

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ СИСТЕМИ УПРАВЛІННЯ ЕНЕРГОРЕСУРСАМИ ІВАНО-ФРАНКІВСЬКОГО НАЦІОНАЛЬНОГО ТЕХНІЧНОГО УНІВЕРСИТЕТУ НАФТИ І ГАЗУ

В.Б.Кропивницька, Т.Г.Гарасимів, С.М.Гаврилко

*ІФНТУНГ, 76019, м. Івано-Франківськ, вул. Карпатська, 15, тел. (0342) 504521
e-mail: ksm@ifdtung.if.ua*

Розглядається питання розробки програмного забезпечення комп'ютерної системи обліку енергоносіїв в умовах ІФНТУНГ, яке призначається для ведення обліку показників лічильників, можливості перевірки витрат енергії в режимі реального часу.

Рассматривается вопрос разработки программного обеспечения компьютерной системы учета энергоносителей в условиях ИФНТУНГ, которое предназначается для ведения учета показателей счетчиков, возможности проверки расходов энергии в режиме реального времени.

In the article the question of computer system of account of power mediums software development is examined in the conditions of IFNTUNG, which targets at the conduct of account of indexes of meters, possibility of real-time verification of charges of energy.

Актуальність роботи. Ми живемо в час, коли витрати енергоресурсів все зростають, а кількість самих ресурсів неминуче зменшується. Адже з сучасними темпами витрат енергії людство вже за століття може залишитись без багатьох видів енергоресурсів. Тому однією з найбільш ефективних і досяжних стратегій для пониження експлуатаційних витрат є скорочення корисних витрат. Реальна енергоефективна економіка вимагає створення сучасних механізмів контролю та управління споживанням енергії.

В роботах [1, 2] показано, що витрати енергоресурсів, в більшості випадків, є нерациональними і необдуманими. Тому для рационального використання енергоресурсів доцільно застосовувати комп'ютерні системи моніторингу за їх споживанням, які б аналізували і вказували на так звані «слабкі місця» в енергоспоживанні підприємства чи організації. Тому на сьогоднішній день по цілому світу багато великих споживачів енергії укладають контракти з постачальниками енергетичних послуг та послуг розробки програмного забезпечення з метою створення проектів моніторингу, аналізу і економії енергії. Це передбачає оцінку того, як енергія використовується в комерційних цілях, встановлення точної базової інформації і виділення місць, де енергоспоживання може бути зменшене.

Метою даної роботи є розроблення програмного забезпечення для комп'ютерної системи обліку енергоносіїв в умовах ІФНТУНГ. Система моніторингу споживання енергоресурсів включає в себе велику кількість компонентів: лічильники електроенергії, газу, води та теплової енергії; GSM модеми, сервери і робочі станції.

Автоматизована система передбачає обмін інформацією між лічильниками та сервером з використанням GSM модемів шляхом

надсилання і одержання SMS повідомлень. На низькому рівні процес передавання інформації через GSM модем за допомогою повідомлень можна реалізувати з використанням AT команд, оскільки їх можна використовувати у випадку, коли немає жодного доступного інструменту для зв'язку з модемом.

У розробленій системі моніторингу споживання енергоресурсів використовується модем Teltonika T-ModemUSB, що є GSM модемом з портом USB2.0. Основою цього модему є Nokia 12i GSM модуль з підтримкою сучасної швидкісної технології EDGE. Для використання AT команд для надсилання і отримання SMS повідомлень через GSM модем виконують такі кроки:

1. Встановлення GSM модему.
2. Використання терміналу під назвою «HyperTerminal».
3. Початкове встановлення AT команд.
4. Надсилання SMS з використанням AT команд.
5. Отримання SMS з використанням AT команд.
6. Використання комп'ютерної програми для надсилання і отримання SMS повідомлень.

Оскільки більшість GSM модемів постачаються на ринок з простою документацією і відповідними драйверами, то їх встановлення не виникає труднощів. Контрольна панель Windows аплікації, що відповідає за встановлення GSM модему, встановлення GPRS з'єднання, а також надсилання і отримання SMS повідомлень, має вигляд, зображений на рис. 1.

Під час розроблення AT команд з використанням *HyperTerminal* полегшується написання коду з використанням технологій і мов вищого рівня, наприклад, VB, C++, C#, Java тощо.

Для запускання терміналу *HyperTerminal* і створення нового з'єднання необхідно скористатись командою «START(Programs)»

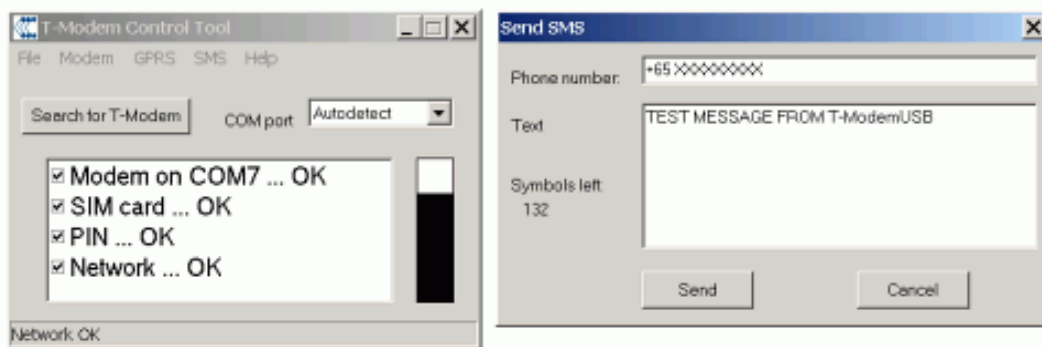


Рисунок 1 – Контрольна панель

Accessories\Communications\HyperTerminal» (для операційної системи Windows XP) з такими можливими налаштуваннями:

- COM Port – так як вказано в контрольній панелі T-modem або в документації;
- Bits per second – 230400 (або повільніше);
- Data Bits – 8;
- Parity – None;
- Stop Bits – 1;
- Flow Control – Hardware.

Для початкового встановлення AT команд необхідно встановити і перевірити статус GSM модему, виконуючи наступні команди:

AT – команда повертає "OK" як підтвердження того, що модем працює коректно;

AT+CPIN="xxxx" – команда для введення PIN-коду для нашої SIM-карти (якщо код доступний);

AT+CREG? – відповідь «0,1» підтверджує, що наш модем під'єднаний до GSM мережі;

AT+CSQ – команда вказує на силу сигналу, максимальним значенням є 31.99.

Організація надсилання SMS повідомлень через контрольну панель визначається такими командами:

AT+CMGF=1 – команда для форматування SMS повідомлення як TEXT повідомлення;

AT+CSCA="+xxxxx" – команда для встановлення номеру центру провайдера.

Для надсилання SMS повідомлень використовують AT команду AT+CMGS:

AT+CMGS="+ууууу" <Enter>
> Тут має бути текст SMS повідомлення <Ctrl-Z>

Параметр "+ууууу" – це є мобільний номер одержувача SMS повідомлення.

Отримання SMS повідомлення з використанням AT команд з GSM модему можливе двома способами.

1. Негайно – як тільки SMS повідомлення отримане, деталі цього повідомлення негайно відправляються на головний комп'ютер (DTE) з використанням AT команди +CMT:

AT+CMGF=1 – команда, щоб сформувати SMS повідомлення як TEXT повідомлення;

AT+CNMI=1,2,0,0,0 – команда для встановлення відповіді модема на отримане SMS повідомлення.

Коли GSM модем отримує нове SMS повідомлення, то DTE отримує таку інформацію:

+CMT: "+61xxxxxxx", ,
"04/08/30,23:20:00+40"

Це – текстове SMS повідомлення, що було надіслане на модем.

Головний комп'ютер DTE промоніторить COM порт та прочитає і проаналізує (розбере по частинах) отримане повідомлення.

2. Інформативно – коли SMS повідомлення буде отримане, головний комп'ютер DTE буде про це проінформований. Потім комп'ютер зчитає повідомлення з визначеної частини пам'яті і очистить цю частину пам'яті після прочитання:

AT+CMGF=1 – команда, щоб сформувати SMS повідомлення як TEXT повідомлення;

AT+CNMI=1,1,0,0,0 – команда для встановлення відповіді модему на SMS повідомлення.

Коли GSM модем отримує нове SMS повідомлення, то DTE отримує наступну інформацію:

+CMTI: "SM",3 – повідомлення, відправлене комп'ютеру. Частина #3 в пам'яті SIM-карти;

AT+CMGR=3 <Enter> – AT команда для зчитування одержаного SMS повідомлення з модему.

Потім модем відправляє комп'ютеру деталі отриманого повідомлення з визначеної частини пам'яті:

+CMGR: "REC READ", "+61xxxxx", ,
"04/08/28,22:26:29+40"

Після прочитання і граматичного аналізу нового повідомлення, головний комп'ютер DTE повинен надіслати на модем AT команду для очищення частини пам'яті, яка була зайнята повідомленням та інформацією про нього:

AT+CMGD=3 <Enter> – команда для очищення частини пам'яті, яка була зайнята повідомленням та інформацією про нього.

Якщо комп'ютер буде намагатись прочитати порожню/очищену частину пам'яті, то йому буде надіслана наступна помилка – +CMS ERROR : 321.

Для надсилання AT ASCII команд до COM порту, з яким є зв'язаний наш GSM модем, і для зчитування повідомлень з цього порту використовуємо розроблену комп'ютерну програму для надсилання і отримання SMS повідомлень.

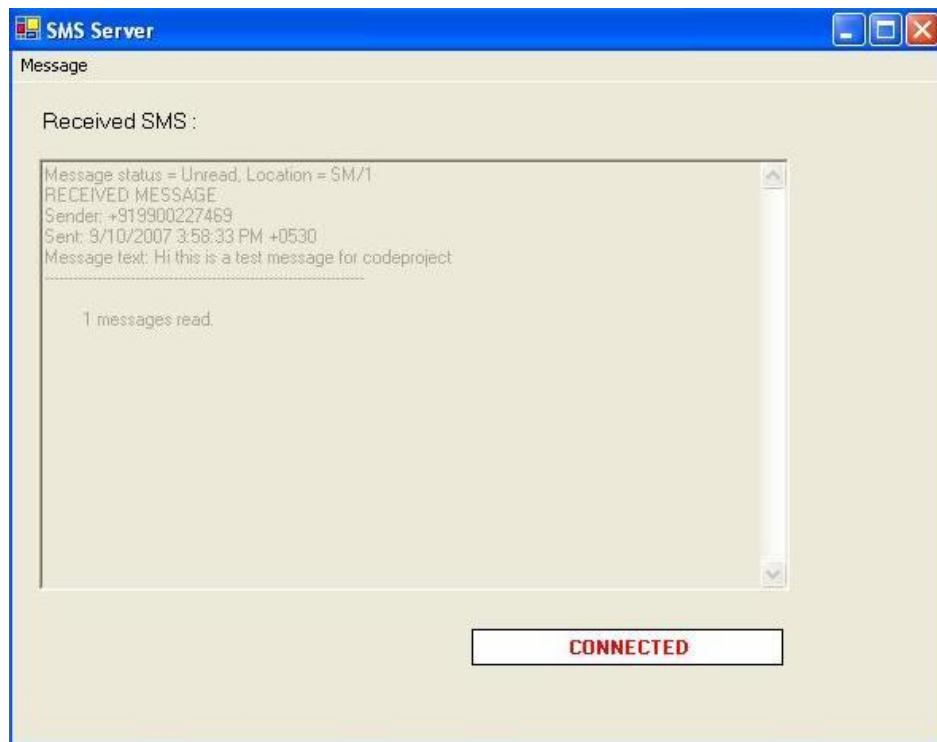


Рисунок 2 – Вікно серверної частини

Для написання програми для обслуговування GSM модемів застосовується технологія C#. SMS клієнт-серверна програма призначена для надсилання, читання і видалення повідомлень (рис. 2). Програма використовує GSM модем для надсилання і одержання повідомлень. Програма очікує (прослуховує порт) на повідомлення, що поступають на модем, зчитує і обробляє повідомлення, що прибули та виконує відповідні дії. Дана SMS програма використовує бібліотеку для мови програмування C# – GSMComm Library.

SMS клієнт і сервер є аплікацією, що використовується для надсилання і отримання коротких повідомлень (SMS). Програма очікує (прослуховує порт) на вхідні повідомлення і після прибуття повідомлення обробляє його і зчитує, якщо воно прибуло у правильному форматі.

Для коректної роботи програми проводиться налаштування комунікаційного порту за допомогою класу «CommSetting».

«Comm» – це об'єкт типу «GsmCommMain», який потрібен для надсилання і отримання повідомлень. У вікні налаштування з'єднань встановлюються порт, номінальну швидкість і час затримки для нашого об'єкта «Comm». За необхідності можна відкрити аплікацію з попередніми установками, клікнувши кнопкою «Test» (рис. 3). Якщо COM-порт є недоступним для відкриття, отримується повідомлення «No GSM modem connected», це означає, що жоден модем не підключений до порту. Це пов'язано з установками номінальної швидкості. Вирішення цієї проблеми полягає в зміні значення в полі номінальної швидкості і клікання кнопкою «Test» доти, доки отримується наступне повід-

омлення – «Successfully connected to GSM modem», що означає, що модем було успішно підключено (рис. 4).

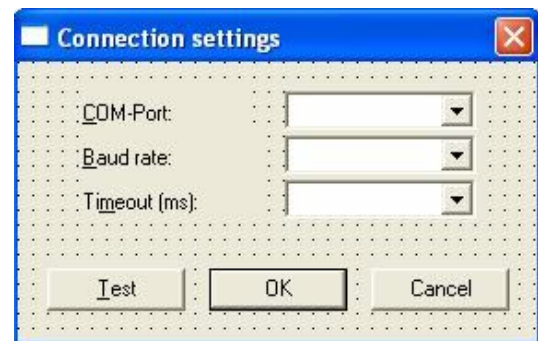


Рисунок 3 – Вікно налаштувань з'єднань



Рисунок 4 – Успішне підключення модему

Перед створенням об'єкта з налаштуваннями типу «GSMComm» потрібно перевірити номер порту, номінальну швидкість і час затримки. Метод «EnterNewSettings()» виконує цю перевірку і повертає значення «true», якщо вона була успішною, та викликає метод

«SetData(port,baud,timeout)» для налаштування з'єднання.

Наступний блок коду призначається для створення з'єднання, і якщо виникають будь-які проблеми, видається повідомлення «No GSM modem connected». При цьому наступними діями може бути повторне клікання кнопкою «Retry» або відміна спроби створення з'єднання шляхом клікання кнопкою «Cancel».

Для отримання вхідного повідомлення необхідно зареєструвати наступні події для об'єкта «comm» типу «GSMComm»:

«PhoneConnected» – ця подія виникає тоді, коли є спроба створити (відкрити) зв'язок з портом комунікації. Обробником події для під'єданого модему є метод «comm_PhoneConnected», який викликає метод «OnPhoneConnectionChange(bool connected)» з допомогою «ConnectedHandler» – класу, що делегує події.

«MessageReceived» – ця подія виникає тоді, коли повідомлення прибуває на GSM модем. Коли вхідне повідомлення прибуває, буде викликаний метод «comm_MessageReceived», який, у свою чергу, викличе метод «MessageReceived()» для того, щоб обробити (розбити на частини) вхідне непрочитане повідомлення. Об'єкт «comm» типу «GSMComm» має метод «ReadMessages», який може бути використаний для читання повідомлень. Цей метод приймає два параметри від модему – статус (All – всі, ReceivedRead – одержані і прочитані, ReceivedUnread – одержані і непрочитані, StoredSent – збережені і надіслані та StoredUnsent – збережені і ненадіслані) та тип збереження – зберегти у пам'ять SIM-карти або зберегти у пам'ять GSM модему.

Наведений вище код зчитує всі непрочитані повідомлення з пам'яті SIM-карти. Метод «ShowMessage» використовується для відображення прочитаного повідомлення, яке може бути повідомленням статусу, збереженням повідомленням, надісланим/не надісланим повідомленням або одержаним повідомленням.

Надсилання повідомлень. З використанням даної аплікації ми можемо посилати SMS повідомлення, попередньо ввівши номер адресата, а також сам текст повідомлення (рис. 5).

Для відправлення надіслати повідомлення на будь-якій мові світу вище наведеному вікні потрібно вибрати опцію «Send as Unicode (UCS2)».

Об'єкт «GSMComm» має метод «SendMessage», який може бути використаний для відправлення SMS на будь-який номер. Для цього попередньо створений об'єкт PDU для надсилання повідомлень, який створюється двома способами: в спрощеній і розширеній (з використанням Unicode) версіях.

Для читання всіх повідомлень з пам'яті модему або з пам'яті SIM-карти потрібно клікнути кнопкою «Read All Messages» (рис. 6).

В даному вікні відображаються такі деталі: відправник (sender), дата-час (date-time), текст повідомлення (text message) і таблиця, кожен рядок якої є інформацією про прочитане повідомлення.

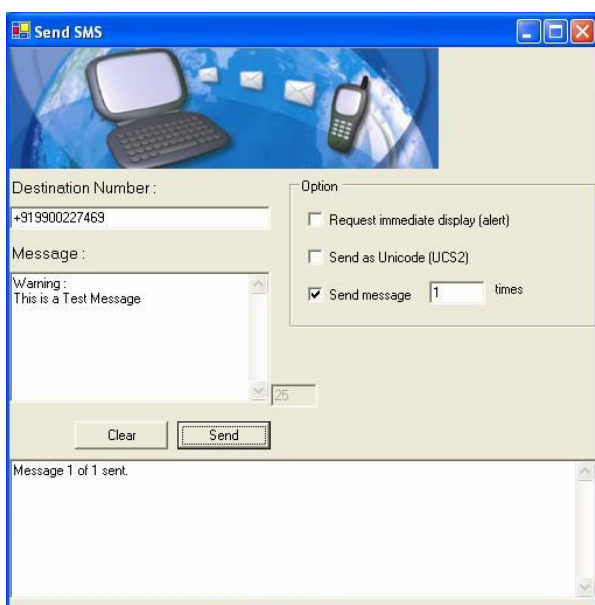


Рисунок 5 – Вікно для надсилання повідомлень

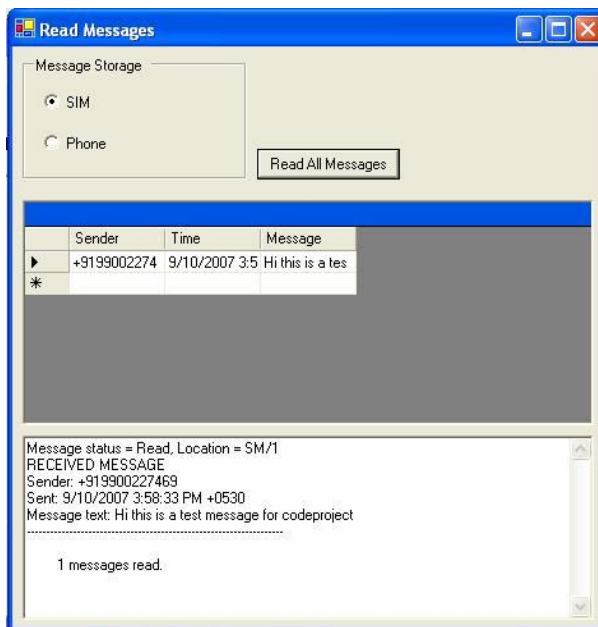


Рисунок 6 – Читання всіх повідомлень

Всі повідомлення, що були надіслані користувачами зберігаються в пам'яті SIM-карти і відображаються в окремій таблиці у вікні аплікації. Для видалення повідомлень з пам'яті SIM-карти, клікнувши кнопкою «Delete All» (рис. 7). Оскільки кожне повідомлення зберігається в пам'яті з унікальним індексом, повідомлення видаляються, базуючись на цьому принципі.

Висновки. Застосування об'єктно-орієнтованого підходу для розробки програмного забезпечення обслуговування GSM модемів дало змогу створити наглядний і функціональний інтерфейс призначений для надсилання і одержання SMS повідомлень між модемом та сервером, що дає змогу здійснювати облік енергоресурсів.

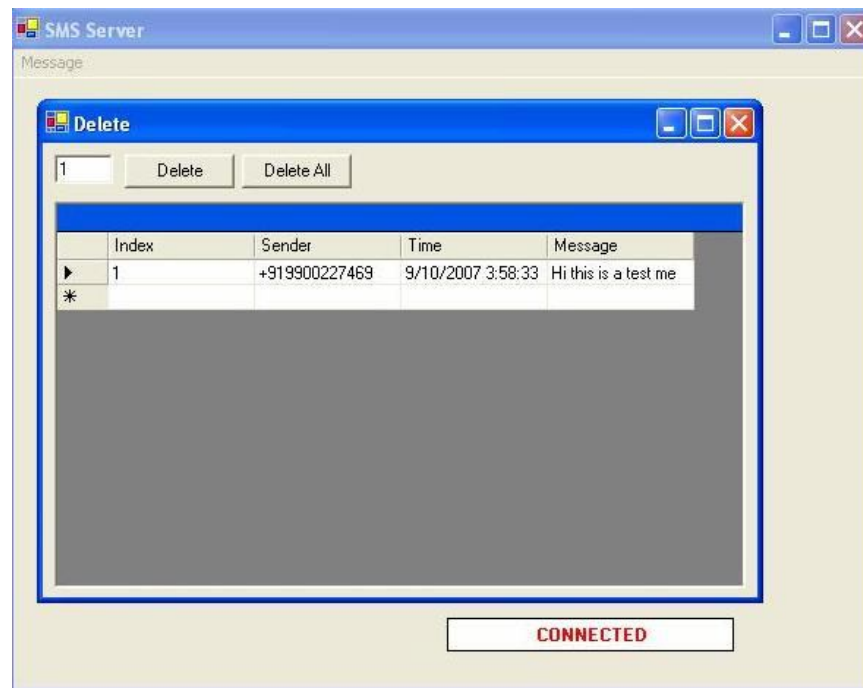


Рисунок 7 – Видалення повідомлень

Література

1 Dorf R. «Systems, controls, embedded systems, energy, and machines.» [PDF] – CRC Press, 2006. – 672 с.

2 Capenhart B. «Web based energy information and control systems: case studies and applications.» [PDF] / B. Capenhart, L. Capenhart – The Fairmont Press, 2005. – 555 с.

3 Using AT commands to Send and Receive SMS [Електронний ресурс] / Режим доступу : http://www.control.com.sg/at_commands_sms.aspx

4 Лафоре Р. «Объектно-ориентированное программирование» [Текст] – СПб: Питер, 2004. – 923 с.

5 Юань Ф. «Программирование графики для Windows.» [Текст] – СПб.: Питер, 2002. – 1072 с.

Стаття постуила в редакційну колегію

22.05.09

Рекомендована до друку професором
Горбійчуком М.І.