

URZĄDZENIA PERYFERYJNE

SPRAWOZDANIE

ANALIZATOR PARAMETRÓW SIECI

Damian Koper, 241292

Wiktor Pieklik, 241282

1 grudnia 2019

Spis treści

1	Cel ćwiczenia	4
2	Podstawowe zagadnienia związane z siecią elektryczną	4
2.1	Napięcie	4
2.2	Prąd	5
2.3	Moc czynna	6
2.4	Moc bierna	6
2.5	Moc pozorna	7
2.6	Współczynnik mocy czynnej do mocy pozornej	7
2.7	Harmoniczne prądu	8
3	Podstawowe pojęcia związane z komunikacją sieciową w modelu warstwowym TCP/IP	8
3.1	Adres IP	8
3.2	Maska podsieci	9
3.3	Brama sieciowa	9
3.4	DHCP	9
4	Protokół modbus TCP/IP	9
4.1	Komunikacja	9
4.2	Ramka wiadomości	10
5	Przebieg laboratorium	11
5.1	Konfiguracja urządzeń	11
5.2	Listing	11

Spis rysunków

1	Moc czynna, chwilowa i moc bierna podczas cykli zmian napięcia	7
2	Analiza zjawiska odkształceń prądów i napięć	8

Spis listingów

1 Cel ćwiczenia

Celem ćwiczenia było zapoznanie się z urządzeniem służącym do analizy parametrów sieciowych używanym w automatyce. Na zajęciach dostępny był analizator parametrów sieci EMA-90N firmy Contrel, który umożliwia kontrolę stanu sieci elektrycznej poprzez mierzenie parametrów takich jak: napięcie, prąd, moc czynna, moc bierna, $\cos(\phi)$, harmoniczne prądu i inne. Urządzenie umożliwia komunikację za pomocą Ethernet przy wykorzystaniu protokołu modbus.

2 Podstawowe zagadnienia związane z siecią elektryczną

2.1 Napięcie

Napięcie elektryczne - to różnica potencjałów między dwoma punktami obwodu elektrycznego. Symbolem napięcia jest U , a samo napięcie wyraża się w Voltach [V].

Napięcie można również wyrazić jako stosunek pracy wykonanej przeciwko polu, podczas przeniesienia ładunku elektrycznego między punktami, dla których określa się napięcie, do wartości tego ładunku. Określa to wzór:

$$U_{AB} = \varphi_B - \varphi_A = \frac{W_{A \rightarrow B}}{q} \quad (1)$$

2.1.1 Napięcie sieciowe

Napięcie sieciowe – napięcie elektryczne występujące w sieci niskiego napięcia danego kraju. Napięcie sieciowe ma charakterystykę sinusoidalną i w zależności od kraju: częstotliwość 50 lub 60 Hz i napięcie od 100 do 240 V (w Polsce 230 V / 50 Hz – określa to Polska Norma PN-IEC 60038). Większość urządzeń powszechnego użytku jest zasilana z wykorzystaniem napięcia sieciowego lub napięciem przetworzonym z napięcia sieciowego, przy użyciu transformatorów napięcia. Należy zaznaczyć, że wartość 230 V to wartość skuteczna napięcia przemiennego. Przy 230 V szczytowa wartość napięcia sieciowego wynosi $230V * \sqrt{2} = 325V$. Dopuszczalne odchylenia wynoszą $\pm 10\%$, czyli od 207 do 253 V.

2.2 Prąd

Prąd elektryczny - to uporządkowany ruch ładunków elektrycznych. Nośnikami tych ładunków mogą być elektrony lub jony.

Ściśle związaną jednostką z prądem jest **natężenie prądu**, czyli wielkość fizyczna charakteryzująca przepływ prądu elektrycznego zdefiniowana jako stosunek wartości przepływającego ładunku elektrycznego do czasu przepływu tego ładunku. Symbolem natężenia prądu elektrycznego jest I i wyrażane jest w Amperach [A]. Natężenie prądu określa się wzorem:

$$I = \frac{\partial q}{\partial t}, \quad (2)$$

Gdzie:

- ∂q - zmiana ładunku równoważna przepływającemu ładunkowi,
- ∂t - czas przepływu ładunku,
- I - natężenie prądu

2.2.1 Prąd zmienny

Prąd zmienny - to prąd elektryczny, dla którego wartość natężenia zmienia się w czasie w dowolny sposób. W zależności od charakteru tych zmian można wyróżnić następujące rodzaje prądu:

- prąd okresowo zmienny
 - prąd tętniący
 - prąd przemienny
- prąd nieokresowy

Wszystkie powyższe pozycje poza ostatnią są przypadkami szczególnymi prądu zmiennego i mają one swoje specjalne znaczenie w elektrotechnice i elektronice. Prąd zmienny nieokresowy może reprezentować prąd o dowolnej zmienności w czasie lub też prąd zmieniający się zgodnie z określoną funkcją matematyczną lub na skutek określonego zjawiska fizycznego. Na przykład uderzenie pioruna powoduje powstanie fali udarowej o określonym kształcie, która przebiega jednorazowo, nie ma więc charakteru okresowego. Często termin prąd zmienny stosowany jest do prądu okresowego przemiennego o przebiegu sinusoidalnym (ang.: alternating current, AC).

Prąd przemienny (ang. alternating current, AC) – charakterystyczny przypadek prądu elektrycznego okresowo zmiennego, w którym wartości chwilowe podlegają zmianom w powtarzalny, okresowy sposób, z określoną częstotliwością. Wartości chwilowe natężenia prądu przemiennego przyjmują naprzemiennie wartości dodatnie i ujemne. Najczęściej pożądanym jest, aby wartość średnia cało okresowa (tzn. składowa stała) wynosiła zero. Największe znaczenie praktyczne mają prąd i napięcie o przebiegu sinusoidalnym. W żargonie technicznym nazwa prąd przemienny często oznacza po prostu prąd sinusoidalny. Jeśli zakłócenia lub nieliniowość powodują zdeformowanie sinusoidalnego kształtu, wówczas taki niesinusoidalny przebieg nosi nazwę przebiegu odkształconego.

2.3 Moc czynna

W układach prądu przemiennego (zarówno prądu zmiennego) jest to część mocy, którą odbiornik pobiera ze źródła i zamienia na pracę lub ciepło. W układach prądu stałego cała moc jest mocą czynną. Moc czynna wyrażana jest w Watach [W].

Moc czynna jest średnią mocą, co dla przebiegu okresowego prądu i napięcia wyraża całka Riemanna:

$$P = \frac{1}{T} \int_0^T u(t)i(t)dt \quad (3)$$

Gdzie:

- P - moc czynna,
- t - czas,
- T - okres,
- u - napięcie chwilowe,
- i - chwilowe natężenie prądu

2.4 Moc bierna

W obwodach prądu zmiennego jest wielkością opisującą pulsowanie energii elektrycznej między elementami obwodu elektrycznego. Ta oscylująca energia nie jest zamieniana na użyteczną pracę lub ciepło, niemniej jest ona konieczna do funkcjonowania maszyn elektrycznych (np. transformatorów, silników). Energia jest pobierana ze źródła w części okresu przebiegu zmiennego, magazynowana przez odbiornik (w postaci energii pola elektrycznego lub magnetycznego) i oddawana do źródła

w innej części okresu, co jest związane z zanikiem pola w odbiorniku. Jednostką mocy biernej (Q) jest var (z ang. var – Volt Ampere Reactive). Dla przebiegów sinusoidalnie zmiennych moc bierna jest definiowana jako iloczyn wartości skutecznych napięcia i prądu, oraz sinusa kąta przesunięcia fazowego między napięciem a prądem:

$$Q = UI \sin(\varphi) \quad (4)$$

Gdzie:

- U, I - wartości skuteczne napięcia i prądu,
- φ - przesunięcie fazowe pomiędzy napięciem i prądem

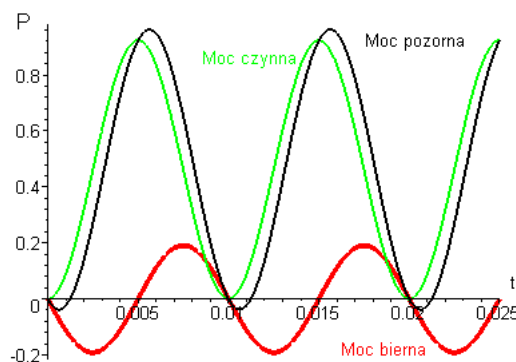
2.5 Moc pozorna

Moc pozorna - to wielkość fizyczna określana dla obwodów prądu przemiennego. Wyraża się ją jako iloczyn wartości skutecznych napięcia i natężenia prądu.

$$S = UI, \quad (5)$$

Gdzie:

- U, I - wartości skuteczne napięcia oraz natężenia prądu



Rysunek 1: Moc czynna, chwilowa i moc bierna podczas cykli zmian napięcia

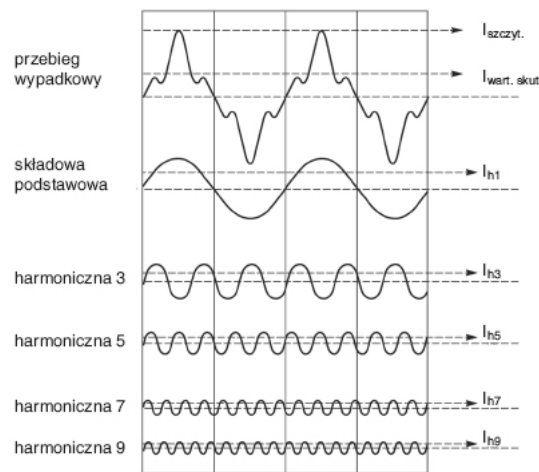
2.6 Współczynnik mocy czynnej do mocy pozornej

Współczynnik mocy $\cos\varphi$ - to stosunek mocy czynnej do mocy pozornej, czyli stosunek mocy użytecznej do iloczynu napięcia i prądu.

$$\cos(\varphi) = \frac{P}{S}, \quad (6)$$

2.7 Harmoniczne prądu

Harmoniczna - jest napięciem lub prądem o wielokrotności częstotliwości podstawowej systemu energii elektrycznej, wytwarzanym przez działanie nieliniowych obciążeń, takich jak prostowniki, oświetlenie wyładowcze lub nasycone urządzenia magnetyczne. Częstotliwości harmoniczne w sieci energetycznej są częstą przyczyną problemów z jakością energii. Harmoniczne w systemach elektroenergetycznych powodują wzrost temperatury w urządzeniach i przewodach, przerwy w zapłonie w napędach o zmiennej prędkości oraz pulsacje momentu obrotowego w silnikach.



Rysunek 2: Analiza zjawiska odkształceń prądów i napięć

3 Podstawowe pojęcia związane z komunikacją sieciową w modelu warstwowym TCP/IP

3.1 Adres IP

Adres IP - jest to liczba nadawana interfejsowi sieciowemu w protokole IP, służąca identyfikacji elementów sieci w warstwie trzeciej modelu ISO/OSI.

IPv4 używa adresu 32 bitowego. Adres ten zapisuje się dziesiętnie oddzielając kolejne oktety kropkami. Na adres ten składają się dwa elementy: numer sieci oraz numer komputera w sieci, przy czym wielkość tych elementów może się zmieniać.

Przykładowy adres IPv4: *192.168.1.10*

3.2 Maska podsieci

Maska podsieci pozwala na wyodrębnienie w adresie IP części będącej adresem podsieci i części, która jest adresem hosta w tej podsieci.

Podobnie jak adres IP, **maska podsieci** to również 32 bitowy ciąg, posiadający wartość 1 na pozycjach bitów odpowiadających numerowi sieci, a wartość 0 na pozycjach bitów odpowiadających numerowi urządzenia w sieci.

Przykładowa maska podsieci: *255.255.255.0*

3.3 Brama sieciowa

Brama sieciowa - to maszyna podłączona do sieci komputerowej, za pośrednictwem której komputery z sieci lokalnej komunikują się z komputerami w innych sieciach.

3.4 DHCP

DHCP umożliwia podłączonym do sieci komputerom pobieranie adresu IP, maski podsieci, adresu bramy i serwera DNS oraz innych ustawień ze skonfigurowanej wcześniej puli adresów. Serwer DHCP może być skonfigurowany na osobnym komputerze i stanowił będzie osobne urządzenie w sieci przydzielające komputerom klienckim adresy IP, może również działać na już istniejącym serwerze jako osobna usługa, osobny proces.

4 Protokół modbus TCP/IP

4.1 Komunikacja

Modbus jest protokołem komunikacyjnym działającym w modelu master-slave, który jest w stanie wspierać komunikację z 247 urządzeniami typu slave zorganizowanych jako magistrala lub topologia gwiazdy. Komunikacja odbywa się w trybie half duplex.

Modbus TCP/IP jest protokołem Modbus wspierającym interfejs TCP działający w Ethernetie. Komunikacja sieciowa może być typu: pytanie-odpowiedź lub broadcastowa. *Pytanie-odpowiedź* jest transmitowana z urządzenia typu master do nawiązanego połączenia z urządzeniem typu slave i po tym następuje wiadomość z odpowiedzią. *Broadcast* jest transmitowany z urządzenia typu master do wszystkich urządzeń typu slave i po nim nie następuje odpowiedź. Struktura przesyłania

komunikatów Modbus to protokół aplikacji, który określa reguły organizowania i interpretowania danych niezależnie od nośnika transmisji danych. TCP/IP odnosi się do protokołu kontroli transmisji i protokołu internetowego, który zapewnia medium transmisyjne dla wiadomości Modbus TCP/IP. Mówiąc wprost, TCP/IP umożliwia wymianę bloków danych binarnych między komputerami. Jest to również światowy standard, który służy jako podstawa dla sieci WWW. Podstawową funkcją TCP jest zapewnienie, że wszystkie pakiety danych zostaną odebrane poprawnie, podczas gdy IP zapewnia, że wiadomości są poprawnie adresowane i kierowane. Zasadniczo komunikat Modbus TCP/IP to po prostu komunikacja Modbus zamknięta w opakowaniu Ethernet TCP/IP. W praktyce Modbus TCP osadza standardową ramkę danych Modbus w ramce TCP, bez sumy kontrolnej Modbus, jak pokazano na poniższym schemacie.

4.2 Ramka wiadomości

Ramka wiadomości Modbus RTU składa się z następujących segmentów:

1. **START FRAME**: znacznik rozpoczynający wiadomość
2. **TRANSACTION ID**: pole identyfikacyjne służy do parowania transakcji, gdy wiele wiadomości jest wysyłanych wzdłuż tego samego połączenia TCP przez klienta bez oczekiwania na wcześniejszą odpowiedź
3. **PROTOCOL ID**: to pole ma zawsze wartość 0 dla usług *Modbus*, a inne wartości są zarezerwowane dla przyszłych rozszerzeń
4. **LENGTH**: to pole jest liczbą bajtów pozostałych pól i obejmuje bajt identyfikatora jednostki, bajt kodu funkcji i pola danych
5. **UnitID**: to pole służy do identyfikacji zdalnego serwera zlokalizowanego w sieci innej niż TCP/IP (do mostkowania szeregowego).
6. **ERROR CHECK**: pole uwzględniające kod korekcyjny (jednakże pole sprawdzania błędów Modbus (suma kontrolna) nie jest używane jako standardowy TCP/IP w Ethernetie. Zamiast tego stosuje się metody sumy kontrolnej warstwy łącza danych w celu zagwarantowania integralności danych)
7. **END OF FRAME**: znacznik kończący wiadomość

5 Przebieg laboratorium

5.1 Konfiguracja urządzeń

Podczas laboratorium połączyliśmy bezpośrednio ze sobą komputer z analizatorem parametrów sieci EMA-90N kablem typu skręta zakończonym złączami RJ45.

Komputerowi nadaliśmy stały adres IP: **192.168.50.50**, natomiast analizatorowi **192.168.50.100**.

Oba te adresy mają domyślną maskę podsieci: **255.255.255.0**. Komunikację między urządzeniami sprawdziliśmy wpierw za pomocą polecenia **ping** protokołu ICMP, a następnie za pomocą programu testowego, który znajduje się pod adresem: http://www.zsk.ict.pwr.wroc.pl/zsk/repository/dydaktyka/up/lab/analizator_sieci/ema90n_demo.zip.

5.2 Listing

```
1 TcpClient client;
2 NetworkStream stream;
3 bool capture = false;
4 float activePowerCalcVal = 0;
5 int currentSleep = 0;
6
7 public Form1()
8 {
9     InitializeComponent();
10    client = new TcpClient("192.168.50.50", 502);
11    stream = client.GetStream();
12    Task.Run(async () => await CaptureProc());
13 }
14 private async Task CaptureProc()
15 {
16     int sleep = 200;
17     var voltageSeries = voltageChart.Series["voltageSeries"];
18     using (StreamWriter file = File.CreateText("data.csv"))
19     {
20         file.WriteLine("time,voltage,current,cosfi,
21             powerActive,powerActiveCalc,powerActiveDiff");
22         while (true){
23             if (capture){
```

```

24     float blabla = currentSleep / 1000;
25     string buff = blabla.ToString("0.00", System.Globalization.CultureInfo.
↪ InvariantCulture);
26     float voltageVal =
27     readRegister(new byte[] { 0x10, 0x02 }) / 1000;
28     voltage.Invoke(new Action(()
29     => voltage.Text = voltageVal.ToString()));
30     buff += voltageVal.ToString("0.00", System.Globalization.CultureInfo.
↪ InvariantCulture) + ",";
31     float currentVal =
32     readRegister(new byte[] { 0x10, 0x10 }) / 100;
33     current.Invoke(new Action(()
34     => current.Text = currentVal.ToString()));
35     buff += currentVal.ToString("0.00", System.Globalization.CultureInfo.
↪ InvariantCulture) + ",";
36     float cosfiVal =
37     readRegister(new byte[] { 0x10, 0x20 }) / 1000;
38     cosfi.Invoke(new Action(() => cosfi.Text = cosfiVal.ToString()));
39     buff += cosfiVal.ToString("0.00", System.Globalization.CultureInfo.
↪ InvariantCulture) + ",";
40     float powerActiveVal = readRegister(new byte[] { 0x10, 0x30 });
41     powerActive.Invoke(new Action(() =>
42     powerActive.Text = powerActiveVal.ToString()));
43     buff += powerActiveVal.ToString("0.00", System.Globalization.
↪ CultureInfo.InvariantCulture) + ",";
44     activePowerCalcVal =
45     Math.Abs(voltageVal * currentVal * cosfiVal) / 10;
46     buff += activePowerCalcVal.ToString("0.00", System.Globalization.
↪ CultureInfo.InvariantCulture) + ",";
47     float diff = Math.Abs(activePowerCalcVal - powerActiveVal);
48     buff += diff.ToString("0.00", System.Globalization.CultureInfo.
↪ InvariantCulture);
49     activePowerCalc.Invoke(new Action(()
50     => activePowerCalc.Text = activePowerCalcVal.ToString()));
51     activePowerDiff.Invoke(new Action(() =>
52     activePowerDiff.Text = diff.ToString()));
53     file.WriteLine(buff);
54     file.Flush();

```

```

55         voltageChart.Invoke(new Action(() => {
56             voltageChart.Series["voltageSeries"].Points.AddXY(currentSleep,
↪ voltageVal);
57             voltageChart.Series["powerSeries"].Points.AddXY(currentSleep,
↪ powerActiveVal);
58             voltageChart.Invalidate();
59         }));
60         currentSleep += 200;
61         Thread.Sleep(sleep);
62     }
63 }
64 }
65 }
66 private float readRegister(byte[] register)
67 {
68     byte[] request = { 0x00, 0x01, 0x00, 0x00, 0x00, 0x06, 0x01, 0x03, register
↪ [0], register[1], 0x00, 0x02 };
69     stream.Write(request, 0, request.Length);
70     byte[] answer = new byte[100];
71     stream.Read(answer, 0, answer.Length);
72     byte[] cut = new byte[4];
73     Array.ConstrainedCopy(answer, 9, cut, 0, 4);
74     Array.Reverse(cut);
75     float value = BitConverter.ToInt32(cut, 0);
76     return value;
77 }
78 private void button1_Click(object sender, EventArgs e)
79 {
80     if (capture){
81         capture = false;
82         button1.Text = "Wlacz mierzonko";
83         button1.BackColor = default(Color);
84     } else{
85         capture = true;
86         button1.BackColor = Color.Red;
87         button1.Text = "Wylacz mierzonko";
88         voltage.Text = 0.ToString();
89         current.Text = 0.ToString();

```

```
90     currentSleep = 0;
91     voltageChart.Invoke(new Action(() => {
92         voltageChart.Series["voltageSeries"].Points.Clear();
93         voltageChart.Series["powerSeries"].Points.Clear();
94         voltageChart.Invalidate();
95     }));
96 }
97 }
```