



Рисунок 3 – Кількість спожитої енергії за серпень.

Як видно з графіків протягом місяця зустрічаються періоди похмурої погоди коли сумарної кількості енергії виробленої сонячною панеллю недостатньо для забезпечення живлення всіх елементів станції. Цей факт варто враховувати при виборі ємності акумуляторів АГМЕС. Для реалізації даного проекту ємність акумулятора вибиралась з розрахунку забезпечення безперервного живлення станції протягом 5 діб за відсутності підзарядки від сонячної панелі.

Представлені графіки є лише одним з параметрів, які можна досліджувати за допомогою спроектованої станції. Перелік вимірюваних параметрів станцією складає понад 70 величин. По кожній вимірюваній величині можна задати час вимірювання та дозволити або заборонити відображення її значень в мережі інтернет.

Для ознайомлення та роботи з результатами вимірів, отриманих з гідрометеорологічних датчиків та датчиків режимів роботи станції АГМЕС було створено веб-сторінку: <https://theorems-dnipro.stu.cn.ua/data/>

#### Список використаних джерел

1. Сучасні автономні гідрометеорологічні вимірювальні станції : монографія / А.Л. Приступа, В.М. Безручко, О.А. Велігорський, А.С. Ревко, Ю.В. Кришньов. – Чернігів : видавець Брагинець О.В., 2019. – 180 с.
2. Дрозд М. С., Бобко Є.О., Приступа А. Л. Статистична оцінка параметрів блоку живлення автономної станції моніторингу // Новітні технології сучасного суспільства (НТСС-2019) : науково-практична конференція (м. Чернігів, 12 грудня 2019 р.) : тези доповідей. – Чернігів : ЧНТУ, 2019. – с. 129 – 131.

УДК 681

## ЗАСТОСУВАННЯ ЦИКЛІЧНИХ КОДІВ ДЛЯ ЗАХИСТУ ІНФОРМАЦІЇ

Браславська В.О., студ. гр. ВТ-161

Науковий керівник: Степенко С.А., к.т.н.

Національний університет «Чернігівська політехніка»

В якості утворюючого поліному у циклічних кодах вибирають неприводимі поліноми – котрі не можна розкласти на два і більше співмножників – аналоги простих чисел.

Циклічні коди мають особливість: якщо знаємо одну комбінацію, що входить в дану циклічну послідовність, то під час зсуву її на один чи більше розрядів, знову отримуємо код, що входить в дану послідовність. Тобто ділення даного поліному на утворюючий буде без залишку.

Циклічні коди дозволяють знаходити і коректувати помилки не тільки одинарної, але й більш високих кратностей. Коректуюча можливість коду залежить від обраного утворюючого поліному:

1)  $2^k \geq n+k+1$ , де:  $k$  – ступінь утворюючого поліному;  $n$  – кількість розрядів початкового коду.

2) Для отримання коду з заданими коректуючими можливостями потрібно, щоб кількість одиниць в утворюючому поліномі було  $\geq d$  – мінімальної кодової відстані.

Циклічні послідовності можна отримати двома способами:

Множенням утворюючого поліному на початковий. В цьому випадку отримаємо код, в якому неможна виділити інформаційні і перевірочні розряди.

Грунтується на зсуві початкового коду і діленням отриманого поліному на утворюючий з наступним додаванням до отриманої, після зсуву, послідовності залишку від ділення. В такому випадку інформаційні і перевірочні розряди будуть розділені (CRC – коди – Cyclic Redundancy Check – Коди з Надлишковим Котролем)

Приклади  $A=10001000$ ,  $n=8$ , отримаємо циклічну послідовність, що дозволяє знаходити і виправляти одинарні помилки ( $d=3$ ):  $2^k \geq 8+k+1$ , отже,  $k \geq 4$ . Неприводимі поліноми для  $k=4$ :  $x^4+x+1$ ;  $x^4+x^3+1$ ;  $x^4+x^3+x^2+x+1$ . Виберемо  $G(x)=x^4+x+1$ ; Оскільки процес виправлення помилок громіздкий і за великої довжини даних його не можна швидко виконати, то коди будують тільки для виявлення помилок.

На практиці для захисту інформації використовують **16-розрядний** код CRC, який додається або до файлу, або до блоку інформації. В якості утворюючого поліному використовують стандартні поліноми CRC16:  $x^{16}+x^{12}+x^5+1 = 1\ 0001\ 0000\ 0010\ 0001 = 0x1021$ ;  $x^{16}+x^{15}+x^2+1 = 1\ 1000\ 0000\ 0000\ 0010 = 0x8005$ ; і т.д.

Під час приймання інформації або її зчитування (з диску, наприклад) йде перевірка. Тобто вихідний файл ділиться по правилу побудування циклічного коду і залишок порівнюється з CRC-кодом, якщо співпадає – помилки нема.

#### Список використаних джерел

1. [https://uk.wikipedia.org/wiki/%D0%A6%D0%B8%D0%BA%D0%BB%D1%96%D1%87%D0%BD%D0%B9\\_%D0%BA%D0%BE%D0%B4](https://uk.wikipedia.org/wiki/%D0%A6%D0%B8%D0%BA%D0%BB%D1%96%D1%87%D0%BD%D0%B9_%D0%BA%D0%BE%D0%B4)
2. [http://antibotan.com/file.html?work\\_id=400220&&event=preview](http://antibotan.com/file.html?work_id=400220&&event=preview)
3. <https://studall.org/all-33287.html>
4. <http://ea.donntu.org:8080/jspui/bitstream/123456789/25462/1/%D0%A2%D0%B5%D0%BE%D1%80%D0%B8%D1%8F%20%D0%BA%D0%BE%D1%80%D1%80%D0%B5%D0%BA%D1%82%D0%B8%D1%80%D1%83%D1%8E%D1%89%D0%B8%D1%85%20%D0%BA%D0%BE%D0%B4%D0%BE%D0%B2.%20%D0%9B%D0%B5%D0%BA%D1%86%D0%B8%D0%B8.%20UA.pdf>

УДК 681.518

## ОСОБЛИВОСТІ ПОБУДОВИ КОДІВ ГРЕЯ У ІНФОРМАЦІЙНО-ВИМІРЮВАЛЬНИХ СИСТЕМАХ

**Віротченко А. А.**, студ. гр. ВТ-161

Науковий керівник: **Степенко С. А.**, к.т.н.

*Національний університет «Чернігівська політехніка»*

Як відомо, для кодування ознак можна скористатись найпростішим варіантом: двійковим значенням цієї ознаки. Тоді легко використовувати бітові рядки фіксованої довжини для подання всіх можливих значень цієї ознаки. Наприклад, десяткові числа 7 і 8 можна легко закодувати у двійкові числа  $V(7)=011$  і  $V(8)=100$ , використовуючи двійкову техніку. Проте, якщо ми хочемо переміститися з фенотипу 7 у фенотип 8, то повинні змінити всі три біти в їх зображенні від  $V(7)=011$  до  $V(8)=100$ .

Щоб уникнути цього, краще використовувати кодування, в якому значення розрізняються на один біт. Таким є код Грея. Розглянемо принцип його побудови: десятковий Код Грея (0 0000, 1 0001). Мета: визначити особливості побудови кодів Грея. Дослідження: кодом Грея