

PRZEMYSŁOWY INSTYTUT AUTOMATYKI I POMIARÓW  
MERA-PIAP

Al. Jerozolimskie 202

02-222 Warszawa

Telefon 23-70-81

Ośrodek Automatyki Elektrycznej

BE 10

440

Pracownia Oprogramowania Wieloprocessorowych Systemów  
Automatyki

Główny wykonawca mgr inż. Andrzej Aderek

Wykonawcy mgr inż. Krzysztof Czarnomski

Konsultant dr inż. Andrzej Syrczyński, mgr inż. Mirosław Słodczyk

Nr zlecenia  
9436 Etap 2.

Analiza możliwości realizacji protokołów  
MIR-PROWAY na dwuprocessorowym  
układzie 8080-8035 oraz na procesorze  
8086.  
Etap 2. Analiza możliwości realizacji  
protokołów MIR-PROWAY na procesorze  
8086.

Zleceniodawca praca własna.

Pracę rozpoczęto dnia 1.10.83

zakończono dnia 31.12.83

Kierownik Pracowni

p.o.Z-oy Dyrektora  
d/s Automatyki

Kierownik Ośrodka

mgr inż. A. Aderek

dr inż. T. Gałązka

prof. dr inż. T. Miśsała

Praca zawiera:

Rozdzielnik - ilość egz:

stron

25

Egz. 1 BOINTE

rysunków

Egz. 2 OAE-8

fotografii

Egz. 3 OBN

tabel

Egz. 4 OAE-83

tablic

Egz. 5

załączników

Egz. 6

Nr rejestr. 5181

## Analiza deskryptorowa

OPROGRAMOWANIE + MIKROPROCESOR + MIR-PROWAY.

## Analiza dokumentacyjna

Opracowanie zawiera założenia kontrolera komunikacyjnego stacji PROWAY zbudowanego na procesorze 8086 i opis programu realizującego protokół PROWAY. Obliczono i zmierzono czas wykonania programu i porównano go z limitem czasu określonym przez protokół.

## Tytuły poprzednich sprawozdań

681.3.06 Oprogramowanie

681.55:621.372.-181.48 Mikroprocesory

UKD

MERA-PIAP/TW 331/78 5000

SPIS TREŚCI

1. Wstęp.....	2
2. Określenie kryteriów analizy możliwości realizacji protokółów MIR-PROWAY.....	2
3. Wymagania funkcjonalne na kontroler komunikacyjny.....	5
3.1. Miejsce kontrolera komunikacyjnego w systemie PROWAY.....	5
3.2. Schemat funkcjonalny kontrolera magistrali.....	7
3.3. Komunikacja między użytkownikiem a kontrolerem.....	9
3.4. Komunikacja między kontrolerem a użytkownikiem.....	10
3.5. Komunikacja między układem sprzęgającym a kontrolerem magistrali.....	12
3.6. Komunikacja między kontrolerem magistrali a układem sprzęgającym.....	12
4. Opis i schemat blokowy programu realizującego krytyczny fragment protokołu PROWAY.....	13
5. Listing programu.....	21
6. Określenie czasu wykonania programu.....	23
7. Podsumowanie.....	24
8. Literatura.....	25

## 1. Wstęp

Niniejsze sprawozdanie jest kolejnym etapem przygotowań do skonstruowania i wdrożenia systemu wymiany informacji w rozłożonych systemach sterowania. W wyniku poprzednio wykonanych prac [1,2] stwierdzono, że nie jest możliwe wykonanie oprogramowania realizującego protokół systemu PROWAY ani na mikroprocesorze INTEL 8080, ani na dwuprocessorowym układzie INTEL 8080 - INTEL80386. W pierwszym wypadku czasy wykonania programu przekraczałyby limity stawiane przez projekt normy PROWAY-a. W drugim - przy wykorzystaniu układu dwuprocessorowego - można zrealizować działania przewidziane w protokole PROWAY przy szybkości transmisji 100 kB/s. Jednak wobec dużego obciążenia układu, teoretycznie znacznie przekraczającego 50% jego maksymalnych możliwości, wynik pracy nie jest zadowalający. Przede wszystkim kontroler z omawianym układem byłby nie rozwojowy - nie dawałby możliwości zwiększenia szybkości transmisji i istotnego skrócenia przerw w linii między transmisjami. Celem niniejszego etapu jest przeanalizowanie możliwości wykonania wymienionego wyżej oprogramowania na mikroprocesorze INTEL 8086. Założenia dotyczące pracy kontrolera komunikacyjnego realizującego protokół PROWAY zostały opracowane na podstawie dokumentów Międzynarodowej Komisji Elektrotechnicznej /IEC/, Podkomitet SC 65C, wydanych pod wspólnym tytułem "Draft-Process Data Highway /PROWAY/ form distributed process control systems".

## 2. Określenie kryteriów analizy możliwości realizacji protokołów

### MIR-PROWAY

Protokół kontrolera komunikacyjnego stacji MIR-PROWAY zawiera schematy blokowe działań wykonywanych przez kontroler w trakcie pracy magistrali. Są w nim również określone limity czasu przeznaczone na realizację tych działań, których czas wykonania nie zależy wprost od przyjętej szybkości transmisji danych. Rozważana tu struktura sprzętowa kontrolera - mikroprocesor INTEL8086, pamięci operacyjne i układy towarzyszące - daje praktycznie nieograniczone możliwości kombinacji jego działań, zależnych od zastosowanego oprogramowania.

Stąd jedynymi ograniczeniami możliwości realizacji protokołów MIR-PROWAY na wymienionym sprzęcie są ograniczenia czasowe. Protokoły te stawiają w tej dziedzinie wysokie wymagania. Stosunkowo skomplikowane działania muszą być wykonywane w krótkich odcinkach czasu, odwrotnie proporcjonalnych do dużej szybkości transmisji.

Ostateczna odpowiedź na pytanie, czy dany sprzęt może spełniać te wymagania byłaby możliwa po zbudowaniu, uruchomieniu i przebadaniu urządzenia prototypowego. W miarę pewną odpowiedź możnaby uzyskać po napisaniu całości oprogramowania kontrolera komunikacyjnego i określeniu czasów wykonywania przezeń wszystkich działań protokołów PROWAY-a - bądź przez obliczenie teoretyczne, na podstawie podanych przez producenta czasów wykonania rozkazów mikroprocesora, bądź przez pomiar doświadczalny po uruchomieniu oprogramowania.

Oba sposoby analizy przekraczają zakres mniejszej pracy. Stąd daje ona jedynie odpowiedź szacunkową. Z protokołu kontrolera komunikacyjnego stacji PROWAY wybrano krytyczny pod względem wymagań czasowych fragment, napisano program realizujący go i określono /teoretycznie i doświadczalnie/ czas jego wykonania.

Podstawowe wymagania czasowe stawiane przez protokół kontrolera komunikacyjnego to:

- 1/ wysłanie odpowiedzi /potwierdzenia/ na przesyłkę typu Send/Confirm, Recovery lub Request/Response przed upływem dopuszczalnego czasu przerwy w linii,
- 2/ powrót do stanu jałowego po zakończeniu obsługi przesyłki, tak by kontroler był gotów na odbiór kolejnej przesyłki przed upływem połowy dopuszczalnego czasu przerwy w linii,
- 3/ rozpoczęcie nadawania po otrzymaniu sterowania - w wypadku gdy użytkownik zgłasza zgodnie wykonanie funkcji typu pytanie - odpowiedź /z potwierdzeniem lub odpowiedzią/  
- w wypadku, gdy użytkownik takiego ządania nie zgłasza, przed upływem dopuszczalnego czasu przerwy w linii.

W powyższych przypadkach protokół kontrolera komunikacyjnego przewiduje wykonanie następujących czynności:

- a) jeżeli w buforze "Hold" znajduje się informacja i jeżeli odebrana ramka ma inny nagłówek od zatrzymanego w "hold" zwolnienie informacji z "hold" dla użytkownika,
- b) kontrola typu ramki,

- dla funkcji typu Send/Confirm
  - c/ kontrola zapamiętania przesłanej informacji /czy był dla niej przygotowany bufor/,
  - d/ zatrzymanie informacji w buforze "Hold",
  - e/ transmisja potwierdzenia /pozytywnego lub negatywnego/
  - dla funkcji typu Recovery
  - f/ zadanie jednego z trzech sprzętowych sygnałów "recovery"
  - g/ transmisje potwierdzenia,
- ad.2/ a/ dla przesyłek z odpowiedzią:
- dla stacji wysyłającej odpowiedź-powrót do stanu "Listen"
  - dla stacji nie uczestniczącej w wymianie przesyłki, działania według punktów 1a-b i powrót do stanu "Listen"
- b/ dla przesyłek bez odpowiedzi:
- punkty 1a-b
  - aktualizacja Live List
  - przekazanie informacji do użytkownika /jeżeli informacja została przyjęta przez kontroler, co następuje jeżeli był dostępny dla zapisania jej bufor/,
  - sprawdzenie, czy baton do mnie
  - powrót do stanu "Listen" /omawiany jest przypadek "baton nie do mnie"/,
- ad 3/ - punkty 1a-b
- aktualizacja Live List
  - przekazanie informacji do użytkownika /jeżeli informacja została przyjęta przez kontroler/
  - sprawdzenie czy baton do mnie
  - na podstawie zaktualizowanej Live List obliczenie Next Live Address i Next Gap Address,
  - sygnalizacja ew. zmiany Live List do użytkownika
  - kontrola zleceń użytkownika
- a/ użytkownik żąda wykonania funkcji typu Send Data with Acknowledge, Recovery lub Request Data:
- wyzerowanie licznika powtórzeń funkcji
  - transmisja ramki z żadaną przesyłką,
- b/ użytkownik żąda wykonania funkcji typu Global Send Data lub nie zgłasza żądań

- kontrole obecności szczeliny /Gap/ pomiędzy adresem mojej stacji a Next Live Address,
- transmisja ramki z przesyłką typu Global Send Data lub typu baton do Next Gap Address /gdy jest szczelina/ lub do Next Live Address.

Na podstawie oszacowania relatywnych czasów wykonania wymienionych wyżej czynności określono przypadek stawiający przed kontrolerem komunikacyjnym najwyższe wymagania czasowe. Jest to przypadek 3b, tzn. przekazanie batonu /sterowania/ do następnej stacji wraz z ewentualną przesyłką typu Global Send Data bezpośrednio po otrzymaniu sterowania od jednej ze stacji poprzednich, co musi nastąpić w czasie nie przekraczającym maksymalnego okresu przerw w linii, to jest 100  $\mu$ sek.

Określenie czasu wykonania programu realizującego wybrany fragment protokołu PROWAY i porównanie go z czasem, jaki protokół ten daje do dyspozycji na jego realizację jest podstawowym kryterium analizy, będącej tematem niniejszej pracy.

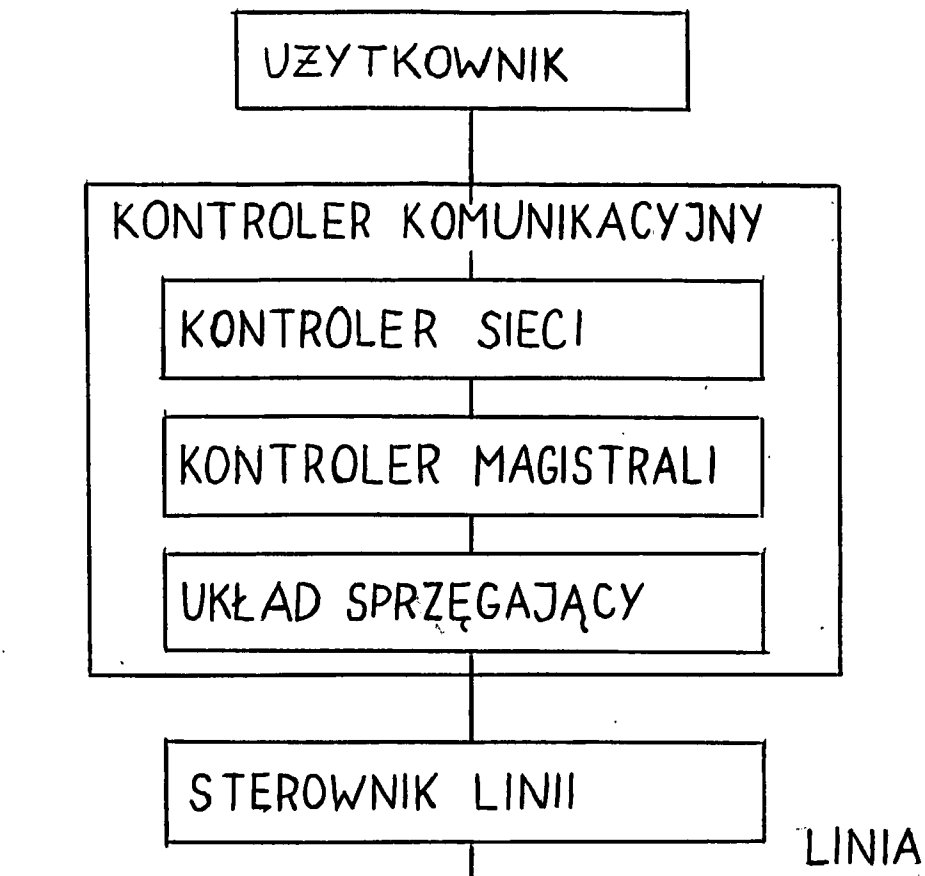
### 3. Wymagania funkcjonalne kontrolera komunikacyjnego.

Ogólne założenia dotyczące zasady działania kontrolera komunikacyjnego zostały zaproponowane w dokumentach JEC. Dla zachowania uniwersalności projektu w dokumentach tych pominięto wszelkie założenia wiążące się ze sprzętem, a rozwiązania szczegółowe pozostawiono twórcom konkretnych implementacji.

W celu przeanalizowania możliwości realizacji kontrolera na bazie mikroprocesora INTEL 8086, według zasad podanych w poprzednim rozdziale, konieczne było uszczegółowienie opisu działania kontrolera i jego współpracy z otoczeniem przy uwzględnieniu wymagań stawianych przez sprzęt.

#### 3.1. Miejsce kontrolera komunikacyjnego w systemie PROWAY

Kontroler komunikacyjny jest jednym z elementów wchodzących w skład stacji systemu PROWAY. Pozostałymi elementami są użytkownik i sterownik linii, rys.1. Użytkownik jest elementem nadrzędnym w stosunku do pozostałej części stacji, którą traktuje jak urządzenie wejścia/wyjścia. Współpraca użytkownika z resztą stacji odbywa się przez przekazywanie poleceń zrealizowanie pewnych czynności i sygnalizację ich wykonania lub zaistnienie sytuacji awaryjnych.



Rys.1. Schemat funkcjonalny stacji systemu PROWAY.

Użytkownik bezpośrednio współpracuje z kontrolerem komunikacyjnym który realizuje protokół PROWAY-a. Kontroler komunikuje się z najniższą warstwą stacji - to znaczy ze sterownikiem linii. Zadaniem sterownika jest dopasowanie elektryczne stacji do linii, nadawanie i odbiór sygnału z linii.

W kontrolerze komunikacyjnym wyróżnia się trzy elementy: kontroler sieci, kontroler magistrali i układ sprzęgający. Kontroler sieci jest układem realizującym najwyższy poziom protokołu PROWAY-a - poziom zarządzania. Układ ten będzie w stacji PROWAY występował opcjonalnie - tylko w stacjach będących zarządcami sieci tzn. systemów PROWAY składających się z więcej niż z jednej linii. Będzie on stanowił dodatkową jednostkę centralną w kasie stacji. Dwie pozostałe części będą się znajdowały w każdym kontrolerze komunikacyjnym. Będą one umieszczone na jednym pakiecie stanowiącym standardową postać kontrolera. Kontroler magistrali będzie realizował protokół magistrali - zasadniczy element protokołu PROWAY obejmujący poziomy: odbierania, odpowiadania, inicjacji, sterowania i nadzoru. Protokół magistrali określa sposób współpracy stacji 8 dołączonych do linii - rodzaje przesyłek pomiędzy stacjami i metodę sprawowania nadzoru nad linią - kolejność dostępu stacji do linii.



W danym momencie do linii ma dostęp /możliwość nadawania/ tylko jedna stacja - przez co unika się zakłóceń w łączności pomiędzy stacjami. Układ sprzęgający rozpoznaje adres stacji, wykonuje konwersję szeregowo-równoległą informacji do/z linii /sterownika linii/, informację odebraną zapisuje, a nadawaną odczytuje z pamięci dostępnej dla użytkownika /aby uniknąć przepisywania/, generuje i kontroluje kody kontrolne, wykrywa błędy synchronizacji i rozmiarów ramki. Kontroler magistrali i układ sprzęgający będą układami mikroprocesorowymi, każdy z nich będzie miał własną mikroprocesorową jednostkę centralną. Przedmiotem badań niniejszej pracy jest tylko kontroler magistrali. Jeżeli w innych jej rozdziałach mówi się o kontrolerze komunikacyjnym odnosi się to zawsze do tej części, która realizuje protokół magistrali.

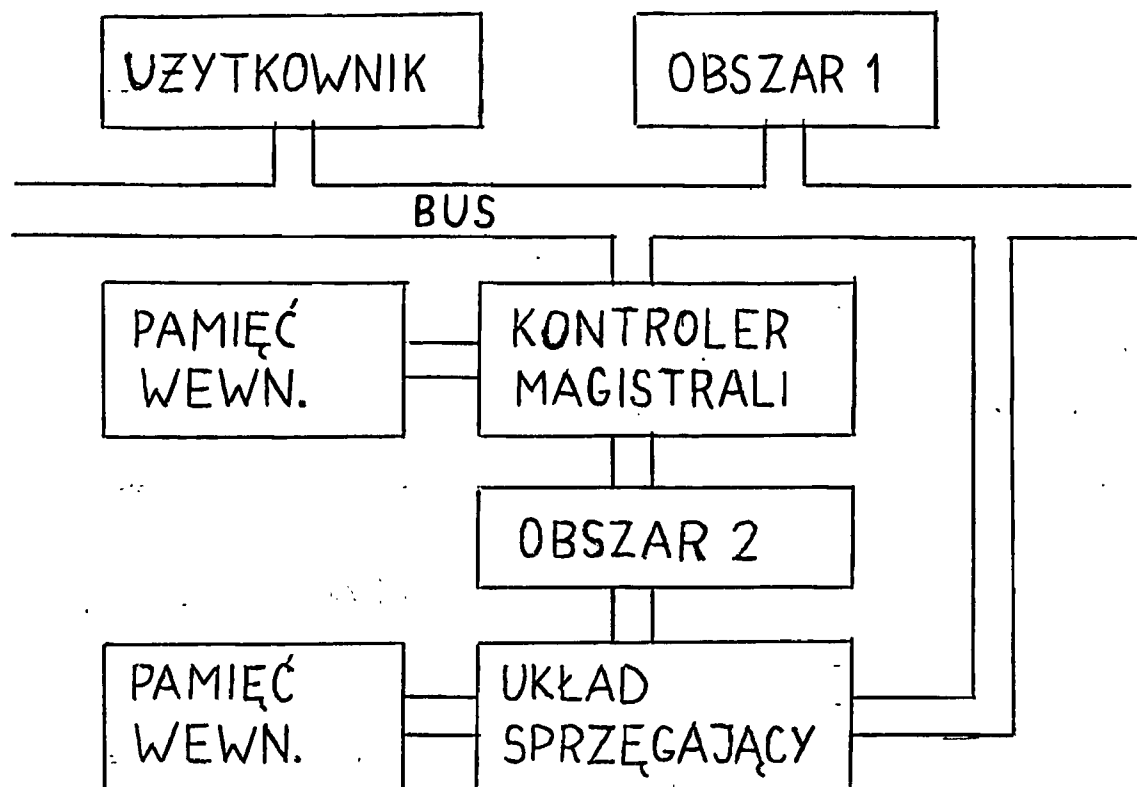
### 3.2. Schemat funkcjonalny kontrolera magistrali

Jak wspomniano na początku rozdziału, do wykonania niniejszej pracy konieczne było sprecyzowanie, w stosunku do dokumentów JEC, opisu działania kontrolera. Potrzebne było określenie sposobu komunikacji kontrolera magistrali z otoczeniem - czyli z układem sprzęgającym, z jednej strony, a użytkownikiem bądź kontrolerem sieci z drugiej strony.

Komunikacja ta odbywa się poprzez wymianę informacji we wspólnych obszarach pamięci, a także przez wysyłanie przerw sprzętowych. Pamięć operacyjna kontrolera magistrali jest podzielona i jej poszczególne obszary są dostępne dla różnych urządzeń. Schemat logiczny tego podziału przedstawia rys.2.

Poszczególne obszary zawierają bufor danych, semafony i wskaźniki. "Obszar: 1" jest dostępny poprzez magistralę kasety PROWAY dla kontrolera magistrali, użytkownika i układu sprzęgającego. Znajdują się w nim:

- bufor zlecenia funkcji Management /BZFM/ - 7 byte,
- semafor bufora BZFM /SBZFM/ - 1 byte;
- semafor bufora BZFR /SBZFR/ - 1 byte;
- semafor bufora BZFG /SBZFG/ - 1 byte,
- pole specyfikacji zdarzenia /PSZ/ - 1 byte,
- bufor na dane przychodzące do stacji /BDP/ - do 1024 byte,
- bufor z danymi do wysłania funkcją Send/Confirm /BFSC/ -  
- do 1024 byte,
- bufor z danymi do wysłania funkcją Global Send Data /BDFG/ -  
- do 1024 byte,



Rys.2. Schemat wymiany informacji między kontrolerem magistrali a otoczeniem przez wspólne obszary pamięci.

- bufory z danymi pozwalającymi na zrealizowanie funkcji Request/Response przychodzących z innych stacji BDFR - do 126 x 1027 byte,
  - bufor na zlecenie wykonania funkcji różnych od Management i Global Send Data /BZFR/ - 7 byte,
  - pole z adresem bufora na dane przychodzące do stacji /BDP/ - 3 byte,
  - bufor na zlecenie wykonania funkcji Global Send Data /BZFG/ - 1 byte,
  - bufor odpowiedzi na zadaną funkcję /BO/ - 7 byte,
  - bufor odpowiedzi dla funkcji Management /BOFM/ - 19 byte,
  - lista adresów buforów BDFR-/LBDRF/ - 3 x 127 byte,
- obszar 2 jest dostępny dla kontrolera magistrali i układu sprzęgającego. Zawiera on:
- bufor na nagłówek ramki odbieranej /NAG/ - 3 byte,
  - słowo stanu /SS/ - 1 byte,
  - licznik bajtów aktualnie ładowanej ramki /LBAR/ - 2 byte,
  - pole z adresem bufora z funkcją do wysłania /ABF/ - 2 byte,
  - bufor na zlecenie wysłania ramki typu Baton /BRB/ - 3 byte.

Oprócz wymiany informacji z otoczeniem przez wspólne obszary pamięci kontroler odbiera i przesyła informacje za pomocą przerwań sprzętowych. Ze względu na długi czas /ok. 30  $\mu$ s/ jaki zużywa INTEL 8086 na sprzętową obsługę przerwania ograniczono do minimum ten sposób wymiany informacji. Przyjęto również że kontroler magistrali odbiera przerwania tylko z układu sprzęgającego a sam może wysyłać sygnał przerwania tylko do użytkownika. Okoliczności, w których następuje wymiana przerwań opisane są w dalszych rozdziałach tego opracowania.

### 3.3. Komunikacja między użytkownikiem a kontrolerem

Kontroler komunikacyjny jest urządzeniem podzależnym w stosunku do użytkownika. Oznacza to, że kontroler wykonuje zlecenia użytkownika oraz informuje go o zdarzeniach związanych z wymianą informacji w linii.

Użytkownik przekazuje zlecenia do kontrolera poprzez bufory znajdujące się w obszarze 1. Pierwszy bajt w buforze oznacza typ zlecanej funkcji przy czym wartość zerowa tego bajtu oznacza brak zleceń. Kolejne bajty buforów zawierają informacje zapisane zgodnie z dokumentami PROWAY-a. Wszystkie bufory zleceń są semaforowane. Użytkownik może dokonywać zmian w buforze wtedy, gdy typ funkcji jest zero lub gdy semafor jest otwarty. Przed wykonaniem zapisu lub zmiany w buforze użytkownik zamyka semafor a po zakończeniu tych czynności otwiera go. Kontroler przed wykorzystaniem informacji zawartej w buforze sprawdza czy semafor bufora jest otwarty, i jeśli tak to zamyka go. Otwarcie semafora następuje po wykonaniu wszystkich czynności związanych ze zleceniem znajdującym się w buforze. Równocześnie zerowany jest bajt typu zlecenia.

Po otrzymaniu sterowania kontroler analizuje zawartość bufora BZFR i podejmuje realizację zapisanych tam przez użytkownika zleceń. Jeżeli bufor BZFR nie zawiera żadnego zlecenia lub jego semafor jest zamknięty, kontroler przekazuje sterowanie /baton/ do kolejnej stacji. Jeżeli bufor BZFR zawiera zlecenie funkcji Global Send Data funkcja ta zostaje dołączona do ramki przesyłanej baton. W wypadku gdy bufor BZFR zawiera zlecenie inne niż Global Send Data to zostaje ono zrealizowane, a następnie kontroler sprawdza zawartość bufora BZFG.

M

Bufor ten może zawierać tylko zlecenie funkcji Global Send Data lub nie zawierać żadnych zleceń.

Ewentualnie występujące zlecenie tej funkcji jest dołączone do ramki przekazującej sterowanie dalej, jaka jest wysyłana po skontrolowaniu bufora BZFG.

Zlecenia wykonania funkcji Management przekazywane są kontrolerowi poprzez bufor BFZM. Zlecenia te mają najniższy priorytet i mogą być realizowane w czasie gdy kontroler nie ma do wykonania innych czynności.

Wykonywanie tego zlecenia może być przerwane przez każde żądanie wykonania obsługi zgłoszone przez układ sprzęgający.

### 3.4. Komunikacja między kontrolerem a użytkownikiem.

Komunikacja między kontrolerem a użytkownikiem odbywa się za pomocą wspólnego obszaru pamięci i przerwań. Kontroler sygnalizuje użytkownikowi wykonanie zleczonej mu operacji lub inne wydarzenie np.: odebranie przesyłki typu Send Data, zmiana w Live List, odłączenie od linii itp., przez wysłanie przerwania. Informacja o typie wydarzenia zapisana jest w polu specyfikacji zdarzenia /PSZ/ i ma następujące znaczenie:

Zawartość PSZ	Znaczenie
1	Zrealizowanie zleczonej funkcji typu Send/Confirm
2	Odebranie przesyłki typu Send/Confirm.
3	Zrealizowanie zleczonej funkcji typu Global Send Data.
4	Odebranie przesyłki typu Global Send Data.
5	Zrealizowanie zleczonej funkcji typu Request/ /Response.
6	Zrealizowanie zleczonej funkcji typu Management.
7	Zrealizowanie zleczonej funkcji typu Station Recovery.
8	Wykrycie zmiany w Live List.
9	Odłączenie stacji od linii.

Dalsze informacje związane z wyspecyfikowaną w PSZ sytuacją znajdują się w buforze BO /wyjątkowo, gdy PSZ=6 to informacja jest w buforze BOFM/ i postać ich jest zgodna z dokumentami PROWAY-a.

12

Po wykorzystaniu informacji znajdujących się w buforze odpowiedzi użytkownik zeruje pierwszy bajt bufora. Jest to sygnał dla kontrolera, że użytkownik jest gotowy do przyjęcia następującej funkcji.

Dla umożliwienia kontrolerowi obsługi funkcji Request/Response przychodzących z innych stacji użytkownik podaje kontrolerowi za pomocą funkcji Management adres listy adresów buforów BDFR /LBDFR/. Każdy z buforów BDFR zawiera:

bajt 0 - wskaźnik stanu bufora /WBDFR/. WBDFR = 1 oznacza, że bufor zawiera informację do wysłania. WBDFR = 0 sygnalizuje użytkownikowi, że informacja została wysłana do odbiorcy, a kontrolerowi, że brak jest danych do wysłania.

bajt 1 - semafor bufora /SBDFR/

bajt 2 - liczba bajtów danych do wysłania /LBD/

bajt 3 do 2 + LBD - dane do wysłania.

Po odebraniu przez kontroler funkcji typu Request/response przeszukiwana jest lista adresów buforów /LBDFR/. Zerowa wartość adresu bufora oznacza brak bufora z danymi dla stacji żądającej. Jeżeli adres jest niezerowy to kontroler sprawdza we wskazanym miejscu wskaźnik stanu, jeśli WBDFR = 1 sprawdzany jest semafor SBDFR. Jeżeli semafor jest otwarty, to kontroler zamyka go uniemożliwiając w ten sposób użytkownikowi modyfikację danych podczas wysyłania ich do stacji żądającej. Po wysłaniu danych semafor jest otwierany a wskaźnik stanu bufora przybiera wartość zerową. W przypadku gdy WBDFR = 0 lub WBDFR = 1 i semafor jest zamknięty lub w liście adresów brak adresu bufora, do stacji żądającej wysyłana jest odpowiedź negatywna.

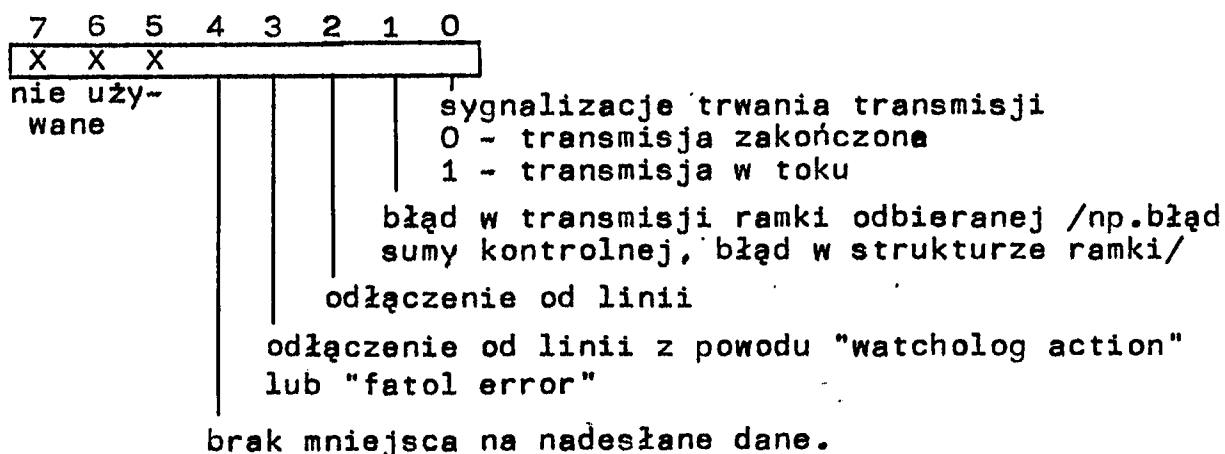
Ze względu na to, że mikroprocesor INTEL 8086 nie zawiera w liście rozkazów rozkazu "ustaw semafor" istnieje konieczność zorganizowania obsługi semafora w inny sposób. Przyjęto, że semafor otwarty ma wartość 0. Zamknięcie semafora odbywa się przez wykonanie rozkazu XCHG semafor, AH, /zamiana zawartości semafora z rejestrem AH/. Rejestr AH ma uprzednio nadaną wartość 1. Następnie wykonywane są rozkazy:

CHP	AH, 0
JNZ	ADR

W przypadku, gdy próba zamknięcia semafora odbyła się już na zamkniętym semaforze zostanie wykonany skok do ADR wyspecyfikowanego w rozkazie. Jeżeli semafor był otwarty skok nie nastąpi.

### 3.5. Komunikacja między układem sprzęgającym a kontrolerem magistrali

Układ sprzęgający komunikuje się z kontrolerem magistrali za pomocą przerw i przekazywania informacji we wspólnych obszarach pamięci. Po rozpoczęciu odbioru ramki układ sprzęgający ładuje nagłówek ramki /tzn.: adres przeznaczenia, kod funkcji, adres źródła/ do bufora NAG. Podczas ładowania nagłówka do bufora układ sprzęgający sprawdza adres przeznaczenia i kod funkcji otrzymywanej ramki. Po odebraniu nagłówka układ sprzęgający wysyła przerwanie do kontrolera. Jeżeli adres przeznaczenia jest zgodny z jego adresem i funkcja wskazuje na przesyłanie informacji zostaje sprawdzone, czy użytkownik podał funkcję Management adres pola na dane odbierane z innych stacji /BDP/. Adres ten ma długość 3 bajtów. Wartość pierwszego bajtu - 1 oznacza, że bufor na dane nie został przygotowany i układ sprzęgający zapala odpowiedni bit w słowie stanu SS. Jeżeli bufor jest przygotowany układ sprzęgający ładuje do niego przychodzące dane. Poprzez słowo stanu układ sprzęgający sygnalizuje również wystąpienie innych sytuacji. Postać słowa stanu /SS/ jest następująca:



### 3.6. Komunikacja między kontrolerem magistrali a układem sprzęgającym

W okresach czasu wolnych od wysyłania i odbierania ramek układ sprzęgający nasłuchuje zlecenia ze strony kontrolera magistrali. Kontroler przekazuje zlecenia poprzez pole ABF, które jest cyklicznie odczytywane przez układ sprzęgający. Zerowa wartość tego pola oznacza brak zlecenia wysłania ramki.

W chwili gdy kontroler chce zainicjować wysłanie ramki w linię, wpisuje w pole ABF adres bufora z informacjami, na podstawie których układ sprzęgający tworzy ramkę przekazywaną w postaci szeregowej do sterownika linii. W polu ABF może znaleźć się adres jednego z trzech buforów: BZFR, BZFG lub BRB. W pierwszych bajtach tych buforów znajduje się informacja jaki rodzaj ramki ma być wysłany. Po zakończeniu transmisji układ sprzęgający oprócz ustawienia odpowiedniego bitu w słowie stanu, zeruje pole ABF.

### 3.7. Wielkości obszarów pamięci dostępnych dla kontrolera komunikacyjnego.

Obszar pamięci wewnętrznej kontrolera magistrali musi pomieścić program procesora 8086 kontrolera magistrali i dane robocze tego programu. Przewiduje się, że pamięć programu /PROM/ powinna wynosić 4k, natomiast pamięć robocza /RAM/ - 1k.

Określenie wielkości pamięci wewnętrznej układu sprzęgającego wykracza poza zakres niniejszej pracy. Wielkość obszaru 1 pamięci operacyjnej jest określona wymaganiami stawianymi przez oprogramowanie procesorów użytkownika odwołujące się do kontrolera komunikacyjnego /system operacyjny i SZPAK-DS/ i wynosi dla MM80 - 32k, a dla MM86 - 64k. Wielkość obszaru 2 wynika z długości przechowywanych w nim buforów i wynosi 11 byte'ów.

### 4. Opis i schemat blokowy programu realizującego krytyczny fragment protokołu PROWAY

Przedstawiając przygotowany na mikroprocesor I8086 program realizujący operacje składające się na określony w rozdziale drugim krytyczny co do wymagań czasowych fragment protokołu PROWAY trzeba przede wszystkim wskazać na ograniczenia jakie przyjęto przy opisanu powyższego programu. Jest on tylko częścią całości oprogramowania kontrolera komunikacyjnego. Oprogramowanie takie nie istnieje, nie ma nawet jego projektu. Tworzenie go w celu wykonania niniejszej pracy było niemożliwe i bezprzedmiotowe. Celem pracy było tylko oszacowanie czasów potrzebnych na realizację protokołu PROWAY na mikroprocesorze I8086 i odpowiedź na pytanie czy ten mikroprocesor może być użyty jako procesor kontrolera komunikacyjnego stacji PROWAY.

Napisany program jest tylko sekwencją rozkazów jakie muszą być wykonane w czasie realizacji określonego fragmentu protokołu. Od tego ciągu rozkazów odchodzą liczne odgałęzienia /jako skoki warunkowe/, które już do krytycznej części nie należą.

Przy tworzeniu całości oprogramowania kontrolera prawdopodobnie pojawiłaby się potrzeba wydłużenia napisanego programu, tak że gotowy kontroler byłby wolniejszy niż wynika to z poniższych rozważań, o czym trzeba pamiętać przy ocenie ich wyników.

Po przedstawieniu przyjętych ograniczeń i zastrzeżeń zostaną teraz przedstawione założenia przyjęte przy tworzeniu programu. Komunikacja między kontrolerem a użytkownikiem stacji PROWAY odbywać się będzie poprzez bufory we wspólnej pamięci oraz przerwania z kontrolera do użytkownika. Przerwań w przeciwnym kierunku nie przewiduje się.

Komunikacja między kontrolerem a układem sprzęgającym odbywać się będzie również poprzez wspólne obszary pamięci i przerwania układu sprzęgającego do kontrolera. I tu także nie przewiduje się przerwań w kierunku przeciwnym.

Z tak rozwiązaną komunikacją wiąże się sposób oprogramowania kontrolera. Przyjęto, że działania protokołu PROWAY związane ze współpracą z linią - odbiór i analiza ramek, aktualizacja Live List, nadawanie ramek /w szczególności przekazywanie baton-u i związane z tym określenie adresów przeznaczenia wysyłanych batonów/ będą przez oprogramowanie kontrolera realizowane na poziomie programu obsługi przerwania. Fragmentem tego programu jest opisywana tu krytyczna sekwencja. Zadane przez użytkownika zlecenia typu Management będą wykonywane w procesorze kontrolera przez program główny, który w każdej chwili może być przerwany i zawieszony przez przerwanie z układu dopasowującego. Nie został określony sposób przekazywania komunikatów i generowania przerwań z kontrolera do użytkownika. Wykonywanie tych działań w podprogramie obsługi przerwania byłoby prostsze, ale komunikaty te musiałyby być wysyłane w miarę działań kontrolera, niezależnie od tego, czy użytkownik jest gotowy na ich przyjęcie. Istniałoby wtedy niebezpieczeństwo zagubienia części komunikatów. Rozwiązanie drugie - wysyłanie komunikatów do użytkownika przez program główny i kolejgowanie ich pozwoliłoby uniknąć powyższego niebezpieczeństwa, ale skomplikowałoby oprogramowanie kontrolera i spowolniłoby jego działanie.



W pracy kontrolera magistrali obok krótkich, krytycznych okresów czasu  $/T_1 = 100 \mu s/$ , w których stacja musi rozpocząć nadawanie po otrzymaniu sterowania lub przesyłki wymagającej odpowiedzi występują stosunkowo długie czasy odbioru ramek z linii.

Przy szybkości transmisji, jaką przyjęto w rozważaniach, wynoszącej 100k bit/sek, czas odbioru jednego bajtu trwa 80  $\mu s$ ek. Każda ramka to minimum 6 bajtów - daje to 480  $\mu s$ ek. Czas ten można wykorzystać na wykonanie wielu działań protokołu PROWAY, co znacznie ułatwia spełnienie jego wymagań czasowych.

W omawianym programie wykorzystano tę możliwość. Układ sprzęgający zgłasza przerwanie do kontrolera po prawidłowym odebraniu nagłówka ramki /bajtu synchronizacyjnego, adresu przeznaczenia, bajtu funkcji i adresu źródłowego/. Kontroler po wystartowaniu programu obsługi przerwania i zapamiętania stanu mikroprocesora na stosie analizuje bajty nagłówka, przygotowuje aktualizację Live List, sprawdza czy użytkownik zgłasza żądane wykonanie funkcji, i jeżeli tak to jakiej. Dzieje się to w czasie odbioru pozostałej części ramki i ew. w czasie przerwy w linii  $T_1$ . Następnie kontroler sprawdza czy transmisja zakończyła się poprawnie i jeżeli tak, inicjuje nadawanie. Inne działania, jakie według protokołu PROWAY powinny być wykonane po zakończeniu odbioru przesyłki - zwolnienie informacji zatrzymanej uprzednio w Hold, przekazania informacji typu Global Send Data, są wykonywane już w trakcie nadawania ramki przez sterownik. Przy tworzeniu programu przyjęto również pewne rozwiązania mające jeszcze przyspieszyć jego działanie.

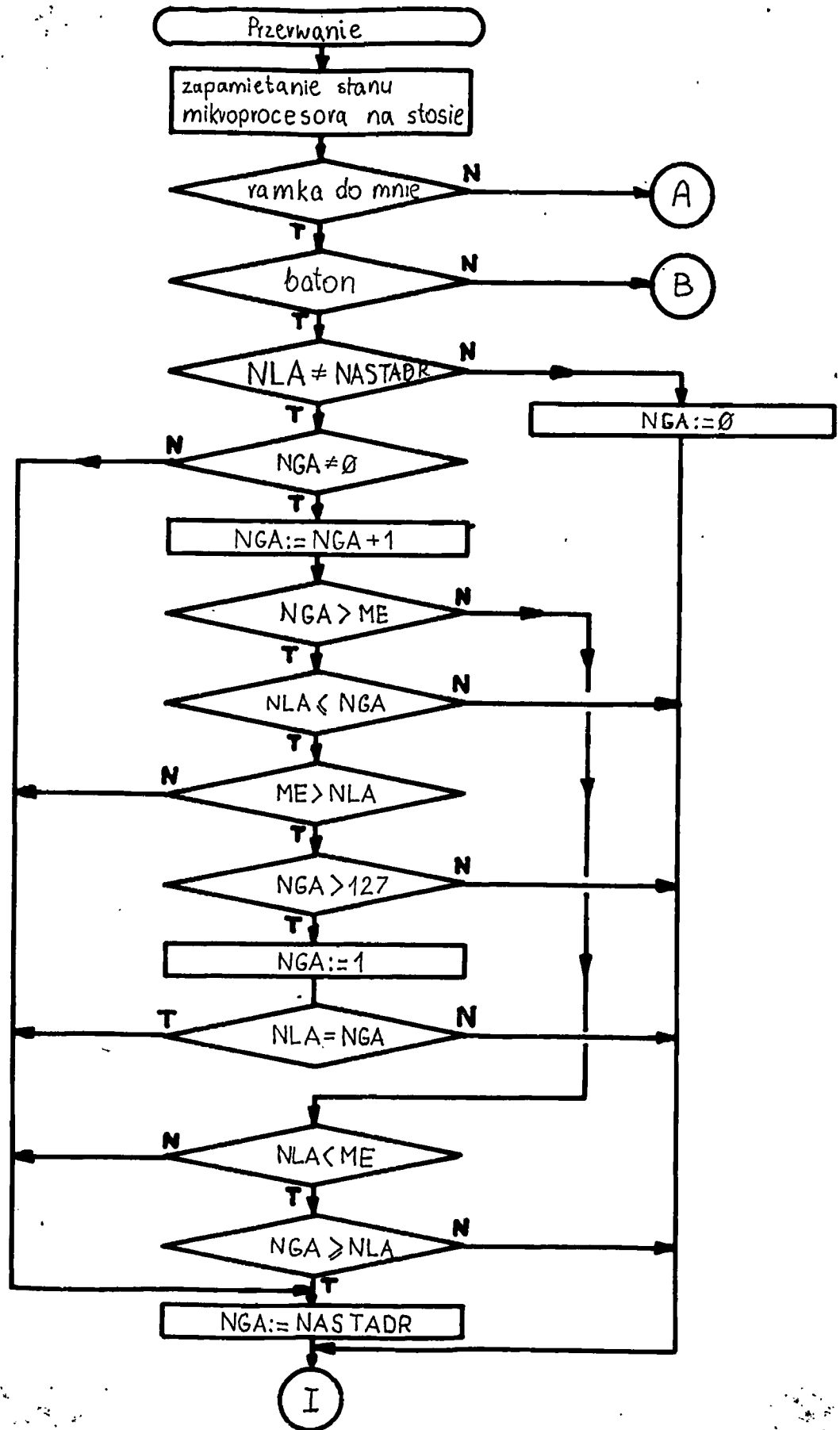
Bufor ze zleceniem funkcji do wykonania z użytkownika do kontrolera nie jest przepisywany w celu przekazania zlecenia sterownikowi, kontroler uruchamia układ dopasowujący podając mu tylko adres bufora uprzednio zapisanego przez użytkownika i zanalizowanego przez siebie. W ten sposób unika się bardzo czasochłonnego przepisywania, wymagającego modyfikacji adresów, odczytu i zapisu do pamięci bajt po bajcie, czy słowo po słowie.

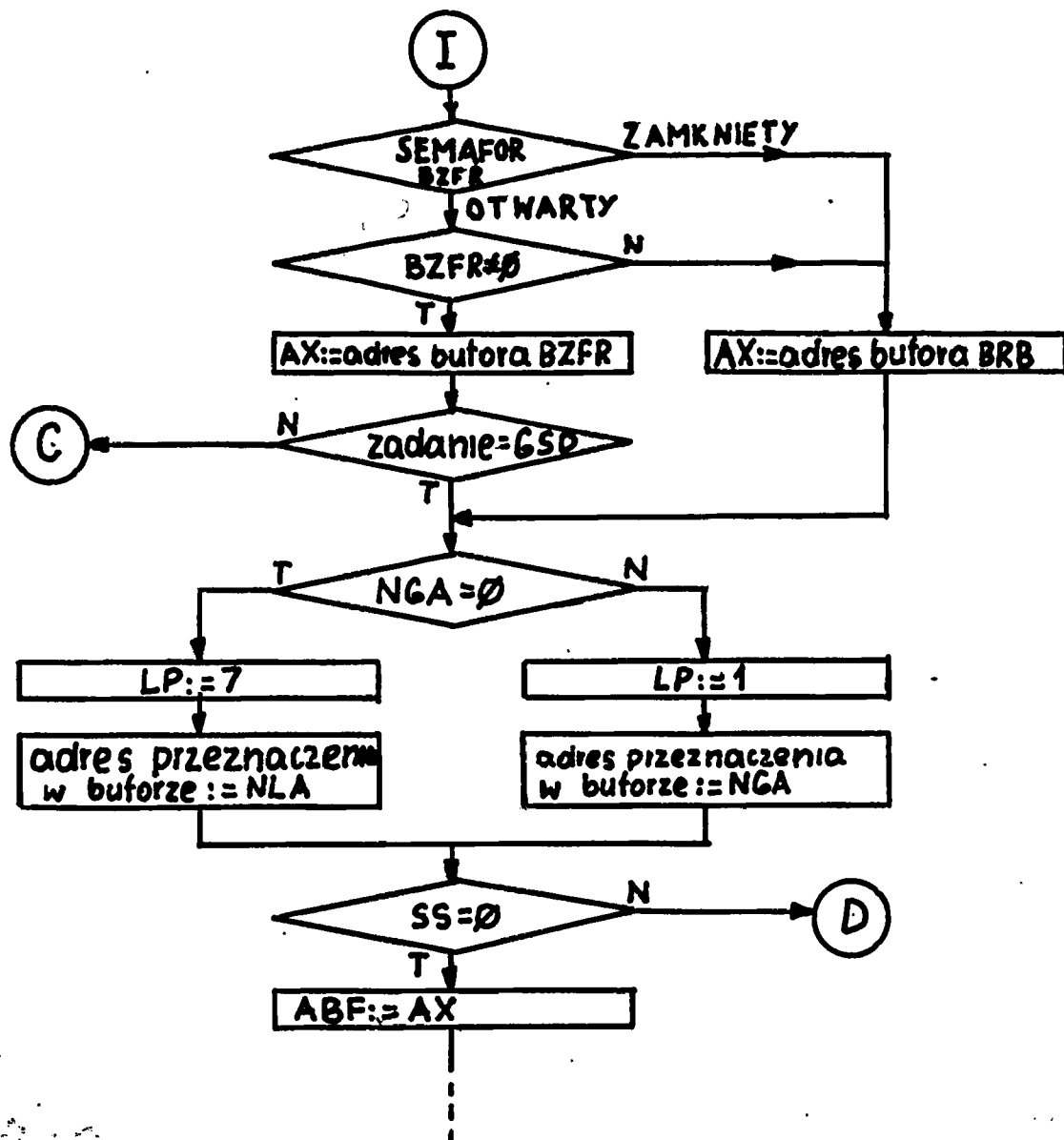
Szczególnie wiele uwagi poświęcono organizacji Live List, tak by jej aktualizacja pochłaniała jak najmniej czasu. Początkowo rozważana Live List miała postać mapy bitowej, na której każdej stacji PROWAY odpowiadałby 1 bit. Wartość tego bitu sygnalizowałaby, czy dana stacja jest aktywna czy nie. Okazało się, jednak, że przeszukiwanie i aktualizacja takiej mapy bitowej, zwłaszcza przy

małej ilości stacji /na możliwych i wymagających sprawdzenia 127 adresów/ byłaby złożona i bardzo czasochłonna. Ostatecznie przyjęto rozwiązanie, w którym Live List jest ciągiem bajtów, w których są zapisane kolejne adresy stacji aktywnych.

Jest także pamiętany wskaźnik określający miejsce ostatnio zapisanego adresu. Kontroler po stwierdzeniu wystąpienia stacji aktywnej dopisuje jej adres jako następny do tworzonej Live List. Aktualizowana w ten sposób Live List zostaje zakończona wtedy gdy kontroler odbierze baton zaadresowany do siebie i "koło" przekazywania sterowania zostanie zamknięte. Kontroler przechowuje w pamięci dwie Live List; formowaną poprzednio, już zakończoną i tworzoną bieżąco. Po zapisaniu nowego adresu kontroler porównuje następstwo adresów stacji aktywnych w obu Live List i w ten sposób może stwierdzić ich zmiany. Zakłada się, że na żądanie podania Live List do użytkownika kontroler będzie podawał wersję już zakończoną.

Działanie programu ścieżki krytycznej przedstawia schemat blokowy. W schemacie zaznaczone zostały odgałęzienia "donikąd", będące potencjalnymi połączeniami napisanego programu z całością oprogramowania procesora kontrolera komunikacyjnego.





Objaśnienia skrótów użytych na schemacie

NLA - Next Live Address

NGA - Next Gap Address

ME - mój adres

NASTADR - adres stacji następnej za ME

NASTADR = ME+1 gdy ME  $\neq$  127

= 1      gdy ME = 127

BZFR - bufor zleceń różnych od funkcji Management /p.rozdz.3/

GSD - funkcja Global, Send Data

LP - liczba powtórzeń prób przesłania batonu

ABF - słowo w pamięci, gdzie jest zapisywany adres bufora  
ze zleceniem dla układu sprzęgającego

AX - rejestr mikroprocesora 8086

A, B, C, D - wyjścia z programu realizującego krytyczny  
fragment protokołu.

5. Listing programu

```

MOJEDANE          SEGMENT
NAG               DB      ???,??,??,?    #BUFOR NA NAGLOWEJ K
                                     #Z UKLADU DOPASOWUJACEGO

LIVELIST1        127    DB      127 DUF (/)  ?)
LIVELIST2        DB      127DUF      7 DUF (?)
NGAI1            DB      ?
SBZFR            DB      / ?
BZFR             DB      7 DUF (?)    #BUFOR NA ZLECENIE UZYTK
BRB              DB      7 DUF (?)    #BUFOR NA ZLECENIE DODAT
LP               DB      ?           #LICZBA POWT TRANSMISJI
ABF              DW      ?           #TU KONTROLER ZAPISUJE ADRES
                                     #ZLECENIA DLA UKL. DOPAS
                                     #OWSZAR NA STOS
STOS              EQU    THIS WORD    #WIERZCHOLEK STOSU
SS               DB      ?           #SLOWO STANU UKL DOPAS
MOJEDANE         ENDS
PROTOCOL         SEGMENT
ASSUME CS: PROTOCOL,
&                DS: MOJEDANE,
&                SS: MOJEDANE
START:          MOV     AX,MOJEDANE    #ZAPISZ DS I SS
                MOV     DS,AX
                MOV     SS,AX
                LEA    SP,STOS        #PRZYGOTUJ STOS
;
;NIEOKRSLONE CZESCI PROGRAMU, ICH PRACA JEST PRZEKRYWANA
;PRZERWANIEM GENEROWANYM PRZEZ UKLAD DOPASOWUJACY PO ODEBRANIU
;BAJTU ADESU ZRODLOWEGO RAMKI Z LINII
;
NASTADR          EQU    122
MYADRES          EQU    121
LINIA            PUSH   AX
                PUSH   DX
                CMP    NAG,MYADRES    #CZY RAMKA DO MNIE
                JNE    DBCY           #SKOK GDY NIE DO MNIE
                MOV    DL,NAG+1       #POBIERZ 2 BAJT
                AND    DL,7FH         #MASKOWANIE
                JNZ    NIEBATON       #SKOK GDY NIE BATON
;GDY BATON DO MNIE - PRZYGOTOWANIE AKTUALIZACJI LIVE LIST
                MOV    DL,NAG+2       #DL: NOWY ADRES ZRODL
                MOV    CBX,LSI,DL     #ZAPAMIETANIE W LL
;LIVE LIST - CIAG KOLEJNYCH ADRESOW AKTYWNYCH
;W SEGMENTCIE MOJEDANE 2 LISTY - POPRZEDNIA I BIEZACA
;ADRESOWANIE TABLIC - BASED,INDEXED
;POCZATKI TABLIC:POPRZEDNIEJ - BP, BIEZACEJ - BX
;INDEKS W TABLICY: SI
;PONIEWAZ UZYCIE REJESTRU BP WSKAZUJE NA TABLICE
;W STACK SEGMENT NALEZY SS=DS
                MOV    DH,CBXJ        #DH: NLA
                CMP    DH,NASTADR     #CZY JEST GAP
                JNE    JESTGAP
                MOV    DL,0           #GDY NIE MA GAP
                JMP    ENDGAP
JESTGAP:         MOV    DL,NGA        #GDY JEST GAP
                CMP    DL,0           #CZY BYL GAP
                JE     NAGGAP         #GDY NIE BYLO GAP
                INC    DL             #DL: NOWY NGA
                CMP    DL,MYADRES     #NGA > ME ?
                JLE    ETI
  
```

*22*

```

CMP      DH,DL          #NLA > NGA ?
JG       ENDGAP
CMP      DH,MYADRES    #NLA > ME ?
JG       NASGAP
CMP      DL,128        #NGA > 127 ?
JL       ENDGAP
MOV      DL,1          #NGA: 1
CMP      DL,DH         #NLA = NGA ?
JNE      ENDGAP
JMP      NASGAP
E11:    CMP      DH,MYADRES    #NLA < ME ?
JG       NASGAP
CMP      DL,DH         #NGA < NLA ?
JL       ENDGAP
NASGAP: MOV      DL,NASTADR    #NGA = NASTEPNIK
#KONIEC WSTEPNEGO PRZYGOTOWANIA LIVE LIST
#      DL: NOWY NGA
#      DH: NOWY NLA
#KONTROLA ZLECEN UZYTKOWNIKA
#CZY SEMAFOR BUFORA BZFR OTWARTY
ENDGAP: MOV      AH,1
LOCK     XCHG     SBZFR,AH
CMP      AH,0
JNE      ZAMKNIETY    #SKOK GDY SEMAFOR ZAMKNIETY
#CZY UZYTKOWNIK ZADA FUNKCJI W BUFORZE BZFR ?
CMP      BZFR,0
JNE      JESTZAU      #SKOK GDY JEST ZADANIE
ADRBI    EQU      WORD PTR BZFR+1
ADRBU    EQU      WORD PTR BRB+1
#ADRBI(1) - ADRES SLOWA ZADRESEM PRZEZNACZENIA RAMKI
#W BUFORZE BZFR(BRB)
ZAMKNIETY: LEA     AX,ADRBI    #GDY NIE MA ZADANIA
JMP      ZA
JESTZAU:  LEA     AX,ADRBI    #GDY JEST ZADANIE
CMP      BZFR,2        #CZY ZADANIE = GSD ?
JNE      ZADANIE      #SKOK DO ZADANIA
#GDY NIE MA ZADANIA LUB ZADANIE = GSD - OKRESLENIE ADRESU PRZEZNACZENI
#BATONU I LICZBY POWTORZEN PRZESLANIA
ZA:      CMP      DL,0      #CZY JEST SZCZELINA ?
JNE      DONLA        #SKOK GDY NIE MA
MOV      LP,1         #LICZBA POWTORZEN = 1
MOV      CAXI,DL      #PRZESLIJ NGA DO BUFORA
JMP      STIR
DONLA:   MOV      LP,7     #LICZBA POWTORZEN = 7
MOV      CAXI,DH      #PRZESLIJ NLA DO BUFORA
#KONTROLA POPRAWNEGO ZAKONCZENIA RAMKI ODBIERANEJ
STIR:    CMP      SS,0     #SLOWO STANU = 0 ?
JNE      TRWA
MOV      AHF,AX       #PRZESLIJ ADRES BUFORA
#ZE ZLECENIEM DO UKLADU
#DOPASOWUJACEGO - START
#NAUWANIA
#PROGRAM NIE ZAWIERA KONTROLI REALIZACJI CZASU PRZERWY W LINII
#
# TU NASTĘPIJĄ DAJSZE CZĘŚCI PROGRAMU
#
# .....
PROTOCOL  ENDS
# .....
END      START

```

## 6. Określenie czasu wykonania programu

Czas wykonania programu zamieszczonego w rozdziale 5 został określony na dwa sposoby: obliczono go teoretycznie i zmierzono po uruchomieniu.

Przy obliczeniach teoretycznych wykorzystano podane przez producenta mikroprocesora INTEL 8086 długości wykonania instrukcji w cyklach zegarowych. Założono maksymalną dostępną częstotliwość zegara mikroprocesora wynoszącą 8 MHz. Nie uwzględniono czasów dostępu do pamięci. Czasy te będą różne w zależności od przyjętej struktury sprzętowej stacji. Największy wpływ na czas dostępu będzie miało to, czy wszystkie urządzenia stacji będą korzystały ze wspólnej pamięci przez wspólną szynę danych, czy będą miały autonomiczne pamięci oraz bufory dostępne dla innych urządzeń. Czas wykonania programu może być różny w zależności od:

- stopnia złożoności określenia Next Live Address i Next Gap Address - występują znaczne różnice w przebiegu przeszukiwania Live List w zależności od wartości adresów,
- stanu bufora BZFR zawierającego zlecenia użytkownika.

Rozpatrzono wszystkie możliwe przypadki i tą drogą określono najniekorzystniejszy z nich i najdłuższy czas wykonania programu. Najgorszy przypadek przeszukiwania Live List następuje gdy Next Live Address = 1, adres mojej stacji jest oczywiście większy od 1, ale mniejszy od 127 / tzn. jest "szczelina" - "gap" pomiędzy moją stacją a następną stacją aktywną/, ostatnio sprawdzany z wynikiem negatywnym - Next Gap Address miał wartość 127. Najgorszy z rozpatrywanego punktu widzenia stan bufora BZFR jest postaci: semafor otwarty, zgłoszone zadanie = Global Send Data. W wypadku kombinacji tych dwóch zdarzeń czas wykonania programu wyniesie 88 usek.

Czas wykonania programu został również zmierzony empirycznie. Program został uruchomiony na skonstruowanym w Zespole Budowy Cyfrowych Urządzeń Systemowych doświadczalnym układzie mikroprocesorowym z mikroprocesorem D8086D produkcji japońskiej firmy NEC. W programie realizującym krytyczny fragment protokołu PROWAY zostały dokonane zmiany umożliwiające jego uruchomienie i zmierzenie czasu wykonania:



- dodano fragment mający być miejscem dotychczasowych "skoków do nikąd" tzn. skoków do nieokreślonych części programu /jest nim tu zawieszenie wykonywania programu/,
  - w początkowej części programu umieszczono instrukcje nadające początkowe wartości rejestrom mającym zawierać dane wykorzystywane w programie,
  - zmiennym definiowanym w programie nadano wartości odpowiadające opisanemu wyżej najgorszemu przypadkowi,
  - zapętłono program, tak by część której czas wykonania należy zmierzyć wykonywała się cyklicznie, każdorazowo tak samo,
  - dodano instrukcję wyjścia umożliwiającą pomiar na oscyloskopie.
- Tak przygotowany program należało przekodować na postać binarną. Ponieważ nie posiadamy assemblera mikroprocesora INTEL 8086 - ASM-86, zostało to zrobione ręcznie. Następnie program został wprowadzony do pamięci zestawu i uruchomiony przy użyciu programu monitora. Zmierzony czas realizacji wyniósł 105  $\mu$ sek.

## 7. Podsumowanie

W poprzednim rozdziale określono czas wykonania programu realizującego krytyczny fragment protokołu PROWAY. Limit czasu będącego do dyspozycji na wykonanie powyższego programu równy jest sumie czasów maksymalnej przerwy w linii i transmisji części ramki po zgłoszeniu przez układ sprzęgający przerwania do kontrolera magistrali.

W przypadku najkrótszej ramki tzn. ramki zawierającej tylko baton, za nagłówkiem są transmitowane jeszcze trzy bajty: dwa bajty sumy kontrolnej i bajt synchronizacyjny. Czas transmisji wynosi więc  $3 \times 80 \mu\text{sek} = 240 \mu\text{sek}$ .

Tak więc w sumie na wykonanie programu kontroler magistrali dysponuje  $240 \mu\text{sek} + 100 \mu\text{sek} = 340 \mu\text{sek}$ .

Czas wykonania programu wynosi /doświadczalnie zmierzony/ 105  $\mu$ sek. Należy go zwiększyć o czas sprzętowej obsługi przerwania przez mikroprocesor 8086 wynoszący 30  $\mu$ sek. Daje to łącznie 135  $\mu$ sek. W stosunku do czasu do dyspozycji wynosi to 40%. Wynik ten pozwala stwierdzić, że jest możliwe zrealizowanie protokołów MIR-PROWAY na mikroprocesorze 8086. Niedokładność wynikająca z przyjętej metody nie powinna mieć na ten wynik zasadniczego wpływu. Rezultat analizy jest zatem pozytywny.

Możliwa jest także realizacja protokołów PROWAY na mikroprocesorze 8086 przy większej szybkości transmisji. Dla 200k bit/sek czas do dyspozycji wyniesie 120 usek + 100 usek = 220 usek. Czas ten zostanie wykorzystany w 61%.

Dla 500k bit/sek czas do dyspozycji wyniesie 48 usek + 100 usek = 148 usek i jego wykorzystanie - 91%.

W oparciu o mikroprocesor 8086 jako procesor kontrolera magistrali możliwe jest zrealizowanie stacji PROWAY pracującej z linią o szybkości transmisji 200k bit/sek. Szybkość 500k bit/sek jest szybkością krytyczną, przyjęcie jej wymagałoby dalszej analizy. Realizacja protokołu PROWAY na mikroprocesorze 8086 dla wyższych prędkości transmisji nie jest możliwa, chyba, że wprowadzono by pewne zmiany do protokołu lub postaci ramek - np. ramki "puste" nie zawierające żadnych informacji przesyłane po to, by uzyskać więcej czasu, lub nadawanie na początku i końcu ramek więcej niż jednego bajtu synchronizacyjnego.

## 8. Literatura

1. "Analiza protokołów MIR-PROWAY pod kątem możliwości ich realizacji na dostępnej w kraju bazie elementowej" - sprawozdanie PIAP 1983r., Nr rej. 5057.
2. "Analiza możliwości realizacji protokołów MIR-PROWAY na dwuprocessorowym układzie 8080-8085 oraz na procesorze 8086. Etap 1. Analiza możliwości realizacji protokołów MIR-PROWAY na dwuprocessorowym układzie 8080-8085" - sprawozdanie PIAP 1983r., Nr rej. 5082.
3. Dokument IEC  
65C/Secretariat/13-Draft -Process data highway /PROWAY/ for distributed process control systems. Part 3 /combined with Parts 4 and 5/: Specification for Highway Protocol Definition.

246