

Лабораторне заняття 2

Організація введення інформації з цифрового датчика вологості DHT11

Мета: ознайомитися з принципами роботи датчика DHT11. Отримати практичні навички по підключенню датчика до мікроконтролера, ознайомитись с діаграмами роботи датчика та організації отримання значень з датчика.

Теоретичні відомості

Вхідні сигнали бувають аналогові і цифрові. Аналоговий сигнал – це сигнал, величина якого безперервно змінюється в часі. Аналогові сигнали природним чином передають мову, музику і зображення. Для обробки аналогових сигналів в МПС здійснюється перетворення сигналу в цифрову форму.

Більшість датчиків є аналоговими, тому необхідно сигнал перетворити в цифрову форму за допомогою аналогово-цифрового перетворювача (АЦП). АЦП може бути як вбудований у датчик, так і бути зовнішнім.

У даній роботі розглянемо роботу датчика DHT11 з вбудованим АЦП (рисунок 2.1). Основне призначення пропонованого датчика – дистанційне вимірювання температури і вологості.

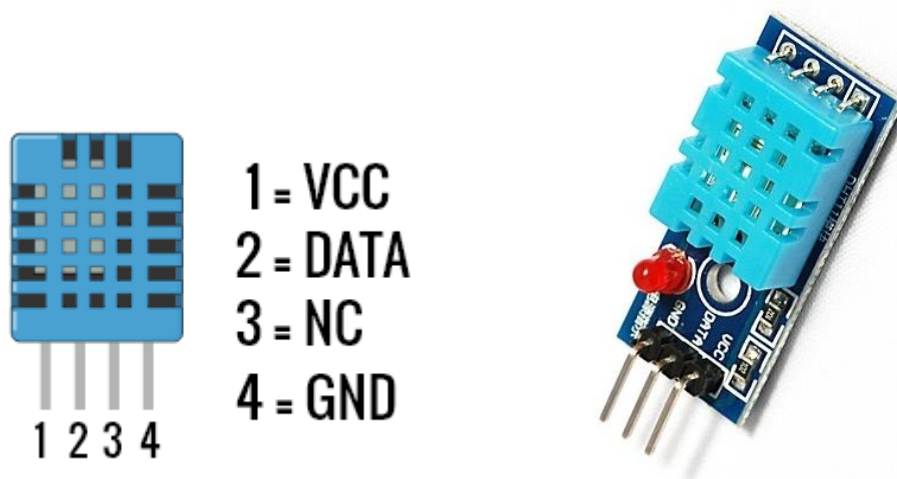


Рисунок 2.1 –Зовнішній вигляд датчика та модуля DHT11

Основні характеристики датчика:

- Вимірювання вологості в діапазоні від 20% до 90% (при температурі зовнішнього середовища 25 градусів);
- Похибка вимірювання вологості становить до 5%;
- Роздільна здатність вимірювання вологості становить 1%;
- Вимірювання температури в діапазоні від 0 до 50 градусів;
- Похибка вимірювання температури становить до 2 градусів;
- Роздільна здатність вимірювання температури становить 1 градус;
- Максимальна відстань підключення датчика до 20 метрів (можна і більше, але потрібно вносити зміни в схему підключення);
- Частота опитування датчика становить 1 Гц;
- Ток споживання до 2,5 мА (максимальне значення при перетворенні даних);
- Джерело живлення – від 3 до 5.5 Вольт;
- Габаритні розміри: 15.5 мм – довжина; 12 мм – ширина; 5.5 мм – висота.

При підключенні датчика DHT11 до МК (рисунок 2.2) необхідно між виводами Vcc і Data розмістити підтягуючий резистор номіналом 4.7-10 кОм. Плата Arduino має вбудовані підтягуючі резистори, однак вони не відповідають номіналу.

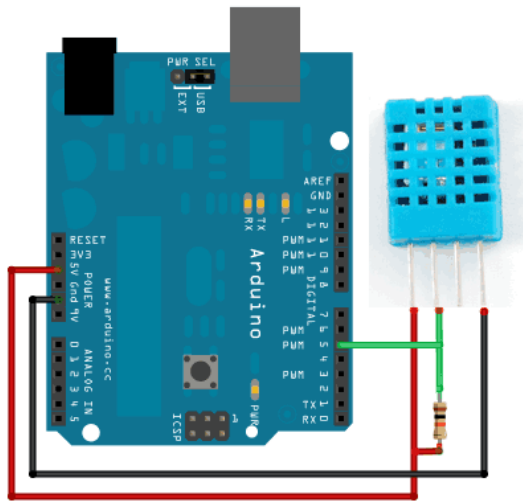


Рисунок 2.2 – Схема підключення датчика DHT11

При підключенні модуля (рисунок 2.3) встановлення резистору не потрібно (він підключений на платі модулю):

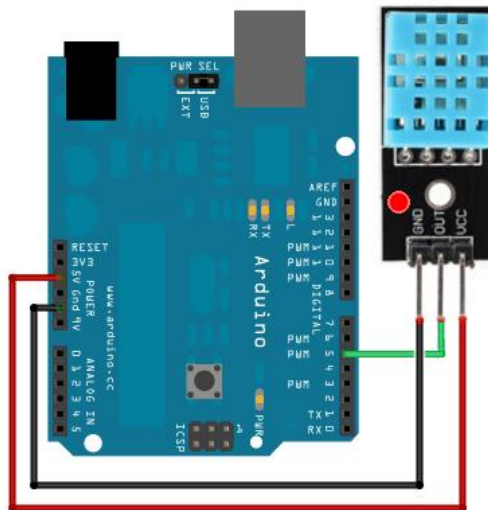


Рисунок 2.3 – Схема підключення модулю DHT11

МК виступає в ролі ведучого пристрою шини і, отже, відповідає за ініціювання зв'язку. Датчик вологості і температури DHT11 завжди залишається в якості підлеглого пристрою і відповідає, коли МК просить його. Протокол, який використовується для зв'язку, простий і може бути представлений наступним чином (рисунок 2.4) та (рисунок 2.5):



Рисунок 2.4 – Протокол обміну з датчиком DHT11

DHT11 Data Format

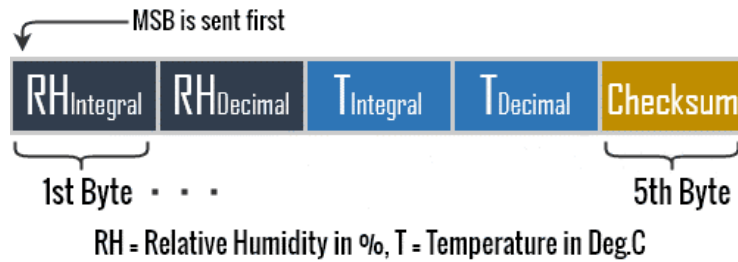


Рисунок 2.5 – Формат даних датчика DHT11

Розглянемо алгоритм роботи з датчиком. Алгоритм можна розділити на 4 етапи.

1. Ініціювання обміну: процес ініціювання обміну починається з підтягування лінії до землі на 18 мс, а далі відпускаємо після 20-40 мкс і переводимо ніжку мікроконтролера у режим читання (рисунок 2.6).

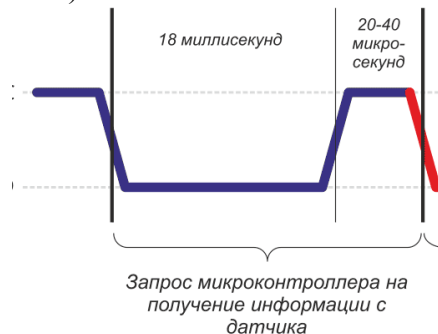


Рисунок 2.6 – Початок обміну датчика DHT11

2. Підтвердження датчика: датчик, виявивши, що лінія звільнилася (рівень стане "1"), підтягує лінію до землі на 80 мкс, за цей час робить вимірювання і перетворення результату. Потім датчик відпускає лінію після 80 мкс, що вказує на те, що він готовий відправити дані (рисунок 2.7).

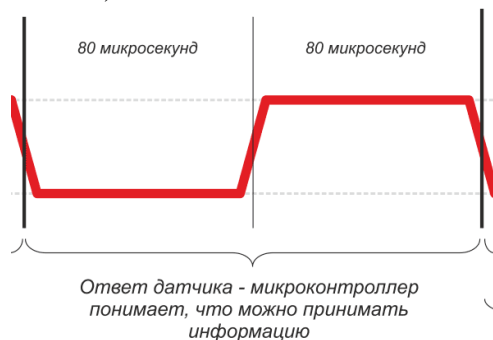


Рисунок 2.7 – Відповідь датчика, щодо готовності до обміну

3. Прийом даних: далі датчик відправляє 40 біт даних. Перед відправкою кожного біта датчик притискає лінію до землі на 50 мкс, за яким слід імпульс тривалістю 26-28 мкс для "0" або 70 мкс для "1" (рисунок 2.8). Після передачі лінія звільняється.

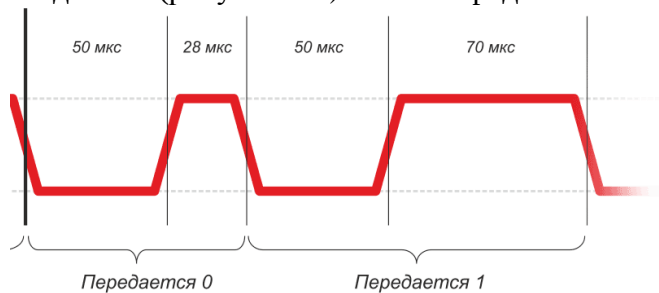


Рисунок 2.8 – Діаграми нуля та одиниці при обміні

4. Перевірка контрольної суми: визначається як сума всіх перших 4х байт.

Приклад:

Лістинг 2.1 – Приклад роботи з датчиком

```
/*Указываем вывод порта к которому подключен датчик*/
#define DHT_PORT PORTC
#define DHT_DDR DDRC
#define DHT_PIN PINC
#define DHT_BIT 6

uint8_t humidity, temperature; /*значение влажности и температуры*/
uint8_t dht_read();

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

// the loop function runs over and over again forever
void loop() {
  digitalWrite(LED_BUILTIN, LOW); /*выключим светодиод*/
  delay(500);
  dht_read();
  digitalWrite(LED_BUILTIN, HIGH); /*включим светодиод, когда данные будут готовы*/
  delay(500);
  /*вывод результата в монитор*/
  Serial.print(humidity);
  Serial.println(" ");
  Serial.print(temperature);
  Serial.println(" *C");
}

/*Функция считывания данных с датчика DHT11*/
uint8_t dht_read() {
  uint8_t datadht[5]; /*массив для значений датчика*/

  /*1.*/
  DHT_DDR |= (1 << DHT_BIT); /*настраиваем вывод на выход*/
  DHT_PORT &= ~(1 << DHT_BIT); /*низкий уровень — подтягиваем линию-разбудим датчик*/
  _delay_ms(18); /*18 мс - старт*/
  cli(); /*запрещаем прерывания*/
  DHT_PORT |= (1 << DHT_BIT); /*отпускаем линию*/
  _delay_us(40);

  /*2.*/
  DHT_DDR &= ~(1 << DHT_BIT); /*настраиваем вывод на вход*/
  _delay_us(10);
  if (DHT_PIN & (1 << DHT_BIT)) { return -1; } /*датчик должен ответить "0"*/
  _delay_us(80);
  if (!(DHT_PIN & (1 << DHT_BIT))) { return -1; } /*по истечению 80 мкс, датчик должен отпустить шину*/

  /*3.*/
  while (DHT_PIN & (1 << DHT_BIT)); /*ожидаем пока контроллер датчика начнет передавать данные*/
  /*передача начинается с нуля*/
  for (uint8_t j = 0; j < 5; j++) { /*считываем 5 байт*/
    datadht[j] = 0;
    for (uint8_t i = 0; i < 8; i++) { /*считываем побитово*/

      while (!(DHT_PIN & (1 << DHT_BIT))); /*ожидаем когда датчик отпустит шину*/
      _delay_us(30); /*задержка высокого уровня на 30 мкс*/
      if (DHT_PIN & (1 << DHT_BIT)) /*если по истечению времени сигнал на линии высокий, значит передается "1"*/
        datadht[j] |= 1 << (7 - i); /*тогда i-й бит устанавливаем в "1"*/
      while (DHT_PIN & (1 << DHT_BIT)); /*ждем окончание "1" */

    }
  }
  sei(); /*разрешаем общее прерывание*/
  _delay_us(50);

  /*4.*/
  /*проверка контрольной суммы*/
  uint8_t sum = datadht[0] + datadht[1] + datadht[2] + datadht[3];
  if (datadht[4] != sum) return -1;
```

```
/*для DHT11 дробная часть равна нулю, поэтому возвращаем только целые значения*/  
humidity = datadht[0];  
temperature = datadht[2];  
  
return 0;  
}
```

Зміст роботи

Завдання 1: Ввести приклад 2.1. перевірити роботу за допомогою монітору середовища розробника. Розібратися з роботою функції читання інформації з датчика.

Завдання 2: Реалізувати аналогічну функції за допомогою бібліотеки Arduino (фреймворкWiring). Значення вологості вивести на семисегментний індикатор.