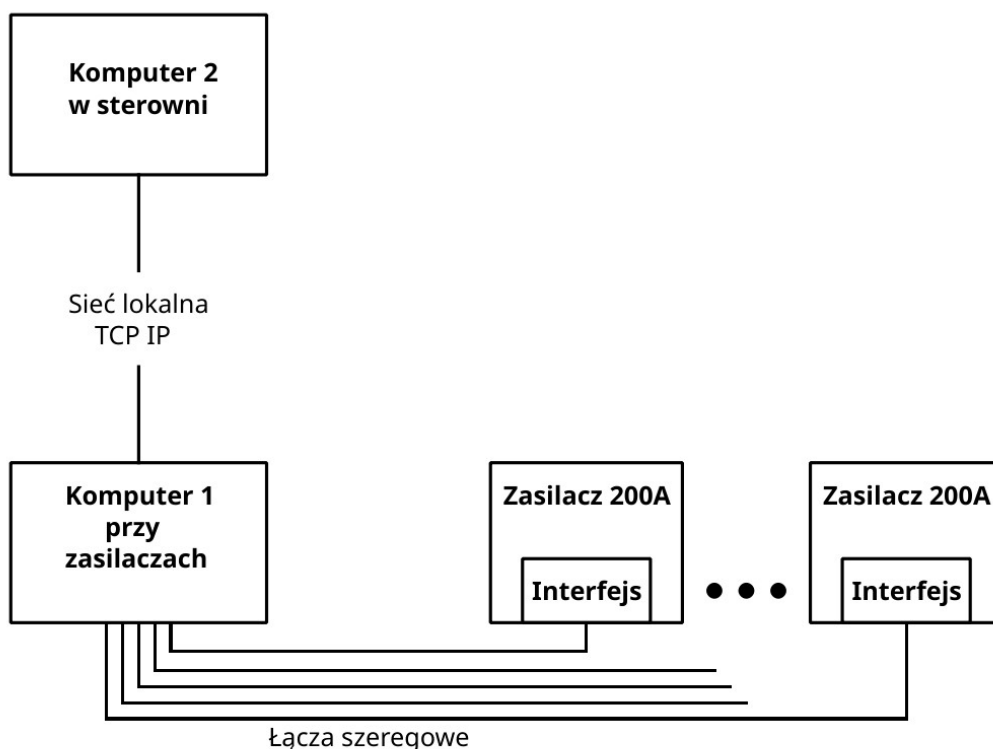


# Opis aplikacji do sterowania zasilaczami 200A z interfejsami Pico

## wersja 1.0

### 1. Opis platformy sprzętowej

Aplikacja działa na komputerach z Linuxem. Platformę sprzętową pokazuje uproszczony schemat na rysunku 1.



Rys. 1.

Ta sama aplikacja działa na komputerze 1 (w trybie lokalnym) i na komputerze 2 (w trybie zdalnym). Tryb pracy aplikacji jest zapisany w pliku konfiguracyjnym (tekstowym), który jest odczytywany w momencie uruchomienia aplikacji. Składnia pliku konfiguracyjnego będzie opisana dalej.

Komputer 1 może mieć kilka - kilkanaście sprzętowych portów szeregowych, połączonych z zasilaczami. Zasilacze są wyposażone w interfejsy do komunikacji Modbus RTU. Interfejsy są zbudowane na bazie płytki Raspberry Pi Pico i są opisane w osobnym dokumencie.

### 2. Opis funkcjonalny aplikacji

Aplikacja pracująca w trybie lokalnym na komputerze 1 umożliwia sterowanie zasilaczami za pomocą protokołu Modbus RTU. Opis tego protokołu jest zawarty w dokumencie nt. interfejsu Pico. Aplikacja udostępnia w sieci TCP/IP serwer (slave) Modbus TCP. Przestrzeń adresowa i komendy obsługiwane przez serwer będą opisane dalej. Aplikacja w trybie lokalnym pozwala użytkownikowi wybrać, czy sterowanie jest lokalne (na komputerze 1), czy zdalne (na komputerze 2).

Aplikacja pracująca w trybie zdalnym (na komputerze 2) może wykonywać wszystkie funkcje sterowania zasilaczami.

### 3. Opis funkcjonalny komunikacji między komputerami z protokołem Modbus TCP

Parametry serwera (slave'a) Modbusa TCP, takie jak numer portu i adres IP, są zapisane w pliku konfiguracyjnym.

Serwer obsługuje następujące komendy Modbusowe:

03 Read Holding Registers

06 Preset Single Register

Rejestry Modbusowe zostały opisane w Tabeli 1. Wszystkie rejestry są 16-bitowe bez znaku.

Przestrzeń adresowa serwera Modbusa TCP składa się z dwóch części.

Część 1 zawiera dane robocze, które mogą się zmieniać w czasie pracy z zasilaczami. Część 1 mieści się w zakresie adresów 1000 - 2621. Część 1 składa się z sektorów zawierających po 22 rejestry każdy. Liczba tych sektorów zależy od liczby zasilaczy zadeklarowanych w pliku konfiguracyjnym na komputerze 1. Liczba sektorów jest o jeden większa od liczby zasilaczy. Kolejne sektory rozpoczynają się od adresów: 1000, 1100, 1200, itd.

Część 2 zawiera dane pomocnicze, niezmiennie w czasie pracy aplikacji. Jest to zestaw tekstów opisujących zasilacze, przekopiowanych z pliku konfiguracyjnego na komputerze 1 i udostępnionych aplikacji na komputerze 2. Część 2 składa się z sektorów, podobnie jak część 1, ale ich rozmiary mogą być różne. Liczba tych sektorów jest taka sama, jak w części 1, czyli jest o jeden większa od liczby zasilaczy zadeklarowanych w pliku konfiguracyjnym na komputerze 1.

Adres	Odczyt/ zapis	Nazwa / zakres wartości	Jednostka	Dodatkowy opis	
1000	ro	Sterowanie lokalne czy zdalne <sup>1</sup> / 0 ... 1			Sektor 0
1001	ro	Liczba zadeklarowanych zasilaczy (w pliku konfiguracyjnym komputera 1) / 1 ... 16			
1002 ...1021	ro	Tekstowy identyfikator wersji aplikacji <sup>2</sup>			
1100	ro	Włączenie / 0 ... 1		zaakceptowane	Sektor 1: Zasilacz 1
1101	ro	Prąd zadany / 0 ... 65535	(200 / 65535) A	wartości zadane (zapisane w interfejsie zasilacza)	
1102	ro	Identyfikator zasilacza (liczba 0 ... 255 ustawiana za pomocą przełączników SW1 i SW2 w zasilaczu <sup>3</sup> )			
1103	ro	Stan zasilacza; bit 0 = Power on; bit 1 = Remote; bit 2 = Ext. error; bit 3 = Sum. error			
1104	ro	Prąd zmierzony - wartość średnia	0.01 A	Statystyka z 32 ostatnich próbek; f <sub>sample</sub> = 89 Hz	
1105	ro	Prąd zmierzony - mediana	0.01 A		
1106	ro	Prąd zmierzony - średnia z 50% próbek po odrzuceniu 50% próbek skrajnych	0.01 A		
1107	ro	Prąd zmierzony - wartość międzyszczytowa	0.01 A		
1108	ro	Prąd zmierzony - odchylenie standardowe	0.01 A		
1109	ro	Napięcie zmierzone - wartość średnia	0.01 V		
1110	ro	Napięcie zmierzone - mediana	0.01 V		
1111	ro	Napięcie zmierzone - średnia z 50% próbek po odrzuceniu 50% próbek skrajnych	0.01 V		
1112	ro	Napięcie zmierzone - wart. międzyszczytowa	0.01 V		
1113	ro	Napięcie zmierzone - odchylenie standard.	0.01 V		
1114	ro	Błędy transmisji Modbus RTU	%	Statystyka z ostatnich	
1115	ro	Najdłuższy ciąg błędów transmisji Modbus RTU		512 wysłanych komend	
1116	ro	Starszy bajt = Status komunikacji <sup>4</sup> Młodszy bajt = Potwierdzenie komunikacji <sup>5</sup>			
1117	ro	Starszy bajt = Status przełącznika mocy (kopia bitu 0 w rejestrze 1103); Młodszy bajt = zgodność identyfikatora odczytanego z zasilacza (rejestr 1102) z identyfikatorem odczytanym z pliku konfiguracyjnego (rejestr 1119): 0 = niezgodny 1 = zgodny			
1118	ro	Starszy bajt = Stan maszyny stanów wyłączania zasilacza <sup>6</sup> Młodszy bajt = Ostatni błąd ramki odpowiedzi Modbusa RTU <sup>7</sup>			
1119	ro	Oczekiwany identyfikator zasilacza (odczytany z pliku konfiguracyjnego)			
1120	r/w	Zlecenie od GUI na komputerze 2 (zdalnym) <sup>8</sup>			
1121	r/w	Wartość zadana prądu (zapis do rejestru oznacza zlecenie od GUI na komputerze 2 zmiany prądu w zasilaczu)	(200 / 65535) A		
1200 ...1221	Sektor 2 jest zdefiniowany, jeżeli rejestr 1001 > 1. Liczba kolejnych sektorów jest określona przez rejestr 1001. Sektor 2, jak również kolejne sektory, mają identyczną strukturę, jak sektor 1, tylko odnoszą się do zasilacza 2 i kolejnych zasilaczy.				Sektor 2

Tabela 1 (będzie kontynuowana).

Adres	Odczyt/ zapis	Nazwa / zakres wartości	Jednostka	Dodatkowy opis	
4000	ro	Długość opisu zasilacza 1 (liczba bajtów)			Sektor 0
4001	ro	Długość opisu zasilacza 2 (liczba bajtów)			
⋮					
4015	ro	Długość opisu zasilacza 16 (liczba bajtów)			Sektor 1
4100	ro	Tekst o długości równej rejestrowi 4000 (sektor 1 zajmuje rejestry w liczbie równej 1/2 rejestru 4000)			
⋮					
4200	ro	Tekst o długości równej rejestrowi 4001 (sektor 2 zajmuje rejestry w liczbie równej 1/2 rejestru 4001)			Sektor 2

Tabela 1 (kontynuacja).

<sup>1</sup> Sterowanie lokalne czy zdalne:

0 = sterowanie lokalne (steruje komputer 1, komputer 2 tylko monitoruje);

1 = sterowanie zdalne (steruje komputer 2, komputer 1 tylko monitoruje)

<sup>2</sup> Tekstowy identyfikator wersji aplikacji (zawierający datę i czas zapisania do repozytorium ostatniej wersji programu) ma zapewnić, żeby na obu komputerach program miał tą samą wersję.

<sup>3</sup> Przełączniki SW1 i SW2 są pokazane na dokumencie BM\_interface\_schematic\_200A.pdf

<sup>4</sup> Status komunikacji

0 = port szeregowy nie otwiera się (nie istnieje, albo nie daje się skonfigurować)

1 = brak ramki odpowiedzi

2 = niekompletna ramka odpowiedzi

3 = błędne CRC ramki odpowiedzi

4 = inny błąd ramki odpowiedzi

5 = ramka odpowiedzi prawidłowa

<sup>5</sup> Potwierdzenie komunikacji:

0 = nie odebrano żadnej prawidłowej ramki

1 = odebrano co najmniej jedną prawidłową ramkę

<sup>6</sup> Stan maszyny stanów wyłączania zasilacza:

0 = maszyna stanów nieaktywna

1 = oczekiwanie na wyzerowanie prądu przed timeout'em

2 = oczekiwanie na wyzerowanie prądu po timeout'cie

<sup>7</sup> Ostatni błąd ramki odpowiedzi Modbusa RTU:

0 = błąd nieokreślony

1 = brak ramki odpowiedzi

2 = niekompletna ramka odpowiedzi

3 = błędne CRC ramki odpowiedzi

4 = inny błąd ramki odpowiedzi

5 = brak błędów

<sup>8</sup> Zlecenie od GUI na komputerze 2 (zdalnym):

0 = brak zlecenia

3 = włącz przełącznik mocy zasilacza

4 = wyłącz przełącznik mocy zasilacza (GUI może wysłać to zlecenie, jeżeli maszyna stanów wyłączania zasilacza jest w stanie 2)

6 = uruchom wyłączanie zasilacza za pośrednictwem maszyny stanów (GUI może wysłać to zlecenie, jeżeli maszyna stanów jest w stanie 0)

## 4. Opis pliku konfiguracyjnego

Plik konfiguracyjny jest plikiem tekstowym o nazwie `powerSource200A_Svedberg.cfg` i musi znajdować się w tym samym katalogu, co program wykonywalny. Plik konfiguracyjny musi zawierać określone deklaracje w określonej kolejności. Składnia pliku konfiguracyjnego będzie przedstawiona na przykładach. Na komputerze lokalnym plik konfiguracyjny może wyglądać następująco:

```
# Plik konfiguracyjny programu do obsługi zasilaczy 200A
```

```
tryb_pracy_komputera=lokalny
numer_portu_tcp=1502
id=13      port='/dev/ttyS0'      opis='Magnes 1'
id=0x66    port='/dev/ttyS1'      opis='Magnes 2'
id=51      port='/dev/ttyS2'      opis='Magnes 3'
id=52      port='/dev/ttyS3'      opis='Magnes 4'
id=67      port='/dev/ttyUSB0'    opis='Magnes 5'
id=53      port='/dev/ttyS4'      opis='Magnes 6'
```

Na komputerze zdalnym przykładowy plik konfiguracyjny może być następujący:

```
# Plik konfiguracyjny programu do obsługi zasilaczy 200A
```

```
tryb_pracy_komputera=zdalny
numer_portu_tcp=1502
adres_tcp=192.168.1.10
```

Pierwsza linia w powyższych przykładach zawiera dowolny komentarz. Może być dowolna liczba linii z komentarzami.

Linia ze słowem kluczowym **tryb\_pracy\_komputera** deklaruje tryb pracy aplikacji: lokalny albo zdalny.

Linia ze słowem kluczowym **numer\_portu\_tcp** deklaruje port TCP używany do komunikacji pomiędzy komputerami 1 i 2.

Pozostałe deklaracje różnią się, w zależności od trybu pracy. W przypadku trybu lokalnego musi być zadeklarowany przynajmniej jeden zasilacz. Opis każdego zasilacza jest zawarty w jednej linii i składa się z trzech słów kluczowych.

Słowo kluczowe **id** oznacza fizyczny numer identyfikacyjny zasilacza, ustawiony za pomocą przełączników SW1 i SW2 (pokazanych na dokumencie `BM_interface_schematic_200A.pdf`). Ten numer można podać w postaci dziesiętnej (bez przedrostka) lub szesnastkowej (z przedrostkiem 0x). Wymagana jest zgodność numeru identyfikacyjnego zadeklarowanego w pliku konfiguracyjnym z numerem fizycznie ustawionym w zasilaczu, aby móc sterować zasilaczem. Jeżeli numery będą niezgodne aplikacja odczyta fizyczny numer identyfikacyjny zasilacza i możemy go zobaczyć w oknie informacji dodatkowych (należy nacisnąć przycisk "i" po prawej stronie okna aplikacji).

Słowo kluczowe **port** oznacza pełną nazwę portu szeregowego wraz ze ścieżką dostępu.

W deklaracji **opis** można podać dowolny tekst (polskie znaki są akceptowane, ale znaki specjalne np. <CR><LF> nie są poprawnie interpretowane). Ta deklaracja jest konieczna, nawet jeśli nie potrzeba żadnego opisu do zasilacza, np.:

```
id=53 port='/dev/ttyS4' opis=''
```

Deklaracje zasilaczy należy umieszczać tylko na komputerze lokalnym.

Deklaracja **adres\_tcp** informuje aplikację na komputerze zdalnym, gdzie ma szukać serwera (slave'a) Modbusa TCP w sieci komputerowej.

W razie problemów z plikiem konfiguracyjnym należy uruchomić aplikację w konsoli z parametrem -v (verbose).

## 5. Wybrane aspekty implementacji

Źródła aplikacji zawierają pakiet freemodbus oraz port do linuxa (podkatalog LINUXTCP), które pochodzą ze strony:

[https://github.com/harryg-oss/freemodbus-rp2040/tree/dev\\_rp2040](https://github.com/harryg-oss/freemodbus-rp2040/tree/dev_rp2040)

Nazwa ściągniętego pliku:

freemodbus-rp2040-dev\_rp2040.zip

**Historia wersji**

Wersja	Data	Opis
1.0	2025-05-22	Utworzenie dokumentu.