

ПРАКТИЧНА РОБОТА №4

Тема: Побудова блок – схеми мікроелектронної інформаційної системи керування електричними мережами

Мета: Способи використання особливостей архітектури та функціонування контролерів Arduino (на базі ATmega328) при створенні мікроелектронних інформаційних систем управління електричними мережами

4.1 Структурна схема пристрою

З урахуванням прийнятих попередніх рішень структурна схема пристрою для вимірювання струму та напруги в мережі живлення повинна включати наступні компоненти:

1) Мікроконтролер ARDUINO NANO або будь-який інший подібний мікроконтролер (наприклад, ARDUINO UNO, MICRO, MINI) використовується як основний керуючий елемент пристрою. Він відповідає за збір даних, обробку і керування пристроєм.

2) Датчик струму ACS712 30A GY-712 використовується для вимірювання струму в мережі живлення. Він здатний вимірювати струм до 30A і генерує аналоговий сигнал, який подається на мікроконтролер для подальшої обробки.

3) Ethernet модуль ENC28J60 дозволяє забезпечити з'єднання з мережею Інтернет і передавати дані з пристрою на віддалений сервер або інші пристрої.

4) Модуль живлення TSP-05 220В - 5В 3Вт або будь-яке інше джерело живлення 5В для Arduino використовується для живлення мікроконтролера і інших електронних компонентів пристрою.

5) Трансформатор 220/12В невеликої потужності використовується для зниження напруги з мережі 220В до потрібного рівня напруги для живлення ізованих компонентів пристрою.

6) Чотири резистори 0.25W використовуються як датчики напруги.

7) Електролітичний конденсатор на 10 мкФ використовується для стабілізації напруги і забезпечення плавного живлення електронних компонентів.

Загалом, ця структурна схема, яка показана на рис. 4.1 дозволяє вимірювати струм та напругу в мережі живлення за допомогою датчика струму ACS712, обробляти та керувати цими даними за допомогою мікроконтролера ARDUINO та передавати результати через Ethernet модуль ENC28J60. Використовуються також інші компоненти, такі як джерело живлення, трансформатор, резистори та електролітичний конденсатор, для забезпечення правильної роботи пристрою.

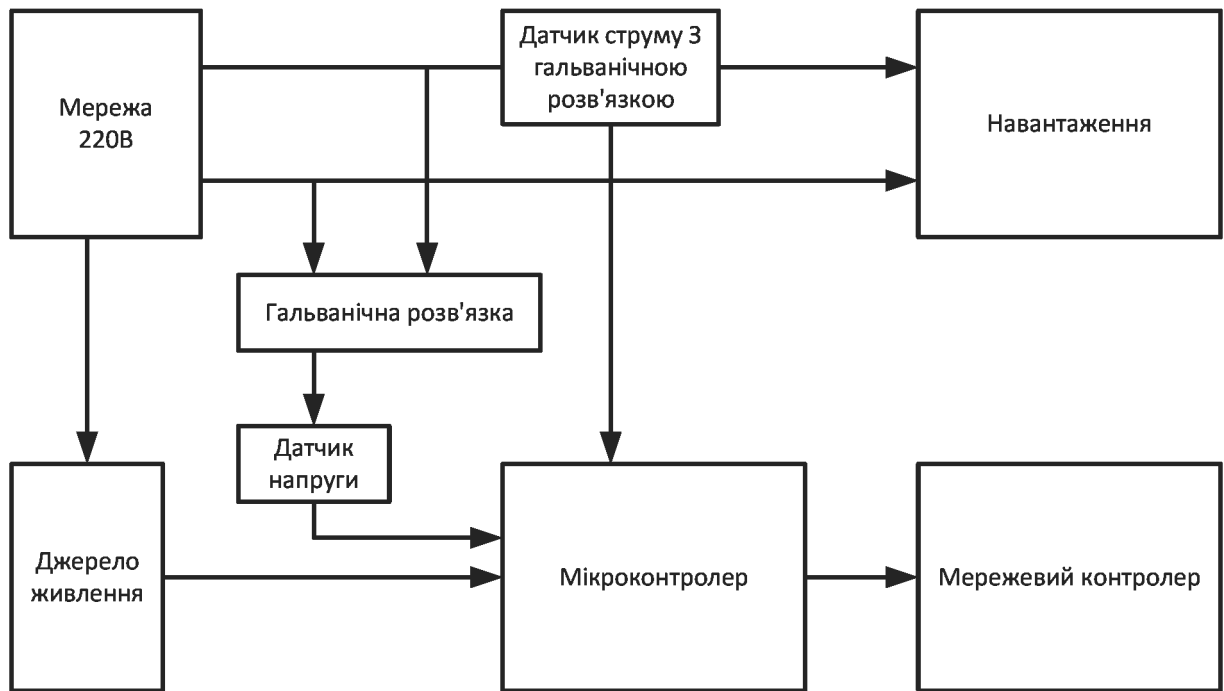


Рисунок 4.1 - Структурна схема пристрою

4.2 Аналіз комплектуючих

Для вимірювання струму в електричній мережі використовується датчик ACS712 30A GY-712, який побудований на однойменній мікросхемі ACS712ELCTR-30A-T фірми Allegro, схема підключення показано на рис 4.2.

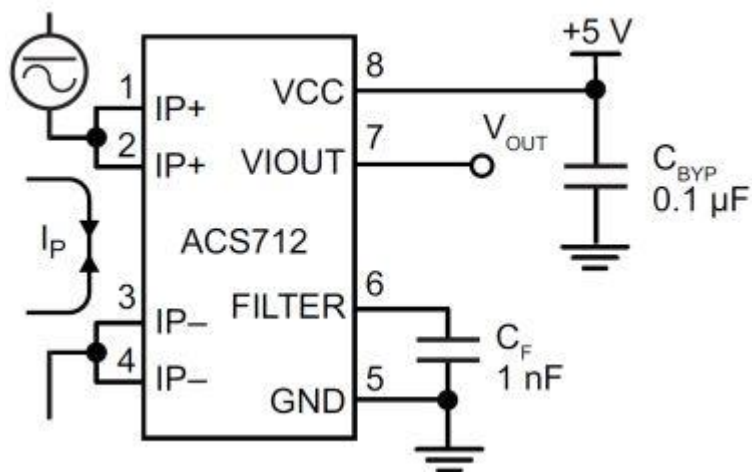


Рисунок 4.2 — Схема підключення датчику струму

Ця мікросхема забезпечує точне вимірювання постійного та змінного електричного струму за рахунок використання ефекту Холла і забезпечує гальванічну розв'язку високовольтної та вимірювальної частини кола. Нижче показано важливу схему включення мікросхеми.

Вимірюваний струм, I_p , протікає від контактів 1, 2 до контактів 3,4. На виході мікросхеми (контакт 7) при $I_p=0$, напруга дорівнює половині напруги живлення мікросхеми $V_{out} = 5/2 = 2.5\text{В}$. При $I_p=+30\text{А}$ (струм протікає у напрямку від контактів 1,2 до контактів 3,4) на виході мікросхеми (контакт 7)

буде напруга $V_{out} = 2.5 + 30 \cdot 0.066 = 4.48\text{В}$. При $I_p = -30\text{А}$ (струм протікає у напрямку від контактів 3,4 до контактів 1,4) на виході мікросхеми (контакт 7) буде напруга $V_{out} = 2.5 - 30 \cdot 0.066 = 0.52\text{В}$.

Таким чином, при протіканні змінного струму в контрольованій електричній мережі, на аналоговому вході А0 мікроконтролера на базі ARDUINO NANO буде напруга, яка змінюється за синусоїдою з частотою, що дорівнює частоті в електричній мережі. Амплітуда напруги на вході А0 буде пропорційна електричного струму (лінія А0 на графіку внизу).

Для вимірювання напруги в електричній мережі використовується звичайний трансформатор 220/12В, який підключений через чотири резистори та конденсатор до аналогового входу А1 мікроконтролера на базі ARDUINO NANO згідно схеми на рис. 4.3. Ці чотири резистори та конденсатор забезпечують зменшення амплітуди коливань напруги зі зміщенням до напруги, рівному половині напруги живлення ($5/2=2.5\text{В}$) на вході мікроконтролера. У результаті на вході А1 при напрузі в електричній мережі рівній нулю буде напруга 2.5В. У точці максимуму амплітуди напруги 220В, напруга на А1 складатиме приблизно 3.6В. У точці мінімуму амплітуди напруги, напруга на А1 буде приблизно 1.44В. Цю напругу можна підігнати до потрібного діапазону шляхом підбору пари резисторів 100кОм та 5.6кОм.

У результаті на вході А1 має бути синусоїда, як показано на графіку рис 4.4.

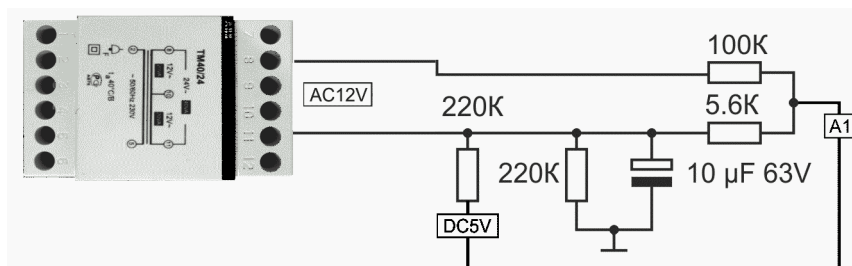


Рисунок 4.3 — Схема вимірювання змінної напруги

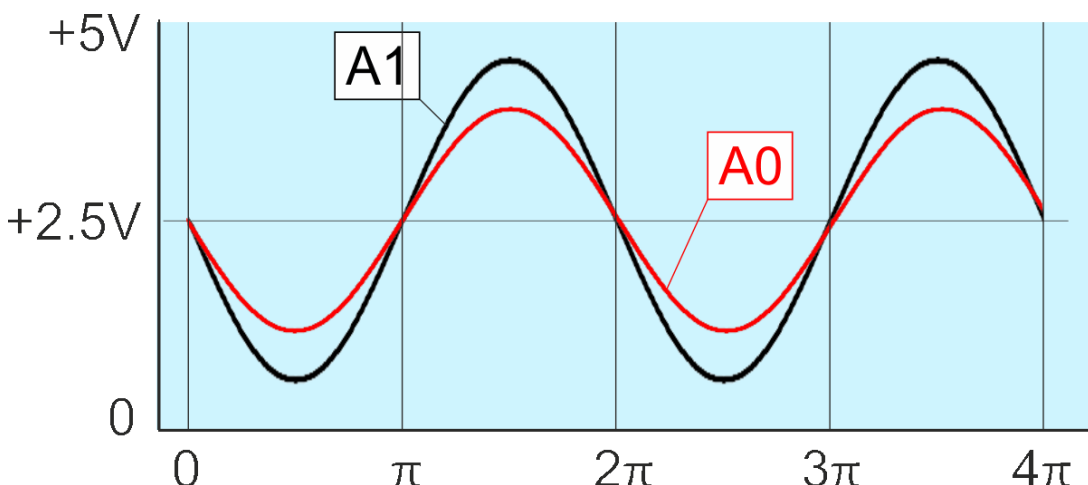


Рисунок 4.4 — Сигнал на вході мікроконтролера

У колах змінного синусоїдального струму, внаслідок постійної зміни значення напруги та струму, потужність не можна обчислити шляхом

простого множення напруги на струм. Тому виділяють відразу три види електричної потужності: активну, реактивну та повну.

Активна потужність у колах синусоїдального струму

Одиниця виміру - ват (позначення: Вт; міжнародне позначення: W).

$$P=U*I*\cos \varphi$$

де P - активна потужність, Вт;

U - середньоквадратична напруга, В;

I - середньоквадратичний струм, А;

φ - кут зсуву фаз напруги та струму, град.

Активна потужність визначає частину електричної енергії, яка використовується безпосередньо на виконання корисної роботи.

Реактивна потужність у колах синусоїдального струму

Одиниця виміру - вольт-ампер реактивний (позначення: вар; міжнародне позначення: var)

$$Q = U*I*\sin \varphi$$

де Q - реактивна потужність, вар;

U - середньоквадратична напруга, В;

I - середньоквадратичний струм, А;

φ - кут зсуву фаз напруги та струму, град.

Реактивна потужність визначає ту частину електричної енергії, яка марно витрачається в електричних мережах.

Повна потужність у колах синусоїдального струму

Одиниця повної електричної потужності - вольт-ампер (позначення: ВА; міжнародне позначення: VA)

$$S= \sqrt{P^2 + Q^2}$$

де S - повна потужність, ВА;

P - активна потужність, Вт;

Q - реактивна потужність, вар;

Повна потужність відповідає всій енергії, яка витрачається в електричних мережах.

4.2 Алгоритм обчислення та код проекту

Через аналогові входи A0 і A1 постійно зчитуємо поточні значення напруги та струму, підносимо їх у квадрат і підсумовуємо. З періодичністю в одну секунду обчислюємо середньоквадратичне значення для напруги та струму. Добуток середньоквадратичного значення напруги і струму дає величину повної потужності.

При розробці пристрою також буде потрібна бібліотека UIPEthernet. В IDE Arduino її можна встановити через меню "Скетч"->"Підключити бібліотеку" - "Керувати бібліотеками" - "Менеджер бібліотек" - UIPEthernet. Також можна завантажити цю бібліотеку з GitHub за адресою <https://github.com/UIPEthernet/UIPEthernet>.

Після завершення проектування лістинг отриманої програми наведено нижче.

```
#include <UIPEthernet.h>
#define DEMO true //встановить false для реального виміру
const char page1[] PROGMEM =
"<!DOCTYPE html>\n"
"<html>\n"
"<meta charset='UTF-8'>\n"
"<title>WEB MONITOR</title>\n"
"<link href='http://liccontrol.com/web/gauge.ico' rel='icon' type='image/x-
icon'/>\n"
"<link rel='stylesheet' href='http://liccontrol.com/web/style.css'
type='text/css'>\n"
"<script src='http://code.jquery.com/jquery-3.1.1.min.js'></script>\n"
"<script src='http://code.highcharts.com/highcharts.js'></script>\n"
"</head>\n"
;
const char page2[] PROGMEM =
"<body>\n"
"<table id='MainTab' border='0' cellspacing='0' cellpadding='0'>\n"
"<tbody><tr ><td><div id='trend'></div></td></tr></tbody>\n"
"</table>\n"
"<script src='http://liccontrol.com/web/trend.js'></script>\n"
"</body></html>\n"
;
const char page3[] PROGMEM =
"HTTP/1.1 200 OK\n"
"Content-Type: text/html\n"
"Pragma: no-cache\n"
"Connection: close\r\n\r\n"
;

int IPin = A1; //Аналоговий вхід для вимірювання струму
int VPin = A0; //Аналоговий вхід для вимірювання напруги
//поточне та сумарне значення струму
double I=0.0, I1 = 0.0, I2 = 0.0;
//поточне та сумарне значення напруги
double V=0.0, V1 = 0.0, V2 = 0.0;
float S = 0.0; //повна потужність
float P = 0.0; //активна потужність
float Q = 0.0; //Реактивна потужність
float P1 = 0.0; //миттєва потужність
float kI = 0.07398; //Коефіцієнт вимірювання струму
float kV = 0.97656; //Коефіцієнт вимірювання напруги
//Унікальна тас-адреса. Якщо буде повторення в мережі,
```

```

//Ethernet - модуль працювати не буде!
byte mac[] = {0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED};
//Встановить відповідну для вас адресу
IPAddress ip(192, 168, 1, 10);
EthernetServer server(80);
String result="";
unsigned long time1 = 0, count=0, count1=0;
unsigned long t1 = 0;
boolean Restart = true;

//Перезапуск мікросхеми ENC28J60
void resetENC28J60() {
digitalWrite(3, LOW);
delay(200);
digitalWrite(3, HIGH);
delay(100);
}

//Процедура вимірювання струму та напруги
void MeasureData() {
unsigned long dt = (unsigned long)(millis() - t1);
//миттєве значення струму, А
I1 = kI*(analogRead(IPin) - 512);
if (DEMO) I1 = sin(1.5*(millis()-time1)/60000.0)*10.0+20.0;
I2 += pow(I1,2);
//миттєве значення напруги, В
V1 = kV*(analogRead(VPin) - 512);
if (DEMO) V1 = sin((millis()-time1)/60000.0)*50.0+200.0;
V2 += pow(V1,2);
P1 += I1 * V1;
if ((dt>1000)and(count>0)) {
count1++;
I = sqrt(I2/count); //Середньоквадратичне значення струму, А
V = sqrt(V2/count); //Середньоквадратичне значення напруги,
S = I * V; //повна потужність, ВА
P = P1/count; //активна потужність, Вт
if (DEMO) P=P*(0.8 + 0.2*sin(0.5*(millis()-time1)/60000.0));
float _Q = pow(S,2)-pow(P,2);
if (_Q>0.0) Q = sqrt(_Q); //реактивна потужність, вар
else Q = 0.0;
P1 = 0.0;
V2 = 0.0;
I2 = 0.0;
count = 0;
t1 = millis();
}
}

```

```

}
count++;
}

//Процедура надсилання до мережі даних, які зберігаються в PROGMEM
void printProgStr (EthernetClient client, const char * str){
char c;
if (!str) return;
while ((c = pgm_read_byte(str++))) client.print (c);
}

//Формуємо JSON рядок даних для відправки її через мережу до браузера
String DataToBrowser() {
int t = int((millis()-time1)/1000);
String s = String(t)+", "+String(V,1)+"", "+
String(I,1)+"", "+String(S/1000.0,2)+"", "+
String(P/1000.0,2);
return s;
}

//Початкові установки та ініціалізація
void setup() {
Serial.begin(9600);
resetENC28J60();
Ethernet.begin(mac, ip);
server.begin();
Serial.print("START SERVER AT:");
Serial.println(Ethernet.localIP());
}

//Головний цикл виконання програми
void loop() {
MeasureData();
EthernetClient client = server.available();
if (client) {
result="";
while (client.connected()) {
if (client.available()) {
char c = client.read();
result+=c;
if (c == '\n') {
int y=result.indexOf("dat");
if ((y>0) and (!Restart)) {
printProgStr(client, page3);
String s = DataToBrowser();

```

```
client.println(s);
} else {
printProgStr(client, page1);
printProgStr(client, page2);
Restart = false;
time1 = millis();
}
break;
}
}
}
delay(1);
client.stop();
}
}
```

Підключіть модуль ENC28J60 Ethernet до мережі. Після завантаження програми в контролер, у пошуковому рядку браузера наберіть <http://192.168.1.10/> та натисніть Enter.

Комп'ютер повинен бути обов'язково підключений до Інтернету. В цьому випадку з'явиться сторінка з демонстраційними графіками. Після цього можна підключити датчик струму - ACS712 30A GY-712, трансформатор та підібрати коефіцієнти kI,kV щоб отримати правильні дані.

Потрібно встановити `#define DEMO false` коли треба вимірювати реальні дані. Це параметр потрібен для відладки.

Контрольні питання

1. Як використовуються мікроконтролери для мікроелектронних інформаційних систем управління електричними мережами?
2. Які комплектуючі потрібні для реалізації типового проекту?
3. Яка платформа використовувалась для вирішення поставлених питань?