

Piotr POREBA<sup>1</sup>, Marcin SIDZINA

Opiekun naukowy: Marcin SIDZINA<sup>2</sup>

## **BADANIE CZASU TRWANIA WYMIANY DANYCH W PROTOKOLE PROFINET DLA WYBRANEJ GRUPY URZĄDZEŃ**

**Streszczenie:** W artykule przybliżono tematykę protokołów komunikacyjnych w szeroko pojętej automatyce przemysłowej, bazujących na sieci typu Ethernet, w szczególności sieci PROFINET RT (ang. Real Time), dla której przeprowadzone zostały badania pomiaru czasu trwania wymiany danych dla wybranych urządzeń. Przedstawiony został program badań doświadczalnych w protokole PROFINET, opis stanowiska laboratoryjnego z wyszczególnieniem użytego sprzętu, metody pomiarowe, sposób w jaki zmieniane były kluczowe parametry oraz wyniki badań w postaci tabeli i wykresów.

**Słowa kluczowe:** PROFINET, Czas wymiany danych, Systemy rozproszone czasu rzeczywistego

## **TESTING THE DURATION OF DATA EXCHANGE IN THE PROFINET PROTOCOL FOR A SELECTED GROUP OF DEVICES**

**Summary:** The article presents the subject of communication protocols in broadly understood industrial automation, based on Ethernet networks, especially PROFINET RT (Real Time) networks, for which research was carried out to measure the duration of data exchange for selected devices. The program of experimental research in the PROFINET protocol was presented, as well as a description of the laboratory stand with the specification of the equipment used, measurement methods, and how the key parameters and test results were changed in the form of tables and graphs.

**Keywords:** PROFINET, Time of data exchange, , Real Time Fieldbus system

---

<sup>1</sup> mgr inż., Akademia Techniczno-Humanistyczna w Bielsku-Białej, Wydział Budowy Maszyn i Informatyki, poreba.piotr.bb@gmail.com

<sup>2</sup> dr inż., Akademia Techniczno-Humanistyczna w Bielsku-Białej, Wydział Budowy Maszyn i Informatyki, msidzina@ath.bielsko.pl

## 1. Wprowadzenie

Niezbędnym elementem nowoczesnych koncepcji automatyki jest komunikacja cyfrowa, wykorzystująca standardowe protokoły komunikacyjne. Dzięki decentralizacji komunikacji systemów automatyki ograniczona zostaje liczba przewodów łączących poszczególne części maszyny, co przekłada się na mniejszy stopień skomplikowania instalacji. Coraz bardziej złożona staje się natomiast tematyka przetwarzania danych w czasie rzeczywistym. Urządzenia oraz technologie nimi zarządzające są specyficzną dziedziną wiedzy, która przenika w strefę informatyki i systemów IT. Do najbardziej popularnych i najczęściej wybieranych obecnie protokołów w szeroko pojętej automatyce przemysłowej są te bazujące na sieci typu Ethernet. Decyzja, którego protokołu komunikacyjnego użyć, zależy od wielu czynników. Najczęściej uzależnione jest to od wymagań, które stawia już istniejąca infrastruktura sieciowa. Często dostawca systemu sterowania narzuca zastosowanie konkretnego rozwiązania, więc już na etapie doboru sprzętu należy przewidzieć, czy parametry techniczne protokołu komunikacyjnego są wystarczające do danej aplikacji. Duże znaczenie ma również cena, która jest powiązana z osprzętem, oraz późniejsze wsparcie producenta czy dostęp do dokumentacji technicznej [1, 9, 17].

## 2. Podstawowe informacje na temat standardu IEEE 802.3

Podstawą nowoczesnych i obecnie stosowanych przemysłowych protokołów komunikacyjnych jest Ethernet, czyli standard rozwiązań sieciowych, które dotyczą warstwy fizycznej oraz łącza danych. Wersje tego standardu ewoluowały wraz z czasem jego istnienia. Początkowo istniały tzw. Thicknet (tzw. gruby Ethernet) oraz Thinnet (cienki Ethernet). Oferowały one jednak niewielką część możliwości względem dzisiejszego Ethernetu. Wersje te pracowały na miedzianym przewodzie koncentrycznym w topologii magistrali, co wymagało sterowania dostępem do nośnika poprzez metodę CSMA/CD, badającą stan kanałów komunikacyjnych oraz wykrywającą kolizje w sieci. Rozwiązanie to nie było jednak wystarczająco efektywne. Wraz z rozwojem sieci zaczęto używać kabla typu skrętka oraz nowej topologii – gwiazdy. W tym przypadku zamiast przełączników będących centralnymi punktami sieci zaczęto stosować koncentratory (ang. *hub*), wysyłające dane do wszystkich podłączonych do niego urządzeń bez względu na to, czy owo urządzenie te dane dotyczyły. Ograniczało to przepustowość sieci oraz zwiększało prawdopodobieństwo wystąpienia kolizji w sieci [16].

Dalsze prace nad rozwojem Ethernetu doprowadziły do zaprojektowania inteligentniejszego rozwiązania, jakim jest przełącznik (ang. *switch*), który rozdziela dane do urządzeń, do których są one adresowane. Wyeliminowało to ryzyko wystąpienia kolizji w sieci. Adresowanie urządzeń odbywa się na podstawie adresów MAC (ang. *Media Access Control*), co jest pewną podstawą działania sieci Ethernet (IEEE 802). Adres MAC jest unikalny dla danego urządzenia bez możliwości jego powielenia w skali światowej. Adres ten nadawany jest przez producenta karty sieciowej. Składa się z 6 bajtów, które określają m.in. producenta karty i jej unikalne ID. MAC adresy przechowywane są w pamięci przełącznika. Pozyskiwanie takiego adresu odbywa się dynamicznie, bez udziału administratora, poprzez odczyt z ramki danych. Switch automatycznie zapamiętuje również numer portu, z którego została

odebrana ramka. Dodatkowo przełącznik może mieć dedykowaną przepustowość danych, która będzie jednakowa dla wszystkich podłączonych urządzeń, np. 100 Mb/s. Przełączniki pracują w trybie pełnego duplexu, co przekłada się na wysyłanie i odbiór danych w jednym momencie [16].

Standard Ethernet stosowany jest w kilku wersjach. Najbardziej popularny w komercyjnym użytkowaniu jest FastEthernet, w którym nominalna przepustowość to 100 Mb/s. W przypadku gdy wymagana jest większa przepustowość, stosuje się standard Gigabit Ethernet z przepustowością 1 Gb/s. Do specjalnych rozwiązań, takich jak rozległe sieci miejskie, stosuje się standardy o przepustowości 10, a nawet 100 Gb/s [16].

Ethernet pracujący w drugiej warstwie modelu OSI, czyli łączy danych, tworzy w procesie enkapsulacji własną ramkę ethernetową. Enkapsulację rozumie się jako przepływ danych przez warstwy na urządzeniu źródłowym, a także opatrywanie ich informacjami sterującymi, pozwalającymi zidentyfikować aplikację i urządzenie docelowe [11].

Pierwsze 8 bajtów ramki ma za zadanie poinformować urządzenie docelowe o nadchodzących danych. Kolejne 12 bajtów informuje o fizycznym adresie MAC nadawcy oraz adresata. Długość/typ jest częścią określającą wielkość ramki oraz to, jaki został wykorzystany protokół wyższej warstwy, np. IPv4. Dane są natomiast pakietem, który został odebrany z warstwy sieciowej, o minimalnej długości 46, natomiast maksymalnej – 1500 bajtów. Rozmiar całej ramki musi zawierać minimum 64 bajty, więc jeżeli pakiet danych jest mniejszy, wypełnia się go losowymi danymi. Ostatnie 4 bajty zawierają sumę kontrolną obliczaną i wpisaną do ramki przez urządzenie wysyłające, a potem weryfikowaną w urządzeniu odbierającym. Sumy takie muszą być równe, gdyż w przeciwnym wypadku ramka danych zostanie odrzucona i traktowana jako uszkodzona. Maksymalną wielkością ramki ethernetowej jest 1518 bajtów, w co nie wlicza się preambuły oraz sygnał początku ramki [16].

### 3. PROFINET – opis systemu

PROFINET jest uniwersalnym standardem bazującym na sieci Ethernet, który może być stosowany we wszystkich obszarach techniki automatyzacji. Rozwojem tego protokołu komunikacyjnego zajmuje się organizacja PROFIBUS&PROFINET International. Do budowania sieci PROFINET producent dostarcza dokładnej dokumentacji z zakresu wytycznych projektowych, instrukcji instalacji oraz wytycznych dotyczących uruchomienia. Dla sieci PROFINET prędkość transmisji danych jest stała i wynosi 100Mb/s. Sieć pracuje w trybie pełnego duplexu (ang. *full-duplex*) – oznacza to, że dane są przesyłane i odbierane jednocześnie, co powoduje zwiększenie przepustowości. Ogólną cechą sieci PROFINET jest sposób, w jaki wymieniane są dane pomiędzy urządzeniami w sieci. Każde urządzenie co określony, zaprogramowany czas (ang. *update time*), np. co 2 ms, wysyła do kontrolera swoją ramkę danych bez względu na to, czy w danym momencie te informacje są potrzebne, czy też nie. Analogicznie sterownik PLC będzie cyklicznie wysyłał do urządzeń peryferyjnych zadeklarowane dane. Jest to duża różnica w stosunku do sieci Profibus DP, w której urządzenia podrzędne (slave) zgłaszają swój stan dopiero po odpytaniu przez sterownik (Master). Przy wymianie danych ważny jest takt nadawania (ang.

*send clock*) – to okres między dwoma kolejnymi cyklami komunikacji, czyli najkrótszy możliwy odstęp w wymianie danych [3]. Cykl aktualizacji danych może być skonfigurowany dla każdego z urządzeń IO. Obliczone czasy aktualizacji to współczynniki redukcji (1, 2, 4, 8, ..., 512) taktu nadawania. Ten minimalny możliwy czas aktualizacji zależy od minimalnego zegara wysyłania kontrolera IO, którego można ustawić, oraz wydajność kontrolera IO i urządzenia IO [3]. Zaleca się, aby dane aktualizowane były z czasem połowy cyklu PLC. Najmniejszy możliwy do uzyskania czas aktualizacji danych w sieci PROFINET zależy od kilku czynników, są to między innymi [4]:

- liczba urządzeń PROFINET IO,
- wydajność sterownika IO i urządzenia IO,
- obciążenie sieci PROFINET IO,
- szerokość zarezerwowanego pasma,
- zegar nadawania serownika IO lub Sync mastera.

Ważne jest również dostosowanie czasu aktualizacji danych urządzenia indywidualnie do potrzeb aplikacji. Należy aktualizować dane wtedy, kiedy jest to konieczne, a nie tak często, jak to możliwe; zapobiega to nadmiernemu obciążaniu sieci i przedwczesnemu zużyciu się urządzeń sieciowych. Urządzenia w sieci PROFINET IO identyfikuje się z wykorzystaniem adresu MAC, adresu IP oraz nazwy stacji. Do utworzenia projektu konieczne jest zadeklarowanie unikalnego adresu IP w sieci oraz nazwy urządzenia, które są niezbędne do wgrania konfiguracji lub wymiany danych podczas pracy cyklicznej [5].

Nowe urządzenia sieci PROFINET IO integruje się przy użyciu plików GSDML (ang. General Station Description Markup Language) poprzez dodanie/zainstalowanie go do projektu PLC. Pliki te dostarczane są przez producenta. Opisane są w nich parametry niezbędne do poprawnej konfiguracji urządzenia, np. jego nazwa, nazwa producenta, liczba interfejsów, modułów, slotów wraz z ich parametrami oraz kody błędów generowane przez urządzenie. Struktura plików GSD w PROFINET IO ma format XML (GSDML) i jest zgodna ze światowym standardem opisu urządzeń ISO 15745 [5].

W sieci PROFINET IO można wyróżnić trzy typy urządzeń [1]:

- IO-Controller – najczęściej PLC lub w szczególnych przypadkach stacja PC jako nadrzędna jednostka sterująca obiektem lub procesem oraz zarządzająca cyklem sieci. Sterownik wymienia sygnały I/O z przydzielonymi urządzeniami peryferyjnymi,
- IO-Device – rozproszone urządzenie obiektowe/perferyjne, które jest przydzielone do jednego ze sterowników IO i udostępnia dla niego informacje lub służy jako urządzenie wykonawcze procesu sterowania. Przykładem takiego urządzenia może być rozproszona wyspa wejść/wyjść, falownik, wyspa zaworowa itp.
- IO-Supervisor – jest to najczęściej programator lub komputer PC jako urządzenie służące do uruchamiania i diagnostyki z odpowiednim oprogramowaniem.

### Parametry jakościowe sieci PROFINET

Ważnym parametrem jakościowym, od którego zależy stabilna praca sieci, jest *jitter* [4]. To różnica pomiędzy planowanym czasem dostarczenia pakietu a rzeczywistym

czasem aktualizacji. Parametr ten jest wprost powiązany z topologią i głębokością segmentacji sieci (różne czasy przekazywania pakietów przez linie przełączników, w zależności od aktualnego obciążenia sieci). Przykładowo, jeżeli dla urządzenia jest zadeklarowane, że aktualizacja danych ma odbywać się co 1 ms do sterownika PLC, a doszło do sytuacji, że sieć jest przeciążona np. przez duży ruch w pakietów TCP/IP, to może się okazać, że pakiety sieci PROFINET nie dotrą do sterownika w zadeklarowanym czasie i są opóźnione o 0.2 ms. W tym przypadku *jitter* wzrasta i wynosi 20%. Zalecaną maksymalną wartością tego parametru jest 50% i może on być ujemny, gdy pakiety odebrane zostaną zbyt wcześnie [6]. Opóźnienie jest również generowane w samych przełącznikach sieciowych, co także ma wpływ na zwiększenie parametru *jitter*, zwłaszcza przy rozległej topologii magistrali. Ważne, aby urządzenia przekazujące pakiety TCP/IP, których ramka zawiera większą liczbę bajtów, podłączone były do przełącznika znajdującego się najbliżej sterownika PLC, aby ograniczyć obciążanie sieci i blokowanie pakietów PN, a tym samym windowanie parametru *jitter* [6].

Na jakość w sieci PROFINET wpływają również przerwy w telegramach (ang. telegram *gap*). Brakujące telegramy są połączone bezpośrednio z funkcją Watchdog. Jeżeli w ciągu zadeklarowanych kolejnych cykli brakować będzie pakietu danych, alarmowany jest problem/zerwanie połączenia. Mechanizm ten można parametryzować w ustawieniach modułu IO [6].

Kolejnym parametrem jest obciążalność sieci (ang. *Network Load*) [4]. Jest to przepustowość danych na jednostkę czasu maksymalnie 100 Mb/s. Obciążalność określana jest dla miejsc, gdzie spodziewana jest największa koncentracja strumieni danych. Najczęściej jest to łącze sieciowe do sterownika PLC. Zaleca się, aby wartość obciążenia sieci wynosiła <20%. W skrajnych przypadkach gdy w sieci pojawia się nadmierna liczba pakietów danych, podłączenie programatora w trybie monitoringu może spowodować przeciążenie, destabilizację sieci i uniemożliwić pracę np. linii produkcyjnej [4].

Konfigurując i uruchamiając sieć przemysłową działającą na protokole transmisji PROFINET należy uwzględnić że opóźnienia komunikacji mogą wynikać z kilku różnych parametrów tj:

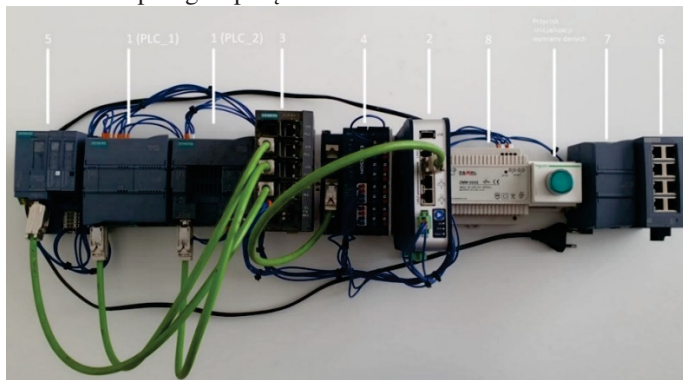
- Opóźnienie propagacji dla przewodu elektrycznego,
- Opóźnienie przetwarzania przełącznika sieciowego,
- Opóźnienie transmisji międzywęzłowej,
- Opóźnienie wynikające z czasu trwania pętli automatu sterowników PLC,
- Opóźnienie wynikające z czasu aktualizacji urządzeń PROFINET I/O.

#### **4. Pomiar czasu trwania wymiany danych w systemie komunikacyjnym działającym w protokole PROFINET**

W celu przeprowadzenia pomiarów czasu trwania wymiany danych w systemie komunikacyjnym zbudowanym na bazie protokołu PROFINET, złożono stanowisko pomiarowe składające się z kilku urządzeń pracujących w wyżej wymienionym protokole. W skład stanowiska wchodziły następujące elementy (Rys. 1):

- (1) Dwa sterowniki PLC Siemens, SIMATIC CPU 1212C DC/DC/DC,
- (2) Switch Siemens Scalance X208,

- (3) Wyspa Siemens ET200SP IM 155-6 PN wraz z modułami 16 DI oraz 16 DO,
- (4) Wyspa IO Emerson EPXPNS001 wraz z modułami 8 DI oraz 8 DO,
- (5) Wyspa Siemens ET200SP IM 155-6 PN wraz z modułami 16 DI oraz 16 DO,
- (6) Switch Siemens Scalance XB008,
- (7) Switch Siemens CSM 1277,
- (8) Zasilacz Zamel ZMM-24/42,
- (9) Komputer PC z oprogramowaniem TIA Portal V15.1 do sterowników rodziny Siemens Simatic, PAC Machine Edition 9.80 SIM 9 do programowania sterowników Emerson. Oprogramowanie WireShark V.3.6.1 do analizy ramki danych PROFINET. Program PC Proneta 3.3.0 wykorzystany w pracy do rozpoznawania topologii i połączeń w sieci PROFINET.



Rysunek 1. Stanowisko pomiarowe do badań czasu trwania transakcji wymiany danych w protokole PROFINET

Scenariusz badań na stanowisku pomiarowym obejmował:

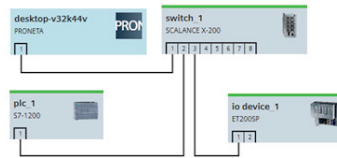
- automatyczny pomiar czasu cyklu sterownika PLC w celu określenia czasu trwania cyklu automatu,
- wymuszanie obciążania jednostki centralnej algorytmem obciążającym w celu zwiększenia czasu cyklu,
- weryfikacja wpływu długości wykonywania pętli na czas wymiany danych,
- weryfikacja wpływu parametru *update time* na czasy wymiany danych.

Pomiary i obliczenia zapisywane były w jednostce centralnej PLC. Aby uzyskane wyniki pomiarów były miarodajne, przy każdej serii badań wykonano pomiar czasu trwania pętli sterownika. W celu zminimalizowania ograniczeń wynikających z systematyczności wykonywanych zadań cyklu sterownika, inicjalizacja wymiany danych wykonana została przy użyciu funkcji przerwań. Zabieg ten pozwolił na zredukowanie różnic w czasach pomiędzy inicjalizacją wymiany a reakcją jednostki centralnej. Oczekiwany wynikiem było uzyskanie czasu, którego składowe stanowiły wszystkie czynniki wpływające na opóźnienie w wymianie danych [9].

W dalszej kolejności mierzono czas trwania dwustronnej wymiany danych (Rys. 2), rozumianej jako taką, której pomiar czasu mierzony jest od momentu wywołania transakcji do zarejestrowania odpowiedzi w sterowniku wywołającym wymianę. Każde z przeprowadzonych badań, niezależnie od topologii i sposobu połączenia urządzeń, bazowało na 100 wykonanych pomiarach czasu wymiany danych oraz 20



miarach czasu cyklu sterownika lub sterowników. W ten sposób zebrane dane dostarczyły informacji na temat minimalnych, maksymalnych oraz średnich czasów. Bazując na uśrednionych pomiarach, można było porównać poszczególne wyniki badań oraz wyciągnąć wnioski wynikające z różnych ustawień parametrów i konfiguracji połączeń.



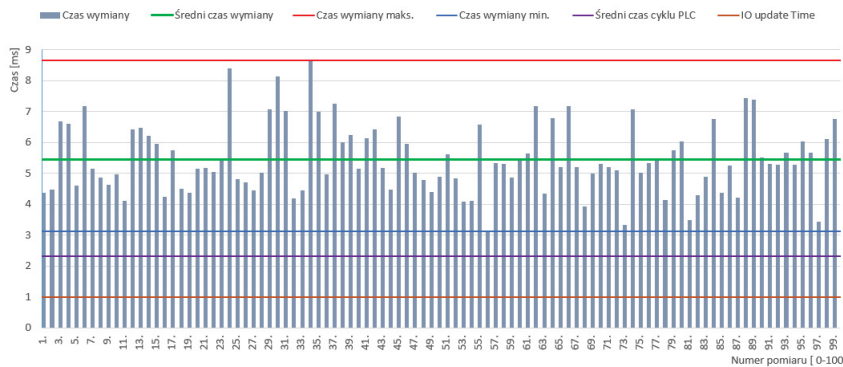
#	Name	Device Type	IP Address	Subnet Mask	MAC Address	Role	Vendor Name	Order Number	Firmware Version	Hardware Revision
1	switch_1	SCALANCE X-200	192.168.0.7	255.255.255.0	00-0e-8c-9b-85-d8	Device	SIEMENS AG	6GKS 208-0BA10-2AA3	V5.2.1	2
2	io device_1	ET200SP	192.168.0.6	255.255.255.0	ec:1c:5d:13:70:a9	Device	SIEMENS AG	6ES7 155-6AR00-0AN0	V3.2.2	5
3	plc_1	S7-1200	192.168.0.1	255.255.255.0	ac:64:17:0a:d7:d9	Controller	SIEMENS AG	6ES7 212-1AE40-0XB0	V4.2.1	6

Rysunek 2. Topologia stanowiska dla pomiaru czasu dwustronnej wymiany danych pomiędzy sterownikiem PLC oraz wyspą rozproszoną I/O

### 5. Wyniki badań

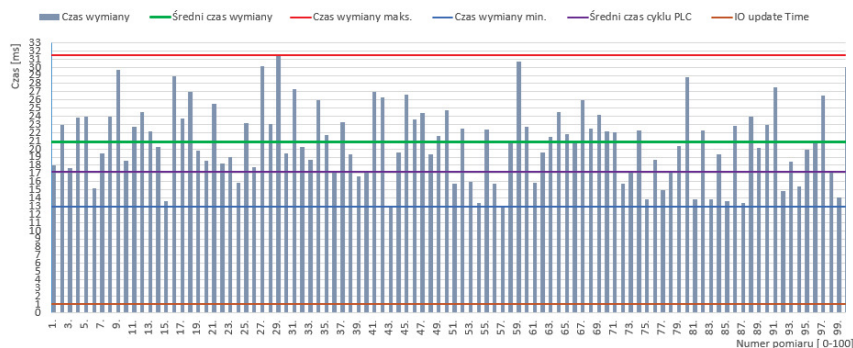
Badania dla poszczególnych konfiguracji zostały przeprowadzone na stanowisku badawczym według przyjętych w poprzednim rozdziale założeń. Poniżej przedstawiono wyniki czasu trwania wymiany danych dla poszczególnych wariantów:

- Wariant 1, wymiana dwóch Bajtów danych pomiędzy sterownikiem IO-Controllerem S7-1212 oraz wyspą IO-Device ET200SP przy nieobciążonym sterowniku



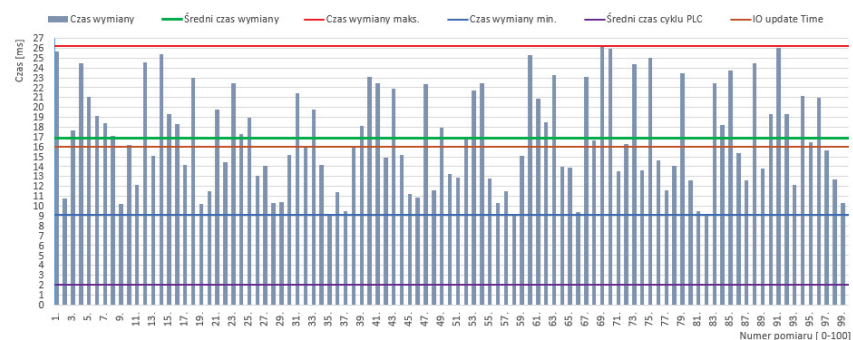
Rysunek 3. Wykres przedstawiający zestawienie wyników pomiarów dla wymiany 2B danych, przy dwustronnej komunikacji sieciowej z wyspą ET200SP przy nieobciążonym sterowniku.

- Wariant 2, Wymiana dwóch Bajtów danych pomiędzy sterownikiem IO-Controllerem S7-1212 oraz wyspą IO-Device ET200SP przy obciążonym sterowniku



Rysunek 4. Wykres przedstawiający zestawienie wyników pomiarów dla wymiany 2B danych przy dwustronnej komunikacji sieciowej z wyspą ET200SP przy obciążonym sterowniku

- Wariant 3, Wymiana 2B danych pomiędzy sterownikiem IO-Controllerem S7-1212 oraz wyspą IO-Device ET200SP przy nieobciążonym sterowniku oraz update time 16 ms



Rysunek 5. Wykres przedstawiający zestawienie wyników pomiarów dla wymiany 2B danych przy dwustronnej komunikacji sieciowej z wyspą ET200SP przy nieobciążonym sterowniku oraz update time 16 ms

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	Siemens_0a:d7:d9	Siemens_19:94:13	PNIO	60	RTCL, ID:0x8000, Len: 40, Cycle:19968 (Valid,Primary,Ok,Run)
2	0.007105	Siemens_19:94:13	Siemens_0a:d7:d9	PNIO	60	RTCL, ID:0x8001, Len: 40, Cycle:23040 (Valid,Primary,Ok,Run)
3	0.016023	Siemens_0a:d7:d9	Siemens_19:94:13	PNIO	60	RTCL, ID:0x8000, Len: 40, Cycle:20480 (Valid,Primary,Ok,Run)
4	0.016993	Siemens_0a:d7:d9	Siemens_9b:85:d6	PNIO	60	RTCL, ID:0x8000, Len: 40, Cycle:20512 (Valid,Primary,Ok,Run)
5	0.023333	Siemens_19:94:13	Siemens_0a:d7:d9	PNIO	60	RTCL, ID:0x8001, Len: 40, Cycle:23552 (Valid,Primary,Ok,Run)
6	0.032021	Siemens_0a:d7:d9	Siemens_19:94:13	PNIO	60	RTCL, ID:0x8000, Len: 40, Cycle:20992 (Valid,Primary,Ok,Run)
7	0.039074	Siemens_19:94:13	Siemens_0a:d7:d9	PNIO	60	RTCL, ID:0x8001, Len: 40, Cycle:24064 (Valid,Primary,Ok,Run)
8	0.048027	Siemens_0a:d7:d9	Siemens_19:94:13	PNIO	60	RTCL, ID:0x8000, Len: 40, Cycle:21504 (Valid,Primary,Ok,Run)
9	0.055098	Siemens_19:94:13	Siemens_0a:d7:d9	PNIO	60	RTCL, ID:0x8001, Len: 40, Cycle:24576 (Valid,Primary,Ok,Run)
10	0.064025	Siemens_0a:d7:d9	Siemens_19:94:13	PNIO	60	RTCL, ID:0x8000, Len: 40, Cycle:22016 (Valid,Primary,Ok,Run)

Rysunek 6. Weryfikacja parametru update time w programie WireShark

Na rysunku 6 pokazany został zarejestrowany wpływ parametru aktualizacji danych, ustawionego na 16 ms. Można zauważyć, że pomiędzy pierwszą a trzecią wysłaną ramką nadaną przez PLC\_1 (0a:d7:d9) do PLC\_2 (19:94:13) czas wynosi około 16 ms. Taka sama różnica czasu została zarejestrowana dla każdej następnego ramki danych.



Wszystkie przeprowadzone badania, miały na celu weryfikację wpływu różnych parametrów na czas wymiany danych. Każde z nich z góry obarczone było błędem spowodowanym brakiem synchronizacji cyklu urządzeń sieciowych, co jest cechą systemów rozproszonych w których cykl pracy poszczególnych sterowników jest asynchroniczny. Brak synchronizacji wprowadza znaczne różnice czasu pomiędzy pomiarami, liczone nawet w niektórych przypadkach kilka lub kilkanaście milisekund. Z tego względu pod uwagę brano nie tylko uśrednione wyniki. Uśrednione wartości nie dają całego obrazu systemu, dlatego szukano najgorszego zmierzonego parametru (maksymalny czas wymiany danych), który określa determinizm czasowy i jest nieprzekraczalnym czasem realizacji przesyłania komunikatu.

Przygotowane stanowiska laboratoryjne dostarczyły danych, z których wnioskować można, w jakim stopniu dany parametr lub konfiguracja urządzeń wpływały na czas dostarczenia lub/i odbioru informacji pomiędzy urządzeniami.

Parametrami, które badano były: zależność czasu cyklu sterownika oraz zależność parametru *update time* na czas odpowiedzi (Tabela 1).

Tabela 1. Zależność czasu cyklu sterownika od parametru *update time* na czas odpowiedzi

Lp.	Nazwa parametru	W1	W2	W3	Jednostka
1.	Średni czas cyklu PLC	2,313	17,16	2,084	[ms]
2.	Minimalny czas cyklu PLC	1,53	16,267	1,367	[ms]
3.	Maksymalny czas cyklu PLC	2,828	17,76	2,392	[ms]
4.	Liczba przesyłanych danych	2	2	2	B
5.	IO-Device Update Time	1	1	16	[ms]
6.	Średni czas wymiany danych	5,447	20,865	16,926	[ms]
7.	Minimalny czas wymiany danych	3,108	12,896	9,083	[ms]
8.	Maksymalny czas wymiany danych	8,654	31,538	26,211	[ms]
9.	Liczba przełączników/urządzeń (segmentacja)	brak	brak	brak	-

Pierwsza seria pomiarów wykonana została na nieobciążonym sterowniku o minimalnej długości cyklu, który wyniósł średnio 2,313 ms dla Siemens S7-1212. W dalszym etapie badania obciążono sterownik obliczeniami zmiennoprzecinkowymi, aby uzyskać wydłużony czas cyklu, który wyniósł średnio 17,160 ms dla sterownika Siemens, co dało czas wymiany danych średnio 20,865 ms, maksymalnie 31,538 ms. Kolejnym krokiem była zmiana parametru *update time* na 16 ms dla wyspy ET200SP, co dało średni czas wymiany danych na poziomie 16,926 ms, maksymalnie 26,211 ms.

## 6. Podsumowanie i wnioski

Na podstawie przeprowadzonych badań można zauważyć, że wyżej opisane parametry mają duży wpływ na uzyskane czasy odpowiedzi. Czas aktualizacji danych jest jednym z parametrów, który w celu uniknięcia zbyt nadmiernego obciążenia sieci jest konfigurowany już na etapie jej projektowania, a co za tym idzie, projektant sam ustala jego wartość względem potrzeb aplikacji. Z przedstawionych badań wynika, że dla *update time* równego 16 ms i zwiększonego przez algorytm czasu trwania cyklu automatu, czas trwania wymiany danych był bardzo nieregularny. Na (Rys. 5) (*update*

*time* 16 ms) widać, że poszczególne zmierzone czasy wymian oscylują pomiędzy 9,083 a 26,211 ms., wynika z tego, że dla przedstawionych badań, czas trwania cyklu automatu a w szczególności moment wywołania zewnętrznego zdarzenia ma duże znaczenie na czas trwania wymiany. Powyższe wnioski podtrzymują tezy z [9], że czas trwania cyklu automatu niezależnie od zastosowanego systemu komunikacyjnego ma wpływ na czas trwania wymiany komunikatów i są nadal aktualne mimo pojawienia się nowych rozwiązań na rynku automatyki.

## LITERATURA

1. MYSTKOWSKI A.: Sieci przemysłowe PROFIBUS DP i PROFINET IO, Białystok 2012.
2. Sieci komputerowe – Ethernet, <https://pasja-informatyki.pl/sieci-komputerowe/ethernet/>, 15.12.2021.
3. Siemens Simatic PROFINET with STEP 7 V14 , A5E03444486-AG, September 2016.
4. PROFINET Commissioning. Guideline for PROFINET, 8082 V1.44, September 2019.
5. Siemens Simatic PROFINET, A5E00298288-04, Podręcznik wydanie I, Warszawa 2009.
6. PROFINET Safety for the future, [https://www.induol.com/fileadmin/user\\_upload/produkte/Indu-Sol-PROFINET-brochure-2020.pdf](https://www.induol.com/fileadmin/user_upload/produkte/Indu-Sol-PROFINET-brochure-2020.pdf), 10.01.2022.
7. Creating GSD (GSDML) Files, [https://rt-labs.com/docs/p-net/creating\\_gsdml\\_files.html](https://rt-labs.com/docs/p-net/creating_gsdml_files.html), 02.02.2022.
8. KUROS J., ROSS K.: Sieci komputerowe Ujęcie całościowe wydanie VII, Helion 2018, 1.4–1.4.4.
9. SIDZINA M., Dynamiczne modyfikacje programu aplikacji sterownika swobodnie programowanego celem zwiększenia częstotliwości wymian komunikatów w przemysłowych systemach rozproszonych Czasu Rzeczywistego, Rozprawa Doktorska, Politechnika Śląska w Gliwicach, Gliwice 2008, 1, 7, 8.3–8.4.
10. Siemens Simatic NET Scalance Industrial Ethernet X-100 and X-200 Product Line – Operating instructions, A5E00349864, September 2006.
11. Komunikacja w systemach automatyki przemysłowej – Ethernet vs fieldbus, <https://automatykab2b.pl/temat-miesiaca/49197-komunikacja-w-systemach-automatyki-przemyslowej-ethernet-vs-fieldbus>, 10.10.2021.
12. PROFINET – podstawy, teoria i praktyka, <https://iautomatyka.pl/profinet-io-podstawy-teoria-i-praktyka/>, 11.10.2021.
13. PROFINET Design Guideline, 8062 V.1.14, December 2014.
14. Siemens S7-1200 Programmable controller – System Manual, A5E02486680-06, April 2012.
15. Czas aktualizacji urządzeń w sieci PROFIBUS, <https://www.diagnozyka-profibus.pl/czas-aktualizacji-urzadzen-w-sieci-profinet/> 21.12.2021.
16. IEEE 802.3 Standard.
17. STÓJ. J.: Cost-effective hot-standby redundancy with synchronization using EtherCAT and real-time Ethernet protocols, IEEE Trans. Autom. Sci. Eng, 2021 vol. 18 iss. 4 s. 2035-2047, 2021.