- проаналізувати сучасну базу мікроконтролерів та платформ на базі них, вибір мікроконтролерів для розробки захисту, виявлення основних векторів атак на них;
- розробити методи та засоби захисту програмних кодів обраних мікроконтролерів і платформ з урахуванням їх можливостей та особливостей;
- реалізувати розроблені елементи методів захисту програмних кодів обраних мікроконтролерів і платформ.

Поняття мікроконтролера

Мікроконтролер (МК) – це аналого-цифровий однокристальний комп’ютер призначений для керування пристроями

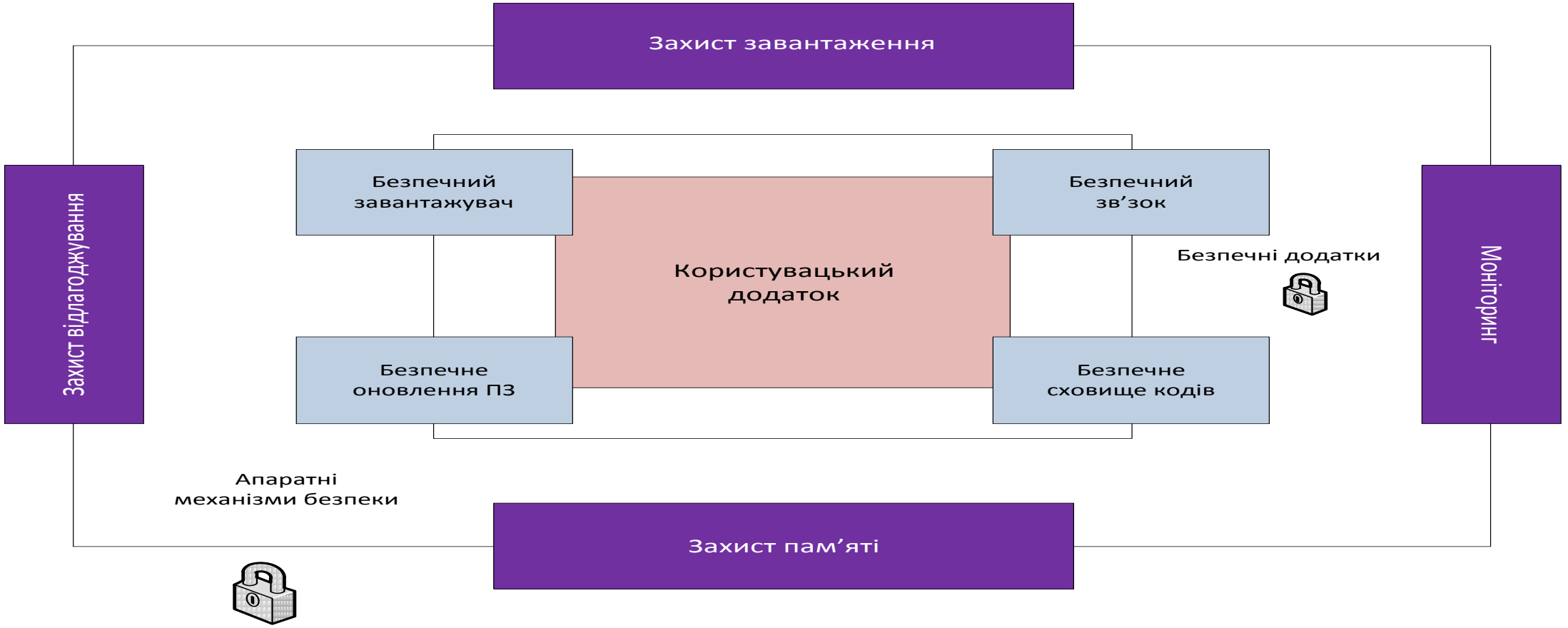
Містить:
- процесор;
- енергозалежну пам’ять;
- енергонезалежну пам’ять;
- периферію.

Периферія:
- генератори тактових сигналів;
- таймери (лічильники, вартові тощо);
- компаратори;
- підсилювачі;
- цифрові порти вводу-виводу (GPIO);
- АЦП;
- ЦАП;
- датчики (температури, тиску тощо);
- контролери ПДП (DMA);
- послідовні шини (UART,SPI,I2C тощо).

Загрози системам побудованих на базі мікроконтролерів

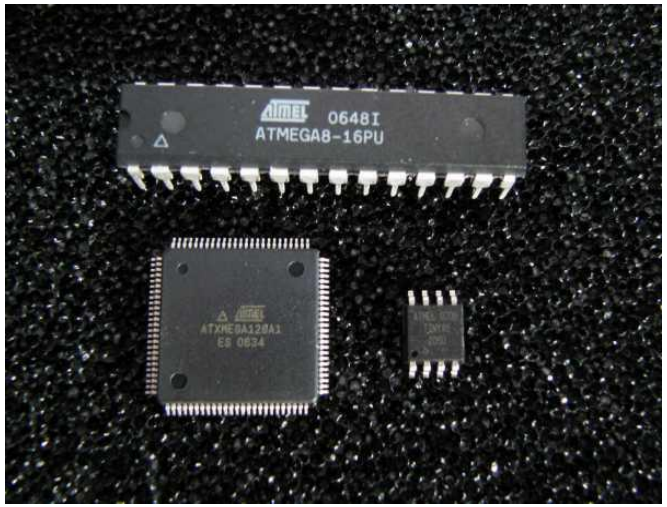
Тип атаки	Атака програмного забезпечення	Апаратна неінвазивна	Апаратна інвазивна
Умови	Локально або віддалено	Потрібно доступ до зібраного пристрою або до друкованої плати	Потрібно доступ до друкованої плати
Методики	Помилки в ПЗ, Уразливості інтерфейсів, Сканування інтерфейсів, Фаззінг	Налагоджувальний порт, Комунікаційні порти, Впроваджений код (code injection), Перешкоди по ланцюгах живлення (power glitches), Позаштатне тактування (clock glitches), Аналіз побічних ефектів (side-channel attack)	Зондування кристала, Лазерне різання, Травлення, Іонне травлення (FIB), Оптична та електронна мікроскопія, Інжекція перешкод (fault injection)
Вартість	Від дуже низької до високої, в залежності від рівня безпеки, наміченого при розробці вбудованого ПЗ	Від середньої до високої. Потрібно обмежений набір відносно нескладного устаткування і персонал, що володіє середньою кваліфікацією	Дуже висока. Необхідно дороге спеціалізоване обладнання та персонал, що володіє експертної кваліфікацією
Цілі	Доступ до конфіденційної інформації (код і дані), перехоплення даних, відмова обладнання	Доступ до конфіденційної інформації і визначення алгоритмів роботи пристрою	Реверс-інжиніринг (отримання повного пакету документації для відтворення пристрою)

Методи протидії атакам спрямованим на порушення безпеки в системах побудованих на базі мікроконтролерів



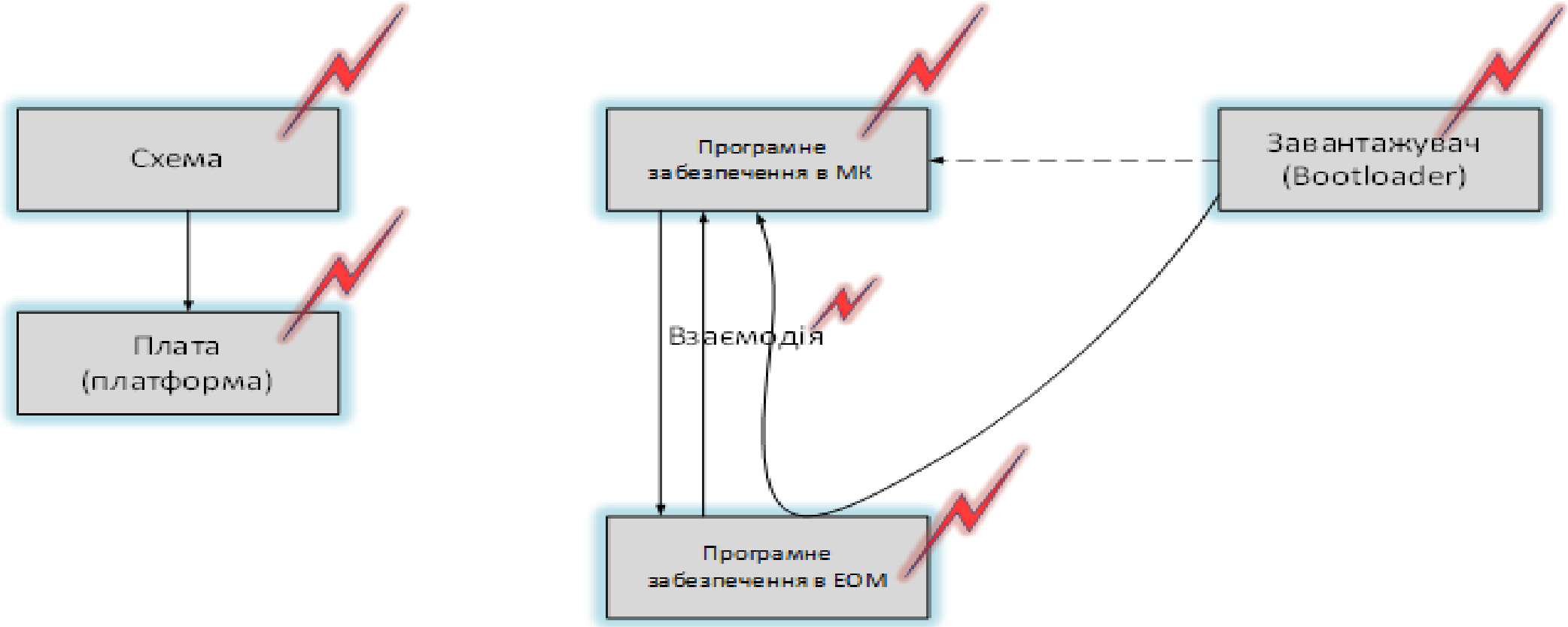
Вибір мікроконтролерів для захисту

Більш гнучкими та універсальними є мікроконтролери AVR компанії Microchip та STM32 на базі ядра ARM Cortex компанії STMicroelectronics. Дані сімейства знаходять МК знаходять застосування майже у всіх галузях людської діяльності. Зокрема, значну роль дані сімейства відіграють у побудові та функціонуванні систем захисту інформації.



Загрози програмним кодам мікроконтролерів AVR і STM32

Структурна схема системи побудованої на базі мікроконтролера



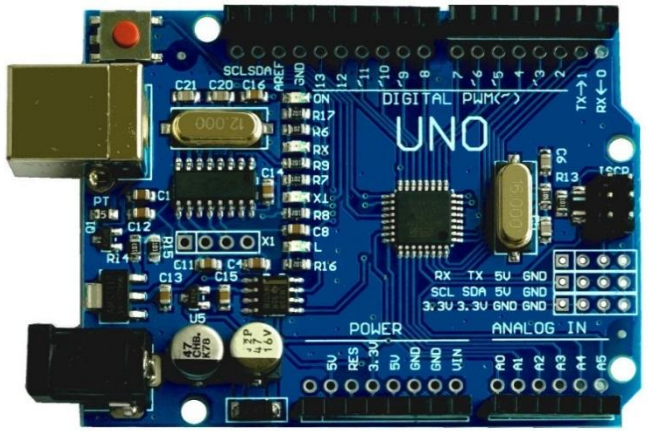
Захист програмного коду мікроконтролерів ATMega і STM32

Проаналізувавши вектори ймовірних небезпек які спричинять загрозу безпеці системи і врахувавши особливості мікроконтролера ATMega на базі платформи Arduino пропонується захист таких елементів системи:

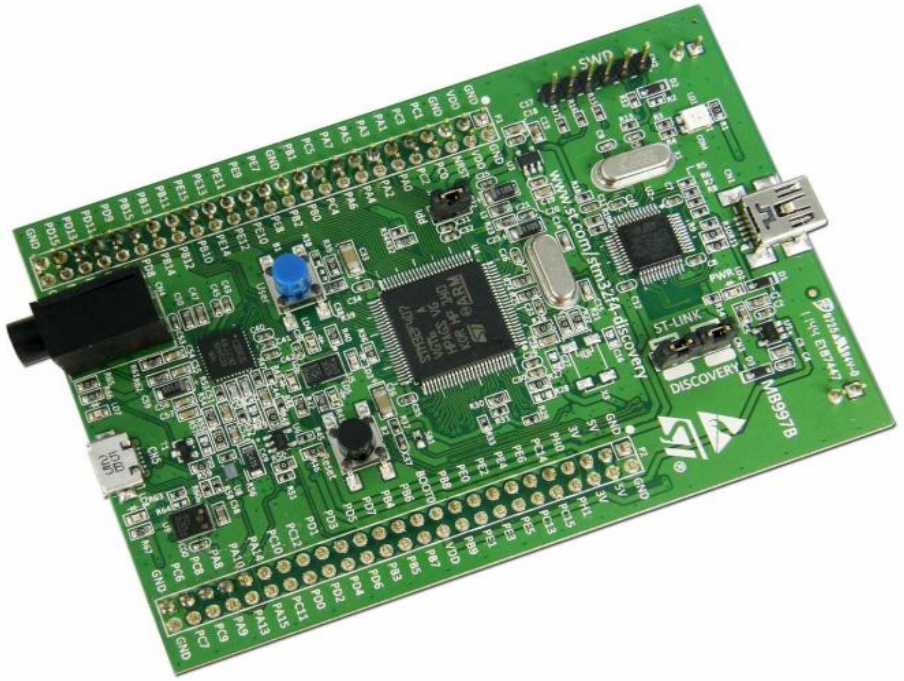
- схема/плата (платформа);
- програма в мікроконтролері;
- взаємодія ЕОМ і мікроконтролера;
- завантажувач (bootloader).

Для розробки захисту було обрано популярну платформу Arduino Uno R3 на основі мікроконтролера ATMega 328 та платформу STM32 Discovery. Дані платформи містять у собі все необхідне для роботи з мікроконтролером і легко взаємодіють з ЕОМ за допомогою середовища програмування. Окрім того багато різних проектів і систем може бути реалізовано на таких платформах, тому важливим є захист цих платформ. Гнучкість платформ дозволить в майбутньому адаптувати розроблений захист до мікроконтролерів AVR і STM32 які програмуються окремо від платформ за допомогою зовнішніх рішень, або адаптувати запропонований захист до інших платформ

Вигляд платформ



Arduino UNO R3



STM32 Discovery

Захист схеми (платформи)

Платформи Arduino і STM32 є відкритим проектом, з відкритою документацією по схемі та конструкції. Тому у зловмисника не виникне певних ускладнень в аналізі таких платформ. Проте для подальшого виключення ймовірності під'єднання до мікроконтролера чи окремих блоків платформи які забезпечують коректну роботу пристрою пропонується ізолювати від можливого такого підключення. Ізоляцію мікроконтролера та блоків які забезпечують зв'язок мікроконтролера з ЕОМ, блоків живлення на платі пропонується компаундом.

Компаунд - термоактивна, термопластична полімерна смола (яка твердне в природних умовах). Використовується в якості електроізоляційного матеріалу і як засіб вибухозахисту.

Таким чином, нанесення на важливі частини платформи ускладнить, а в деяких випадках і виключить можливість фізичного доступу до важливих блоків платформи, зокрема мікроконтролера.

Захист програми в мікроконтролері

В більшості мікроконтролерів захист програм у ньому передбачена виробниками. Однак у випадку використання платформи Arduino цей захист по замовчуванню при записі програми в мікроконтролер не активується.

Захист програми в мікроконтролері на рівні вбудованого захисту реалізується за допомогою спеціальних конфігураційних бітів. В платформі Arduino для виключення виходу з ладу плати можливість їх програмувати прихована, оскільки дана платформа використовується також і серед непрофесійних людей з низькою кваліфікацією. Проте при роботі з платами Arduino самими розробниками цих платформ передбачена можливість запрограмувати конфігураційні, оскільки вони використовують мікроконтролери AVR, а середовище програмування використовує всі рішення для програмування мікроконтролерів цього виду.

Захист програми в мікроконтролері STM32 на рівні вбудованого захисту реалізується за допомогою спеціальних option bytes. Option bytes налаштовуються розробником прошивки в залежності від потреб. Це кілька байт, в різних контролерів їх кількість різний, за допомогою яких можна встановити деякі настройки і захистити флеш-пам’ять від запису і читання. Захистивши потрібні сторінки флеш-пам’яті від запису, можна уникнути помилкового пошкодження програми під час роботи. Установка захисту від зчитування унеможливлює зчитування і копіювання прошивки, так як при спробі зняти біти захисту спрацює стирання мікроконтролера.

Контроль цілісності даних

Цілісність даних це показник, того що дані не були змінені при зберіганні, передачі, відображенні і т.п. Цілісність даних може перевірятися в будь-якій ситуації, потенційно небезпечної для руйнування даних. Зазвичай контроль достовірності даних виконують:

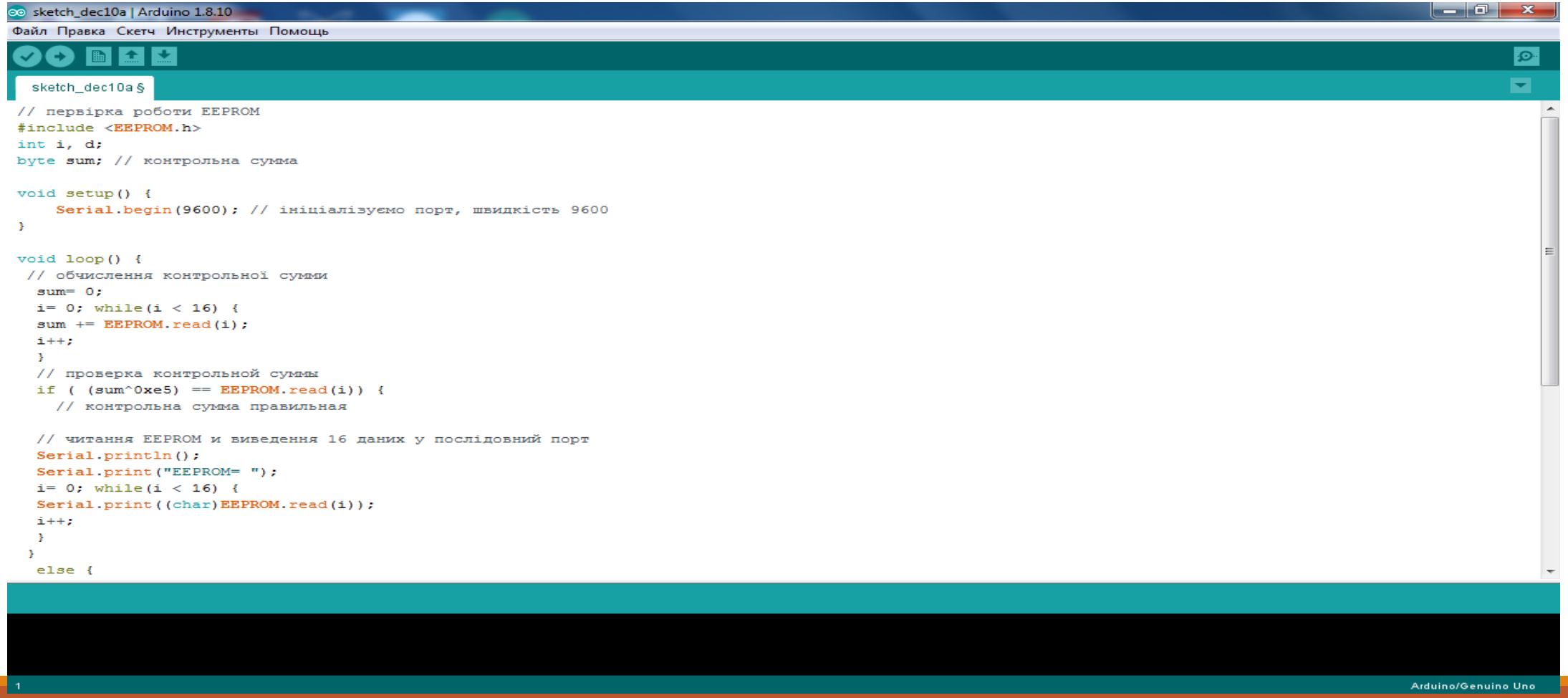
- у незалежній пам'яті (EEPROM, FLASH, HDD ...).
- при передачі даних (послідовні інтерфейси, WiFi, GSM, TCP/IP ...).
- в оперативній пам'яті для особливо важливих даних.
- деякі компоненти (наприклад, датчики), які мають протокол обміну даними з контролем цілісності.

Для контролю цілісності до блоку даних додається контрольний код. Цей код розраховується за певним алгоритмом при запису даних. При перевірці даних код перераховується і порівнюється з кодом, сформованим під час запису. Якщо коди не співпадають - дані помилкові.

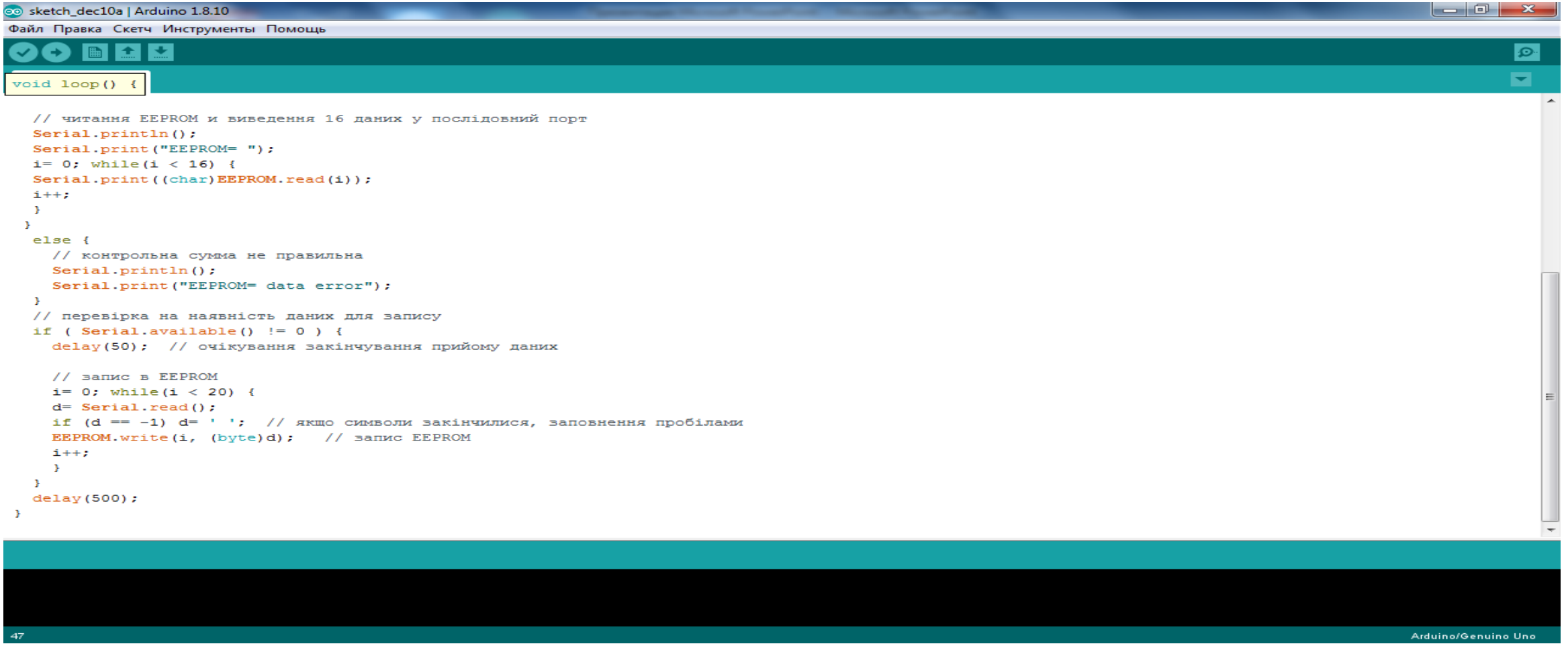
Алгоритм розрахунку контрольного коду визначає ймовірність визначення помилки даних. Існує велика кількість алгоритмів формування контрольних кодів: циклічні коди, різні функції хешування. Але найпростішим способом обчислення контрольного коду є контрольна сума.

Просто робиться математична сума всіх байтів блоку даних. Такий алгоритм вимагає мінімуму ресурсів для реалізації, і в той же час дозволяє з досить високою ймовірністю визначити помилкові дані в разі невеликих обсягів інформації. Контрольна сума може бути різною розрядності. У мінімальному варіанті це один байт. В цьому випадку відбувається переповнення результату суми, що зменшує ймовірність виявлення помилок.

Реалізація контролю цілісності даних



Реалізація контролю цілісності даних



Захист взаємодії ЕОМ і мікроконтролера

Передбачається, що пристрій який реалізуватиме певну систему захисту буде зібране в захисний корпус, на окремі блоки плати будуть нанесені ізоляційні матеріали, потрібно передбачити взаємодію ЕОМ і мікроконтролера для постійної підтримки системи, контролю системи, оновлення системи, оновлення безпеки тощо. Проводове з'єднання через послідовні порти, порти USB виключені. В такому випадку організацію взаємодії ЕОМ і мікроконтролера слід організовувати через бездротові інтерфейси, такі як Bluetooth і WiFi. Передача даних по Bluetooth має обмеження по відстані, при хороших умовах розташування відстань буде складати порядку 10 м. Тому доцільно, з точки зору дальності, використовувати обмін даними мікроконтролера і ЕОМ по WiFi. Такий обмін не потребуватиме окремого захисту передачі, так як буде можливість під'єднання елементів системи до існуючої мережі яка на рівні маршрутизатора може бути захищена. Проте для більшого захисту передачі окрім захисту мережі можна вжити додаткових заходів по захисту передачі. Гнучкість системи дозволяє корегувати дані аспекти захисту, оскільки вимоги до функціонування можуть бути різними. Ще однією суттєвою перевагою є апаратна частина яка реалізуватиме сполучення ЕОМ і мікроконтролера по бездротовій мережі. На сьогодні існують готові модулі які повністю сумісні з платформою Arduino і не потребують зусиль у поєднанні роботи модуля з периферійними пристроями які входитимуть у склад системи. Вартість одного такого модуля складає не більше 2$, що при використанні їх у складі системи не призведе до здорожчання системи.

Захист завантажувача (bootloader)

Завантажувач може самостійно перепрограмувати мікроконтролер. В разі використання завантажувача код програми залишається незахищеним. Завантажувач це та сама програма яку розробник може написати для тієї чи іншої системи.

Є можливість захистити вміст коду, якщо застосувати шифрування оновлюваної прошивки firmware.

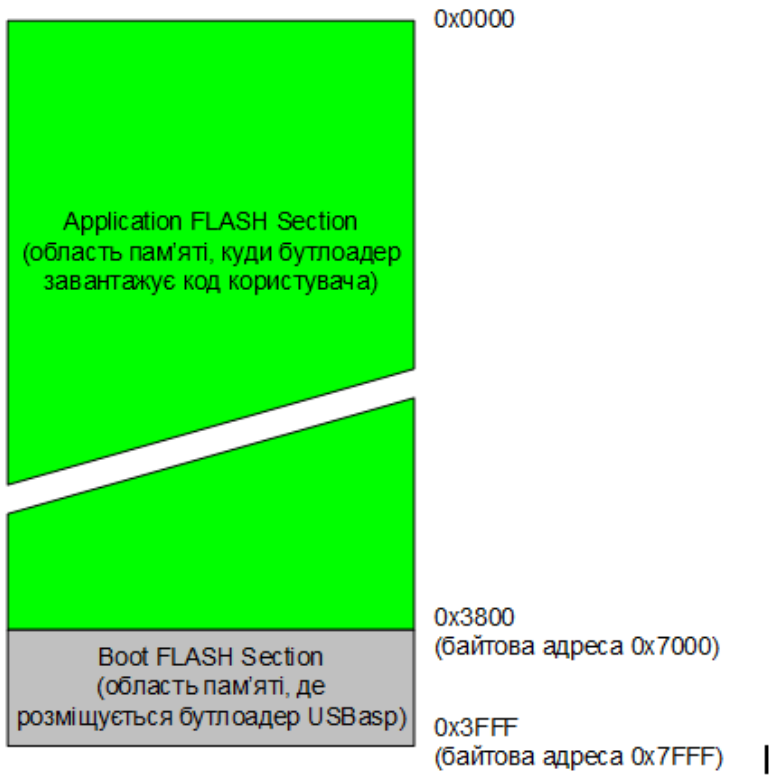
Захищений завантажувач для ATMega

Для мікроконтролерів AVR, у яких інтерфейс USB реалізований на бібліотеці V-USB, існує завантажувач (bootloader) USBasp, що емулює поведінку однойменного популярного програматора USBasp. Спочатку цей завантажувач приймає для прошивки незахищені HEX-файли, що іноді неприйнятно для захищеного оновлення програмного коду мікроконтролера (firmware).

Просте і надійне шифрування - накладання псевдовипадкової маски XOR (згенерований ключ, зазвичай псевдовипадкова послідовність електронних даних) на дані прошивки. Як відомо, логічна операція XOR це виклюне АБО, в нашому випадку виключне АБО між бітами даних прошивки і бітами даних псевдовипадкового ключа. Щоб розшифрувати прошивку, завантажувач при отриманні даних повинен накласти по XOR точно таку ж маску (такий же псевдовипадковий ключ, який накладався по XOR при шифруванні).

Після закінчення шифрування файл прошивки виходить захищеним - його можна вільно передавати по відкритих каналах зв'язку. Причому для неавторизованих користувачів він буде марний - його можна прошити тільки в ті цільові пристрої, в яких мікроконтролер має точно такий алгоритм генерації псевдовипадкової послідовності даних, яка була застосована при шифруванні прошивки.

Захищений завантажувач для ATMega



Для шифрування прошивки (firmware мікроконтролера, яке завантажує bootloader) використовується консольна утиліта XOR-firmware-cryptor.exe, яка на вході приймає 3 параметра командного - ім'я вхідного HEX-файлу прошивки, яка шифрується (опція -fi), ім'я вихідного HEX-файлу зашифрованою прошивки (опція -fo) і ключ шифрування (опція -seed).

Для того, щоб шифрування і дешифрування проходило успішно, на стороні шифрування і дешифрування повинен бути однаковий алгоритм генерування випадкових чисел, з однаковим значенням ключа для ініціалізації алгоритму LFSR. Ключ шифрування 1234 в цьому випадку використовується для ініціалізації регістра зсуву генератора псевдовипадкових чисел LFSR. Значення параметра для опції -seed може бути десятковим числом від 2 до 4294967295 (максимальне значення 2 певною мірою 32 мінус 1, так як регістр зсуву LFSR 32-розрядний). Якщо вказати для seed значення 0 або 1, то генератор LFSR працювати не буде (вся генерована послідовність чисел буде нульова: 0x00, 0x00, 0x00, ..). Це особливість використовуваного алгоритму LFSR.

Захищений завантажувач для мікроконтролера STM32

Кінцеве призначення і використання даного завантажувача ідентичне до реалізованого у попередньому пункті завантажувачі. Суттєвою відмінністю є мікроконтролер і платформа на якій це буде реалізовано, так як мікроконтролери STM32 мають інший інструментарій.

Архітектура STM32 передбачає плоску адресацію пам'яті, коли в одному адресному просторі знаходиться флеш-пам'ять, RAM, регістри периферії і все інше. Завантажувач - це програма, яка починає виконуватися під час запуску мікроконтролера, перевіряє, чи потрібно виконати оновлення прошивки, якщо потрібно - виконує його, і запускає основну програму пристрою. Механізм оновлення програмного коду розробляється у випадку оновлення через SD-карту, але можна використовувати будь-яке інше джерело, в тому числі, можна організувати оновлення по бездротовим каналам зв’язку. При цьому адаптування під такий спосіб оновлення і під інший тип мікроконтролера буде не складним.

Шифрування прошивки пропонується за допомогу алгоритму AES128. Реалізація даного типу шифрування програмного коду забезпечить необхідну стійкість і простоту реалізації.

Висновки

Дана магістерська кваліфікаційна робота була присвячена проблемі підвищення захисту від несанкціонованого доступу до програмних кодів мікроконтролерів систем безпеки.

Було розглянуто предметну область магістерської кваліфікаційної роботи. Визначено поняття мікроконтролера, його види і кола задач які даний пристрій може вирішувати. Також визначено роль мікроконтролерів у захисті інформації, в складі яких систем і пристроїв вони можуть бути. Таким чином було встановлено, що мікроконтролери або платформи основою яких вони є у складі пристроїв і систем безпеки потребують захисту від можливих атак з боку зловмисників, так як злам мікроконтролерів створює загрозу безпеці інформації.

Аналіз можливих загроз для мікроконтролерів показав основні нині відомі слабкі місця у використанні даних пристроїв. Розглянуті у першому розділі методи атак показують яким чином через мікроконтролер можливо зламати програмний код. На основі аналізу загроз мікроконтролерам було розглянуто методи захисту програмних кодів мікроконтролерів

Другий розділ магістерської кваліфікаційної роботи було присвячено було розробці методів та засобів захисту від несанкціонованого доступу до програмних кодів мікроконтролерів. Також було проаналізовано та обрано мікроконтролер ATMega 328 в складі платформи Arduino UNO R3 та мікроконтролер STM32 в складі платформи Discovery. Також в рамках виявлення векторів атак було проведено експериментальне дослідження можливості доступу до програмного коду мікроконтролера ATMega 328 на платі Arduino, яке показало яким чином може відбуватись копіювання коду мікроконтролера.

В третьому розділі магістерської кваліфікаційної роботи було представлено реалізації розроблених елементів захисту програмних кодів мікроконтролерів ATMega 328 і STM32 від несанкціонованого доступу. Наведено реалізацію конфігурування бітів захисту мікроконтролера ATMega 328 на платі Arduino. Для підвищення захищеності мікроконтролерів від несанкціонованого доступу було розроблено і реалізовано програму контролю цілісності даних, що зберігаються в EEPROM. Представлено реалізацію захищених завантажувачів (bootloader) для мікроконтролерів ATMega 328 і STM32. Для мікроконтролера ATMega 328 реалізовано завантажувач програмного коду в мікроконтролер з XOR – шифруванням, а для мікроконтролера STM32 було реалізовано завантажувач з AES шифруванням.

Дякую за увагу!