

ДОДАТОК А  
Графічний матеріал атестаційної роботи

Спеціалізовані комп'ютерні засоби для  
діагностування  
стану річкової води й їх модульна різноманітність

Спікер – Федосеєнко В.Ю.

Студент 6 курсу, групи СКСм-20-2



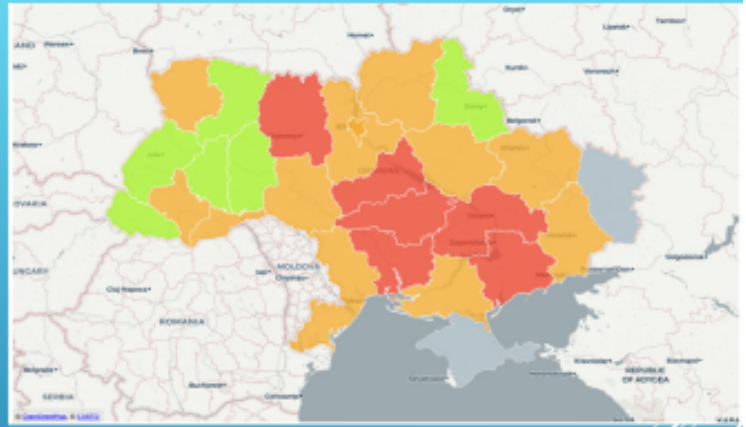
ВОДА Є ОСНОВНИМ ПРИРОДНИМ РЕСУРСОМ ЖИТТЯ ДЛЯ КОЖНОГО ЖИВОГО ОРГАНІЗМУ НА ЗЕМЛІ, ЩО РОБИТЬ ЇЇ ОДНИМ ІЗ ГОЛОВНИХ СКАРБІВ НАШОЇ ПЛАНЕТИ.



У ТЕПЕРІШНЬОМУ ЧАСІ ПРОВОДЯТЬСЯ ШИРОКІ ДОСЛІДЖЕННЯ, ГЛОБАЛІЗАЦІЯ, ІНДУСТРІАЛІЗАЦІЯ ТА НОВІ ТЕХНОЛОГІЧНІ ВІДКРИТТЯ, АЛЕ ІСНУЮТЬ ТАКОЖ ПОТЕНЦІЙНІ ЕКОЛОГІЧНІ ЗАГРОЗИ, ЩО СЛІДУЮТЬ ЗА НИМИ, ТАКІ ЯК ЗАБРУДНЕННЯ НАВКОЛИШНЬОГО СЕРЕДОВИЩА, ГЛОБАЛЬНЕ ПОТЕПЛІННЯ, ЩО ПРИЗВОДИТЬ ДО ШКОДИ ПРИРОДІ ТА ЇЇ МЕШКАНЦЯМ.



ТИПОВА ВОДОРозПОДІЛЬНА МЕРЕЖА СКЛАДАЄТЬСЯ З ДЖЕРЕЛ, ТРУБ ТА ПІДРАВЛНИХ ЕЛЕМЕНТІВ, З'ЄДНАНИХ РАЗОМ ДЛЯ ПОДАЧІ ЗАПРОПОНОВАНОГО КІЛЬКОСТІ ВОДИ ПРИ БАЖАНОМУ ТИСКУ І ЯКОСТІ В РІЗНИХ ТОЧКАХ ПОПИТУ. АЛЕ ЯК ВЖЕ БУЛО ЗАЗНАЧЕНО ТАКА СИСТЕМА В БАГАТЬОХ ВИПАДКАХ НЕ ПЕРЕДБАЧАЄ НАВІТЬ НАЙПРИМІТИВНІШІ МЕТОДИ ЗАХИСТУ ПИТНОЇ ВОДИ. . МІСІЯ ПО ЗАХИСТУ ЗАЗВИЧАЙ ВПАДАЄ НА ПЛЕЧІ ВОДНИХ ІНСТАНЦІЙ І ТИХ ТЕХНОЛОГІЙ, ЯКИМИ ВОНИ ВОЛОДІЮТЬ.



ОДНІЄЮ З КЛЮЧОВИХ ЗАДАЧ ПРИ СТВОРЕННІ СИСТЕМ ЕКОЛОГІЧНОГО МОНІТОРИНГУ ВОДНОГО СЕРЕДОВИЩА Є ПРОБЛЕМА ЗАБЕЗПЕЧЕННЯ ОПЕРАТИВНОСТІ РОБОТИ СИСТЕМИ МОНІТОРИНГУ.

ВАЖЛИВІСТЬ ЦЬОЇ ПРОБЛЕМИ ОЧЕВИДНА, ОСКІЛЬКИ ДУЖЕ ПІЗНЄ НАДХОДЖЕННЯ ІНФОРМАЦІЇ ПРО СТАН ВОДНОГО СЕРЕДОВИЩА НЕ ДАСТЬ ЗМОГИ ОРГАНІЗУВАТИ ЗАХИСТ.

ОСНОВНИМ СПОСОБОМ РОЗВ'ЯЗАННЯ ЦЬОЇ ЗАДАЧІ Є СТВОРЕННЯ СИСТЕМ ЕКОЛОГІЧНОГО МОНІТОРИНГУ ВОДНОГО СЕРЕДОВИЩА НА ОСНОВІ КОМП'ЮТЕРНИХ ТЕХНОЛОГІЙ ОПРАЦЮВАННЯ ДАНИХ У РЕАЛЬНОМУ ЧАСІ І ЗАСТОСУВАННЯ НАЙДОСКОНАЛІШИХ СИСТЕМ ПЕРЕДАВАННЯ ДАНИХ.

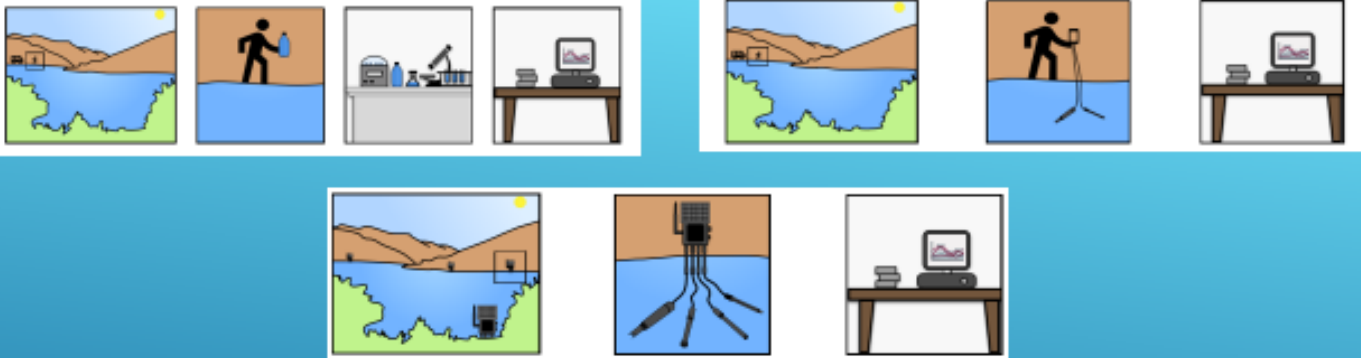
### Індекси якості води

- Для кількісного оцінювання якості водним застосовують індекси якості води, які пропонують визначати такі параметри:
  - розчинений кисень,
  - фекальні речовини,
  - рН,
  - біохімічну потребу у кисні,
  - температуру,
  - загальні фосфати,
  - нітрати,
  - каламутність,
  - тверді речовини.
- Використовується така система балів оцінювання якості:
  - 90-100 – відмінна;
  - 70-90 – добра;
  - 50-70 – середня;
  - 25-50 – погана;
  - 0-25 – дуже погана.



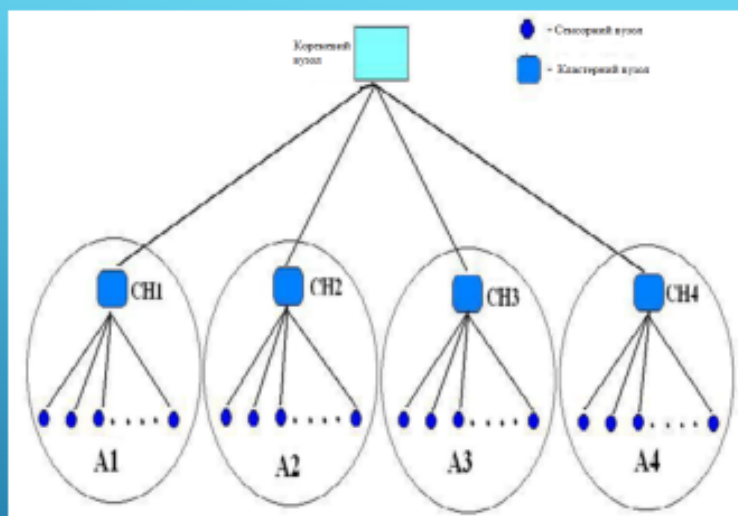
БУЛО ПРОВЕДЕНО БАГАТО ДОСЛІДЖЕНЬ, СПРЯМОВАНИХ НА ВИЗНАЧЕННЯ РІЗНИХ ПАРАМЕТРІВ, ЯКІ СЛІД ВИМІРЯТИ ДЛЯ ВИЗНАЧЕННЯ ЯКОСТІ ВОДИ, А ТАКОЖ ДАТЧИКІВ, ЯКІ МОЖУТЬ ВИМІРЮВАТИ ЦІ ПАРАМЕТРИ.

ТЕХНОЛОГІЧНИЙ ПРОГРЕС ДОПОМІГ УПРАЗДНИТИ ПРОЦЕС ЗБОРУ ТА АНАЛІЗУ ДАНИХ З РІЗНИХ ДЖЕРЕЛ, ПРИРОДНИХ ТОЩО.



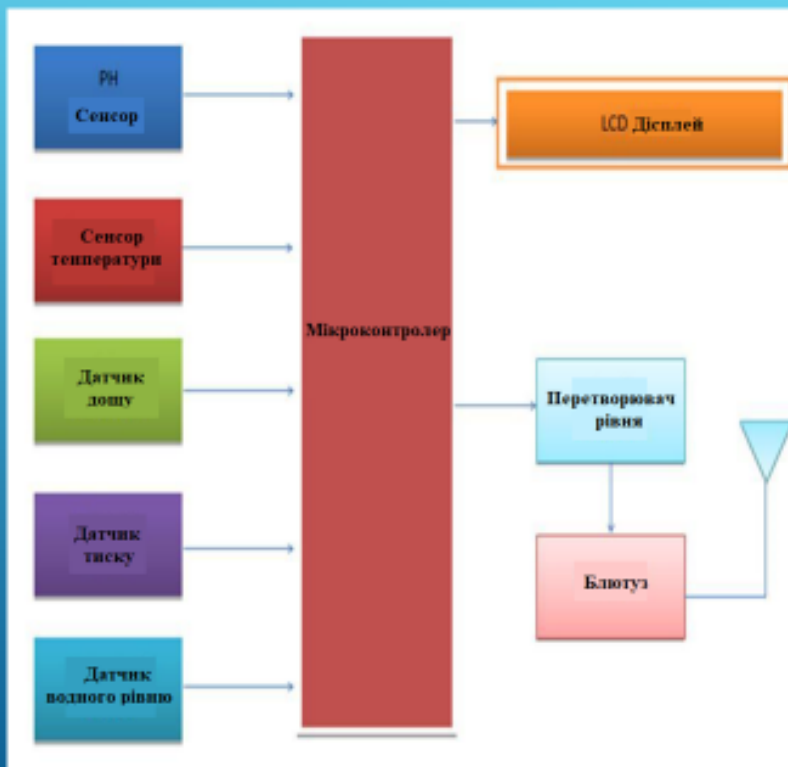
WSN (WIRELESS SENSOR NETWORK/БЕЗДРОТОВА СЕНСОРНА МЕРЕЖА) - РОЗШИРЮЮТЬ МОЖЛИВОСТІ СИСТЕМ МОНІТОРИНГУ НА МІСЦІ. ХОЧА ТРАДИЦІЙНІ СИСТЕМИ IN-SITU ДОЗВОЛЯЮТЬ АНАЛІЗУВАТИ НА МІСЦІ, ВОНИ ВИМАГАЮТЬ, ЩОБ ЗІБРАНІ ДАНІ ТРАНСПОРТУВАЛИСЯ ВРУЧНУ ДО ВІДАЛЕНИХ ОФІСІВ АБО ЦЕНТРІВ УПРАВЛІННЯ ДЛЯ ПОДАЛЬШОГО АНАЛІЗУ ТА ДІЙ.

WSN ДОЗВОЛЯЮТЬ АВТОМАТИЧНО ПЕРЕДАВАТИ ЦІ ДАНІ, А ТАКОЖ ЗАБЕЗПЕЧУЮТЬ МЕХАНІЗМ ЗВОРОТНОГО ЗВ'ЯЗКУ В ДЕЯКИХ ВИПАДКАХ, ЩОБ УТОЧНИТИ ДЕТАЛЬНІСТЬ ЗБОРУ ДАНИХ.



ТАКИМ ЧИНОМ, КОЖНА ЗОНА МОНІТОРИНГУ УТВОРЮЄ КЛАСТЕР, ЩО СКЛАДАЄТЬСЯ З КІЛЬКОХ БЕЗДРОТОВИХ СЕНСОРНИХ ВУЗЛІВ, ВІДПОВІДАЛЬНИХ ЗА ЗОНДУВАННЯ, ЗБІР І ОБРОБКУ ДАНИХ І ЗВ'ЯЗОК.

КОЖЕН СЕНСОРНИЙ ВУЗЕЛ МОЖЕ МІСТИТИ ДАТЧИКИ ДЛЯ МОНІТОРИНГУ ОСНОВНИХ ПАРАМЕТРІВ ЯКОСТІ ВОДИ, ТАКИХ ЯК РН, ПРОВІДНІСТЬ, ТЕМПЕРАТУРА, РОЗЧИНЕНИЙ КИСЕНЬ І КАХАМУТНІСТЬ.



СИСТЕМА ВКЛЮЧАЄ В СЕБЕ МОНІТОРИНГ ЯКОСТІ ВОДИ АКЦЕНТУЮЧИ УВАГУ НА АСПЕКТАХ НИЗЬКОЇ ВАРТОСТІ, ПРОСТОЇ СПЕЦІАЛЬНОЇ УСТАНОВКИ ТА ПРОСТОГО ВИКОРИСТАННЯ ТА ОБСЛУГОВУВАННЯ.

ВІН ДАЄ АВТОМАТИЧНЕ СПОВІЩЕННЯ ПРО ЗМІНУ ЯКОСТІ ВОДИ. ВІН МАЄ ДОДАТКОВУ ФУНКЦІЮ, ЩОБ ПЕРЕВІРИТИ ПОКАЗАННЯ ПАРАМЕТРІВ ВОДИ ТА ПЕРЕВІРИТИ ПОГОДУ, ЧИ ВОДА ПРИДАТНА ДЛЯ СПОЖИВАННЯ, БЕЗ НАЯВНОСТІ ЕКСПЕРТНИХ ЗНАТЬ.

ЗАПРОПОНОВАНА СИСТЕМА МОЖЕ БУТИ ЛЕГКО ВСТАНОВЛЕНА В РІЗНИХ СФЕРАХ, ТАКИХ ЯК ВОДНІ РЕСУРСИ СУСПІЛЬСТВА, МІСЦЕВІ ВОДЯНІ ТРЕБІ ТА ІНШІ ГАЛУЗІ. ТОМУ ЦЕ ДУЖЕ ВИПІДНА І ЕФЕКТИВНА СИСТЕМА ДЛЯ МІСЦЬ, ДЕ НЕМОЖЛИВО НЕГАЙНО ВЖИТИ ЗАХОДІВ ЩОДО ПРОБЛЕМ ЗІ ЗДОРОВ'ЯМ, СПРИЧИНЕНИХ ЯКІСТЮ ВОДИ.



- ▶ У цьому докладі представлено огляд сучасного стану розробки та впровадження систем WQM на основі WSN, опис різних структур систем WQM на основі WSN та обговорення технологій, що використовуються на різних етапах процесу моніторингу.
- ▶ Подібні системи моніторингу вважаються більш ніж перспективними й наукові розробки в цьому напрямку, дозволять захистити, настільки важливий, для нашої цивілізації, ресурс, як запас питної води.

ДЯКУЮ ЗА УВАГУ!

## ДОДАТОК Б ПРОГРАММНА СКЛАДОВА

### DataSync ZigBee 1:

```
import serial
import time
from xbee import ZigBee

print 'Asynchronously printing data from remote XBee'

serial_port = serial.Serial('/dev/ttyUSB0', 9600)

def print_data(data):
    """
    This method is called whenever data is received.
    Its only argument is the data within the frame.
    """
    print data['samples']

zigbee = ZigBee(serial_port, escaped=True, callback = print_data)

while True:
    try:
        time.sleep(0.001)
    except KeyboardInterrupt:
        break

zigbee.halt();
serial_port.close()
```

### DataSync ZigBee 2:

```
import serial
from xbee import ZigBee
```

```

print 'Printing data from remote XBee'

serial_port = serial.Serial('COM4', 9600)
zigbee = ZigBee(serial_port, escape=True)

while True:
    try:
        print zigbee.wait_read_frame()
    except KeyboardInterrupt:
        break

zigbee.halt()
serial_port.close()

```

## Temperature Sensor:

```

import serial
from xbee import ZigBee

def getTemperatureFromData(data):
    temperature = data['samples'][0]['adc-0']
    # temperature = temperature*18.0/212.0
    temperature = 3.3 * temperature * 100.0 / 1024.0
    return temperature

def turnFanOn(xbee):
    print 'Threshold exceeded! Activating Fans'
    xbee.send('remote_at',
              frame_id='A',

dest_addr_long='\x00\x13\xA2\x00\x40\x8B\x96\x2E',
              dest_addr='\xFF\xFE',
              options='\x02',
              command='D1',
              parameter='\x05')

def turnFanOff(xbee):
    xbee.send('remote_at',
              frame_id='A',

dest_addr_long='\x00\x13\xA2\x00\x40\x8B\x96\x2E',

def main():
    print 'Receiving temperature values...'
    serialPort = serial.Serial('COM4', 9600)

```

```

xbee = ZigBee(serialPort, escaped=True)

while True:
    try:
        data = xbee.wait_read_frame()
        if data.keys().count('samples') !=
0:
            temperature =
getTemperatureFromData(data)
            print temperature
            if (temperature > 21.0):
                turnFanOn(xbee)
            else:
                turnFanOff(xbee)

    except KeyboardInterrupt:
        xbee.send('remote_at',
                    frame_id='A',

dest_addr_long='\x00\x13\xA2\x00\x40\x8B\x96\x2E',

dest_addr='\xFF\xFE',

                    options='\x02',
                    command='D1',

parameter='\x04')
        break

    xbee.halt()
    serialPort.close()

if __name__ == '__main__':
    main()
else:
    pass

```

## WSN Node 1

```

int BUTTON = 2;
int ACK_led = 12;

void setup()
{
    pinMode(BUTTON, INPUT);

```

```

// initialize the digital pin as an output.
pinMode(ACK_led, OUTPUT);
Serial.begin(9600);
}

void loop(){
  // SEND
  // send a capital D over the serial port if the button is pressed
  if(digitalRead(BUTTON)==HIGH)
  {
    Serial.print("D");
    delay(300); // prevents overwhelming the serial port
  }

  // RECEIVE
  if (Serial.available()>0)
  {
    if (Serial.read() == 'A')
    {
      digitalWrite(ACK_led, HIGH);
      delay(500);

      digitalWrite(ACK_led, LOW);
    }
  }
}

```

## WSN Node 2

```

const int BELL = 5;

void setup()
{
  pinMode(BELL, OUTPUT);
  Serial.begin(9600);
}

void loop()
{
  if (Serial.available() > 0)
  {
    if (Serial.read() == 'D')
    {
      Serial.print("A");

      analogWrite(BELL, 240);
      delay(10);
      analogWrite(BELL, 0);
    }
  }
}

```

```
}

```

## WSN Node 3

```
function Decoder (bytes, port) {
  var result = {};
  var transformers = {};

  if (port==33) {
    transformers = {
      'temperature': function transform (bytes) {
        value=bytes[0]*256 + bytes[1];
        if (value>=32768) value=value-65536;
        return value/100.0;
      },
      'humidity': function transform (bytes) {
        return (bytes[0]*256 + bytes[1])/100.0;
      },
      'pressure': function transform (bytes) {
        return (bytes[0]*256 + bytes[1])/10.0;
      },
      'altitude': function transform (bytes) {
        return (bytes[0]*256 + bytes[1])/10.0;
      },
      'vbattery': function transform (bytes) {
        return (bytes[0]*256 + bytes[1])/1000.0;
      }
    }
  }

  result['temperature'] = {
    value: transformers['temperature'](bytes.slice(0, 2)),
    uom: 'Celsius',
  }

  result['humidity'] = {
    value: transformers['humidity'](bytes.slice(2, 4)),
    uom: 'Percent',
  }

  result['pressure'] = {
    value: transformers['pressure'](bytes.slice(4, 6)),
    uom: 'hPa',
  }

  result['altitude'] = {
    value: transformers['altitude'](bytes.slice(6, 8)),
    uom: 'Meter',
  }
}
```



