

074

PRZEMYSŁOWY INSTYTUT AUTOMATYKI I POMIARÓW  
MERA-PIAP

Al. Jerozolimskie 202 02-222 Warszawa Telefon 23-70-81

Ośrodek Automatyki Elektrycznej

Pracownia Oprogramowania Wieloprocessorowych Systemów Automatyki

i Robotów

Główny wykonawca - mgr inż. Andrzej Aderek

Wykonawcy - mgr inż. Krzysztof Czarnoński

- mgr inż. Wojciech Hernik

Konsultant - dr inż. Andrzej Syrczyński

- mgr inż. Kazimierz Majdan

Nr zlecenia - 9425

"Analiza protokołów MIR-PROWAY

pod kątem możliwości ich realizacji

na dostępnej w kraju bazie elementowej".

Zleceniodawca - praca własna

Pracę rozpoczęto dnia maj 1983 r.

Kierownik Pracowni

zakończono dnia 30.06.83

Kierownik Ośrodka

mgr inż. A. Aderek p.o. Z-ca Dyrektora  
d/s Automatyki

prof. dr inż. T. Missala

dr inż. T. Gałazka

Praca zawiera:

Rozdzielnik - ilość egz:

stron 34

Egz. 1 BOINTE

rysunków 5

Egz. 2 OAE-8

fotografii

Egz. 3 OBN

tabel 2

Egz. 4 OAE-83

tablic

Egz. 5

załączników

Egz. 6

Nr rejestr. 5057

## Analiza deskryptorowa

OPROGRAMOWANIE + MIKROPROCESOR + MIR-PROWAY +

## Analiza dokumentacyjna

Opracowanie zawiera założenia na kontroler komunikacyjny stacji PROWAY i jego oprogramowanie oraz oszacowanie możliwości realizacji protokołów PROWAY przez kontroler z procesorem 8080.

- c.d.

## Tytuły poprzednich sprawozdań

12' - 23 Oprogramowanie

6 81.32:621.32 - 1977.02 15.15 01.15

UKD

MERA-PIAP/TW 331/78 5000-

2

Spis treści

	str.
1. Wstęp . . . . .	3
2. Założenia na kontroler komunikacyjny . . . . .	3
3. Określenie krytycznego fragmentu protokołu PROWAY . . . . .	12
4. Opis i schemat blokowy programu realizującego krytyczny fragment protokołu PROWAY . . . . .	14
5. Listing programu . . . . .	26
6. Określenie czasu wykonania krytycznego frag- mentu protokołu PROWAY . . . . .	33
7. Podsumowanie . . . . .	34

## 1. Wstęp

Niniejsze opracowanie zawiera sprawozdanie z pracy wykonanej w Pracowni Oprogramowania Wieloprocessorowych Systemów Automatyki i Robotów Ośrodka Automatyki Elektrycznej MERA-PIAP. Stanowi ono wstępny etap prac nad skonstruowaniem i wdrożeniem systemu wymiany informacji w rozłożonych systemach sterowania.

Celem etapu było przeanalizowanie możliwości wykonania oprogramowania na mikroprocesorze INTEL 8080 realizującego protokół systemu PROWAY. Analiza miała dotyczyć czasów wykonywania programów i miała dać odpowiedź na pytanie czy spełnione będą wymagania czasowe stawiane przez projekt normy PROWAY-a. 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/ for distributed process control systems".

## 2. Założenia na kontroler komunikacyjny

Ogólne założenia dotyczące zasady działania kontrolera komunikacyjnego zostały zaproponowane w dokumentach IEC. 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 8080 koniecznym stało się uszczegółowienie opisu

działania kontrolera i jego współpracy z otoczeniem przy uwzględnieniu wymagań stawianych przez sprzęt.

## 2.1. Komunikacja kontrolera z otoczeniem

Kontroler komunikacyjny jest jednym z elementów wchodzących w skład stacji systemu PROWAY. Pozostałymi elementami są użytkownik, sterownik i złącze, rys.2.1. 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 zrealizowania pewnych czynności i sygnalizację ich wykonania lub zaistnienia sytuacji awaryjnych. Użytkownik bezpośrednio współpracuje z kontrolerem komunikacyjnym, który realizuje protokół PROWAY-a. Kontroler komunikuje się z niższą warstwą stacji tzn. ze sterownikiem. Zadaniem sterownika jest organizacja i nadzór przesyłania porcji informacji w linii. Najniższą warstwą stacji jest złącze, którego główną funkcją jest dopasowanie elektryczne linii i stacji oraz zamiana informacji otrzymanej ze sterownika na postać w jakiej ma ona być przesyłana w linii.

Kontroler komunikacyjny jest urządzeniem bezpośrednio współpracującym ze sterownikiem linii oraz z użytkownikiem /np.: kasetą MIR-PROWAY/ i przeznaczony jest do koordynacji procesu wymiany informacji między stacją, w której jest zainstalowany, a innymi stacjami systemu PROWAY. Współpraca odbywa się poprzez wymianę informacji we wspólnych obszarach pamięci a także przez wysyłanie przerw sprzętowych. Na rys.2.2. przedstawiono schemat logiczny wymiany informacji między kontrolerem a otoczeniem.

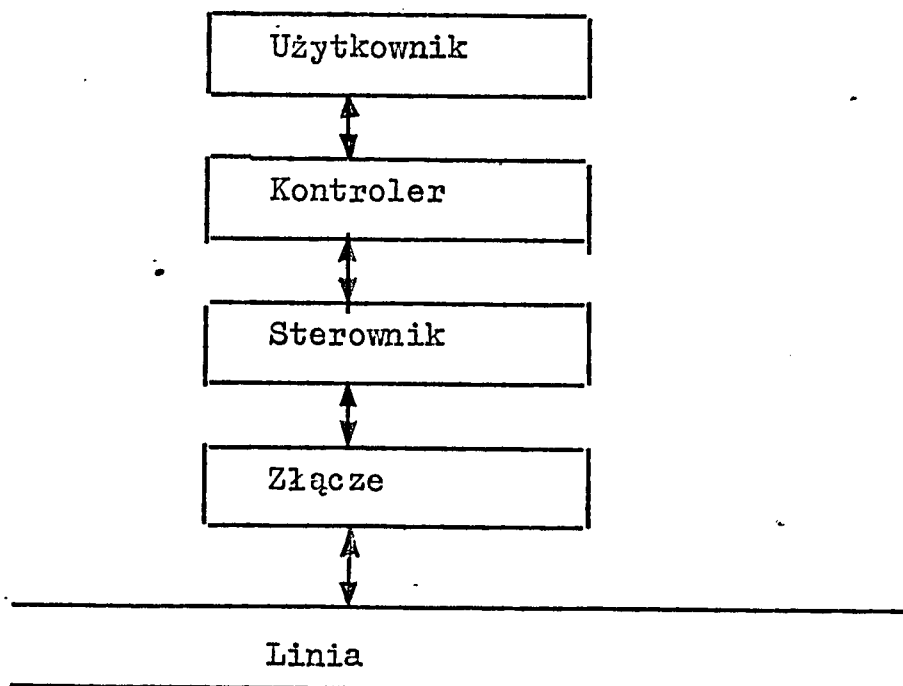
Poszczególne obszary zawierają bufory danych, semafony i wskaźniki. "obszar 1" jest dostępny dla kontrolera i użytkownika. Znajdują się w nim:

- bufor zlecenia funkcji Management /BZFM/ - 7 byte,
- semafor bufora BZFM /SBZFM/ - 1 byte,
- semafor bufora BZFR /SBZFR/ - 1 byte,
- semafor bufora funkcji Global Send Data /SBZFG/ - 1 byte,

- pole specyfikacji zdarzenia /PSZ/ - 1 byte,
  - bufor odpowiedzi na zadaną funkcję /BO/ - 7 byte,
  - bufor odpowiedzi dla funkcji Management /BOFM/ - 19 byte
  - lista adresów buforów BDFR-/LBDFR/ - 3 x 127 byte
- obszar 2 jest dostępny dla kontrolera i sterownika.

Zawiera on:

- 2 bufony na nagłówki ramek /NAG1 i NAG2/ - 2 x 3 byte,
- słowo stanu /SS/ - 1 byte,
- pole z adresem bufora, do którego aktualnie jest ładowana ramka /ABAR/ - 2 byte,
- pole z adresem bufora, do którego poprzednio była ładowana ramka /ABPR/ - 2 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,
- bufor na zlecenie wysłania innych ramek /BRI/ - 7 byte



rys.2.1. Schemat funkcjonalny stacji systemu PROWAY.

Obszar 3 dostępny jest dla użytkownika i sterownika.

Zawiera on:

- 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.

Adresy pól znajdujących się w obszarze 3 przekazywane są do kontrolera poprzez obszar 1 i 4.

Obszar 4 dostępny jest dla kontrolera, użytkownika i sterownika. Znajdują się w nim:

- 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 z danymi pozwalającymi na zrealizowanie funkcji Request/Response przychodzących z innych stacji BDFR- do 126 x 1027 byte.

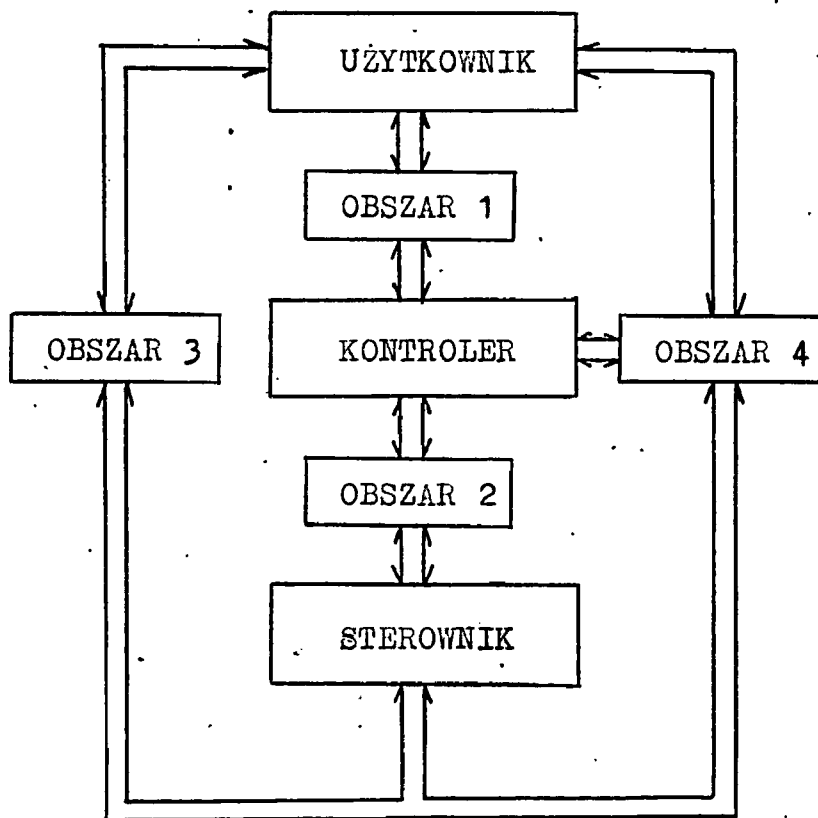
Oprócz wymiany informacji z otoczeniem przez wspólne obszary pamięci kontroler odbiera i przesyła informacje za pomocą przerwań sprzętowych. Ze względu na długi czas /ok. 20  $\mu$ s/ jaki zużywa INTEL 8080 na sprzętową obsługę przerwania ograniczono do minimum ten sposób wymiany informacji.

Przyjęto również, że kontroler odbiera przerwania tylko ze sterownika a sam może wysyłać sygnał przerwania tylko do użytkownika. Okoliczności, w których następuje wymiana przerwań opisane są w dalszych rozdziałach tego opracowania.

## 2.2. Komunikacja między użytkownikiem a kontrolerem

Kontroler komunikacyjny jest urządzeniem podrzędnym 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 obszarach 1 i 4. 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.



rys.2.2. Schemat wymiany informacji między kontrolerem a otoczeniem przez wspólne obszary pamięci.



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 przesyłającej baton. W wypadku gdy bufor BZFR zawiera zlecenie inne niż Global Send Data to zostaje ono zrealizowane, a następnie kontroler sprawdza zawartość bufora BZFG.

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

Ewentualnie występujące zlecenie tej funkcji jest dołączane 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 sterownik.

### 2.3. Komunikacja między kontrolerem a użytkownikiem

Komunikacja między kontrolerem a użytkownikiem odbywa się za pomocą wspólnych obszarów 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.

Zawartość PSZ	Znaczenie
4	Odebranie przesyłki typu Global Send Data.
5	Zrealizowanie zleconej funkcji typu Request/Response.
6	Zrealizowanie zleconej funkcji typu Management.
7	Zrealizowanie zleconej funkcji typu Station Recovery.
8	Wykrycie zmiany w Live List.
9	Odlączenie 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. Po wykorzystaniu informacji znajdujących się w buforze odpowiedzi, użytkownik zeruje pierwszy bajt bufora. Jest to sygnał dla kontrolera, że użytkownik jest gotowy do przyjęcia następnej informacji.

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 kontrolera 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 SBDFF. 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 WBDFF=0 lub WBDFF=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 8080 nie zawiera w liście rozkazów rozkazu "ustaw semafor" istnieje konieczność zorganizowania obsługi semafora w inny sposób. Przyjęto, że semafor otwarty ma wartość -1. Zamknięcie semafora odbywa się przez wykonanie rozkazu INR M /zwiększenie o 1 zawartości komórki pamięci o adresie znajdującym się w parze rejestrów H,L/, co powoduje zapalenie flagu Z. Następnie wykonywany jest rozkaz JZ ADR /skok gdy zapalony jest flag Z/. W przypadku gdy próba zamknięcia semafora odbyła się na już zamkniętym semaforze, flag Z nie zapali się a tym samym nie zostanie wykonany skok do adresu wyspecyfikowanego w rozkazie.

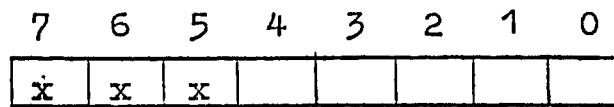
#### 2.4. Komunikacja między sterownikiem a kontrolerem.

Sterownik komunikuje się z kontrolerem za pomocą przerwania i przekazywania informacji we wspólnych obszarach pamięci. Po rozpoczęciu odbioru ramki sterownik wysyła przerwanie do kontrolera i ładuje nagłówek ramki /tzn.: adres przeznaczenia, kod funkcji, adres źródła/ do bufora naprzemian NAG 1 lub NAG 2. O tym, do którego bufora aktualnie ma być ładowany nagłówek, decyduje adres zawarty w polu ABAR.

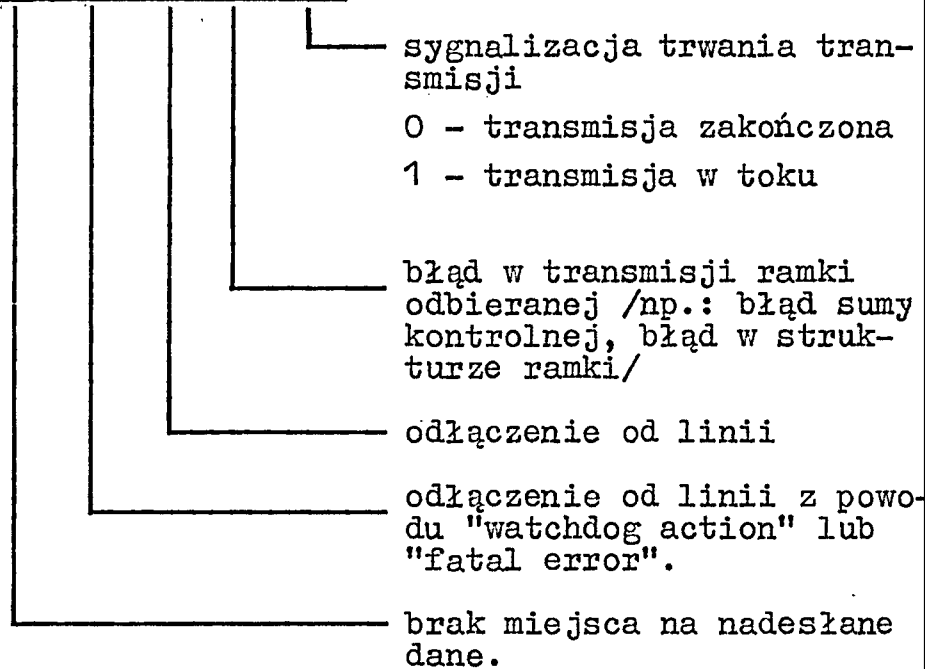
Podczas ładowania nagłówka do bufora sterownik sprawdza adres przeznaczenia i kod funkcji otrzymywanej ramki. Jeżeli adres przeznaczenia jest zgodny z jego adresem i funkcja wskazuje na przesyłanie informacji to sterownik sprawdza, czy użytkownik podał funkcją Management adres pola na dane odbierane z innych stacji /BDP/. Adres ten ma długość 3 bajtów. Wartość pierwszego bajtu - 1 oznacza, że bufor na dane nie został przygotowany i sterownik zapala odpowiedni bit w słowie stanu SS.

M

Poprzez słowo stanu sterownik sygnalizuje również wystąpienie innych sytuacji. Postać słowa stanu /SS/ jest następująca:



nie używane



## 2.5. Komunikacja między kontrolerem a sterownikiem

W okresach czasu wolnych od odbierania i wysyłania ramek sterownik nasłuchuje zleceń ze strony kontrolera. Kontroler przekazuje zlecenia poprzez pole ABF. Zerowa wartość tego pola oznacza brak zlecenia wysłania ramki. W chwili gdy kontroler chce zainicjować wysłanie ramki do linii, wpisuje w pole ABF adres bufora z informacjami, na podstawie którego sterownik buduje ramkę do wysłania. W polu ABF może znaleźć się adres jednego z czterech buforów: BZFR, BZFG, BRD lub BRI. W pierwszych bajtach tych buforów znajduje się informacja jaki rodzaj ramki ma być wysłany. Po zakończeniu transmisji sterownik, oprócz ustawienia odpowiedniego bitu w słowie stanu, zeruje pole ABF.

### 3. Określenie krytycznego fragmentu protokołu PROWAY

Po przyjęciu szkicowych założeń na kontroler komunikacyjny konieczne było ustalenie, jaki fragment protokołu PROWAY stawia największe wymagania co do szybkości działania procesorowi kontrolera, tak by zbadanie możliwości realizacji tego fragmentu na mikroprocesorze I8080 pozwoliło dać szacunkową odpowiedź na pytanie będące tematem niniejszej pracy.

Projekty norm dostarczone przez IEC na urządzenia i protokół transmisji PROWAY stawiają wysokie i różnorodne wymagania czasowe przed zespołami stacji nadzorującymi współpracę z linią transmisyjną i bezpośrednio realizującymi nadawanie i odbiór przesyłek.

Są to następujące wymagania:

- 1/ realizacja transmisji z zadaną szybkością 100 kb na sekundę,
- 2/ analiza na bieżąco odbieranych z linii nagłówków ramek i ew.przesyłanie do pamięci stacji otrzymywanych informacji,
- 3/ wysłanie potwierdzenia na funkcję typu Send/Confirm lub Recovery,
- 4/ wysłanie odpowiedzi na funkcję Request/Response,
- 5/ powrót do stanu jałowego, tak by stacja była gotowa na odbiór nowej przesyłki po odebraniu informacji typu Global Send Data z "batonem" nie do mnie lub po wysłaniu odpowiedzi /potwierdzenia/ na funkcję typu Send/Confirm, Recovery, Request/Response,
- 6/ rozpoczęcie nadawania po otrzymaniu sterowania -
  - w wypadku gdy użytkownik zgłasza żądanie wykonania funkcji typu pytanie odpowiedź /z potwierdzeniem lub odpowiedzią/,
  - w wypadku gdy użytkownik takiego żądania nie zgłasza.

Wymaganie przedstawione w pkt.1 dotyczy sprzętu /sterownika linii i łącza/ i nie jest przedmiotem opracowania.

Działania opisane w pkt.2 mają być według założeń zawartych w poprzednim rozdziale, wykonywane przez sterownik dysponujący własnym procesorem. Sterownik ładuje kolejne bajty nagłówka ramki /adres przeznaczenia, funkcja, adres źródła/ do bufora dostępnego dla kontrolera, sprawdza adres przeznaczenia i funkcję i jeżeli ramka zawiera przeznaczone dla danej stacji dane, na które jest miejsce w pamięci stacji, to sterownik przesyła bajty informacji do pamięci stacji.

Na wykonanie 3 operacji porównania i zapisu bajtu sterownik ma 160, a od otrzymania ostatniego bajtu - 80  $\mu$  sek. Jest to czas wystarczający na wykonanie wymienionych czynności przez procesor sterownika.

Operacje wymienione w pkt.3 i 4 sprowadzają się do kontroli poprawności otrzymanej przesyłki /słowa stanu sterownika/ i załadowania bufora ze zleceniem transmisji dla sterownika.

Analizując punkt 5 należy pamiętać, że według przyjętych założeń na komunikację kontroler-sterownik, procesor kontrolera może zakończyć obsługę przesyłki /odpowiedzi, potwierdzenia/ przed ukończeniem jej nadawania przez sterownik. Tak więc na powrót do stanu jałowego /nie związanego z obsługą przesyłek/ kontroler dysponuje czasem transmisji wysyłanej ramki, czasem przerwy między ramkami, czasem odbioru bajtu synchronizacyjnego następnej ramki. W sumie daje to co najmniej  $480+100+80=660 \mu$  sek. W tym czasie kontroler musi zdążyć osiągnąć gotowość do odbioru następnej ramki.

Sprowadza się to przede wszystkim do powrotu z przerwania /p.rozdział następny/.

Znacznie więcej czynności w krótszym okresie czasu musi zostać wykonane w przypadkach wymienionych w pkt.6.

W tym czasie kontroler musi analizować nagłówek odebranej ramki, po zakończeniu odbioru - sprawdzić czy był on prawidłowy, czy otrzymana ramka jest ważna, następnie jeżeli w buforze "hold" jest przechowywana informacja porównać ją z aktualnie odebraną i jeżeli są różne, przechowywane dane przekazać do użytkownika, kolejno kontroler sprawdza funkcję odebranej ramki, jeżeli jest nią pałeczka /"baton"/, następuje aktualizacja Live List i dalej - jeżeli jest to pałeczka do mnie - określenie nowych Next Live Address i Next Gap

Address i kontrola zgłoszeń użytkownika.

Funkcja zgłoszona przez użytkownika jest realizowana, jeżeli zgłoszenie nie występuje - następuje przesłanie pałeczki do Next Gap lub /jeżeli go brak/ do Next Live Address. Schemat powyższych czynności jest przedstawiony w dokumencie IEC nr 65C /Secretariat/13 jako zestaw schematów blokowych stanów kontrolera według protokołu PROWAY.

Na podstawie szacunkowej analizy wymienionych wyżej punktów określono fragment protokołu PROWAY stanowiący najwyższe wymagania czasowe procesorowi kontrolera komunikacyjnego. Jest nim działanie kontrolera po otrzymaniu przezeń pałeczki /"baton"/, opisane wyżej jako punkt 6, przypadek drugi.

Przedstawiony wybór ma charakter szacunkowy i nakazuje pewną ostrożność przy ocenie wyników dalszych badań. Odpowiedź negatywna na pytanie będące tematem pracy dana na podstawie powyższego wyboru będzie odpowiedzią pewną, odpowiedź pozytywna będzie tylko odpowiedzią przybliżoną. Dla uzyskania pewnej odpowiedzi pozytywnej należałoby przeanalizować cały protokół PROWAY, wykonać i oprogramować kontroler komunikacyjny, a następnie przetestować go w systemie.

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

Przedstawiając przygotowany na mikroprocesor I8080 program realizujący operacje składające się na określony w poprzednim rozdziale krytyczny co do wymagań czasowych fragment protokołu PROWAY trzeba przede wszystkim wskazać na ograniczenia jakie przyjęto przy pisaniu powyższego programu. Jest to 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 I8080 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.

Nie został określony sposób pomiaru czasu - m.in. okresów czasu w jakich powinny być odbierane kolejne bajty w trakcie odbioru ramki z linii. Należy wykluczyć pomiar czasu przez użycie zegara, generującego przerwania i zliczanie przerwania zegarowych, ze względu na długi w stosunku do mierzonych czasów czas obsługi przerwania.

Pomiar czasu należałoby realizować poprzez wykonywanie, zapewne w pętli, określonych ilości rozkazów o znanym czasie wykonania. Tak więc uwzględnienie pomiaru czasu wydłużyłoby napisany program o segmenty organizujące powyższe pętle.

Przy tworzeniu całości oprogramowania kontrolera prawdopodobnie pojawiłaby się potrzeba wydłużenia napisanego programu o inne jeszcze segmenty, 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 sterownikiem odbywać się będzie również poprzez wspólne obszary pamięci i przerwania ze sterownika do kontrolera. I tu także nie przewiduje się przerwania 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 sek-



wencja. 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 ze sterownika. Nie został określony sposób przekazywania komunikatów i generowania przerw z kontrolera do użytkownika. Wykonywanie tych działań w podprogramie obsługi przerwania ze sterownika 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 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 100 k bit/sek, czas odbioru jednego bajtu trwa  $80 \mu sek$ . Każda ramka to minimum 6 bajtów - daje to  $480 \mu sek$ . Czas ten można wykorzystać na wykonanie wielu działań protokołu PROWAY oszczędzając w ten sposób czas  $T_1$  tylko na te operacje, których nie można wykonać kiedy indziej.

W ten sposób został napisany omawiany program. Odbierane z linii bajty są analizowane przez kontroler w miarę ich przekazywania ze sterownika. Sterownik zgłasza przerwanie do kontrolera po prawidłowym odebraniu bajtu synchronizacyjnego. Kontroler po wystartowaniu programu obsługi przerwania i zapamiętaniu stanu mikroprocesora na stosie analizuje adres przeznaczenia i funkcję odbieranej przesyłki oraz przygotowuje aktualizację Live List w czasie trwania odbioru.

Po zakończeniu odbioru ramki zostaje tylko sprawdzone, czy transmisja była poprawna i, jeżeli tak, wykonania jakiej funkcji żąda użytkownik. Następnie kontroler inicjuje nadawanie ze sterownika. Inne działania, jakie według protokołu

PROWAY powinny być wykonane po zakończeniu odbioru przesyłki - zwolnienie informacji zatrzymanej uprzednio w Hold, przekazanie 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. Wskaźniki używane w trakcie analizy odbieranej z linii ramki są przechowywane w rejestrach, na których działania są wykonywane znacznie szybciej niż na argumentach przechowywanych w pamięci RAM. Dzięki temu unika się długotrwałego i kłopotliwego na mikroprocesorze I8080 adresowania. W wypadku braku wolnych rejestrów wskaźniki są zapamiętywane na stosie, co także daje oszczędności czasowe w stosunku do korzystania z miejsc w pamięci adresowanych bezpośrednio.

Bufor ze zleceniem funkcji do wykonania z użytkownika do kontrolera nie jest przepisywany w celu przekazania zlecenia sterownikowi, kontroler uruchamia sterownik 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.

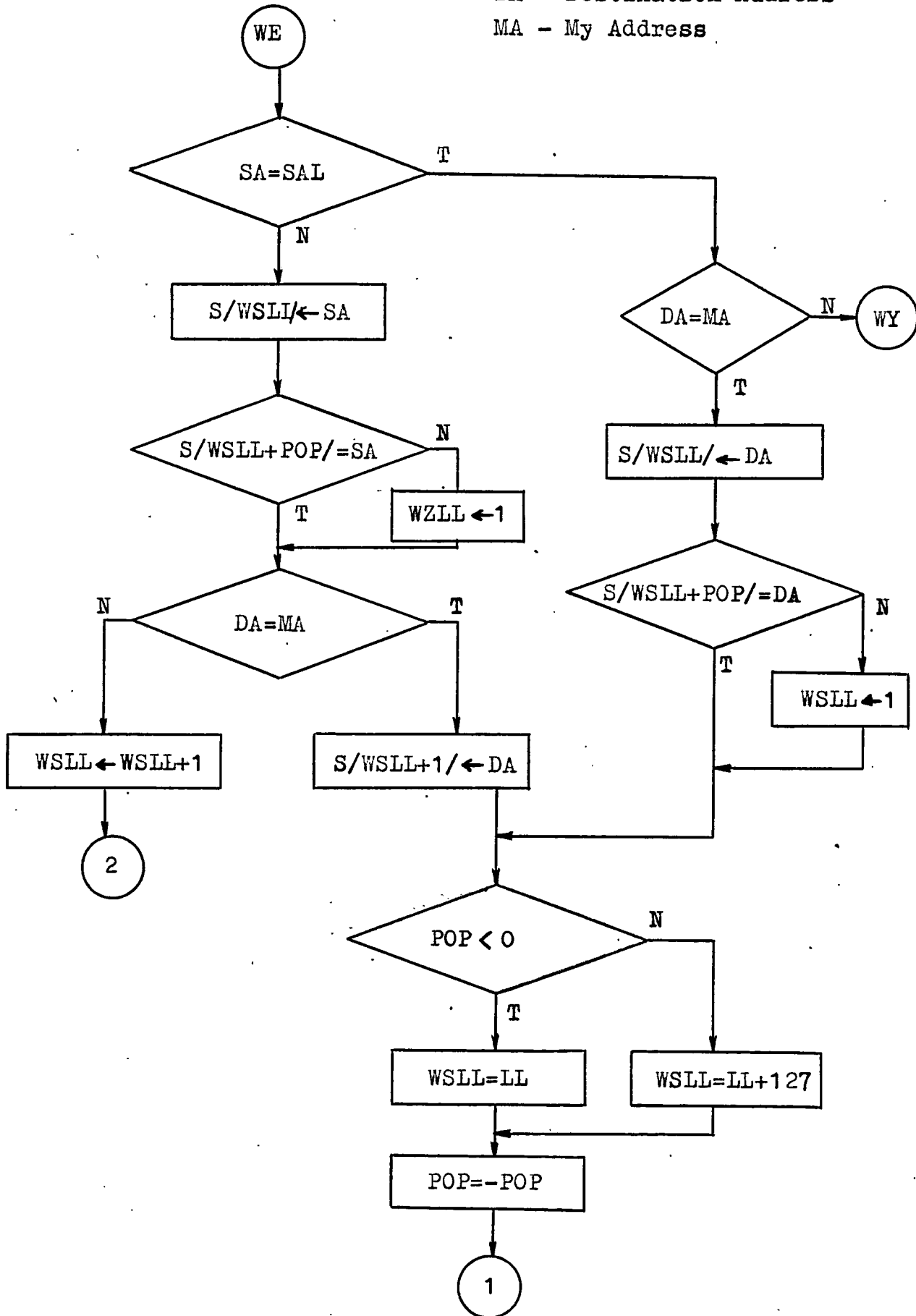
Szczególnie wiele uwagi poświęcono organizacji Live List, tak by jej aktualizacja pochłaniała jak najmniej czasu. Początkowo rozważana Live List miała postać mapy bitowej, na której każdej stacji PROWAY odpowiadałby 1 bit. Wartość tego bitu sygnalizowałaby, czy dana stacja jest aktywna czy nie. Okazało 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 słów-bajtów, w których są zapisane kolejne adresy stacji aktywnych.

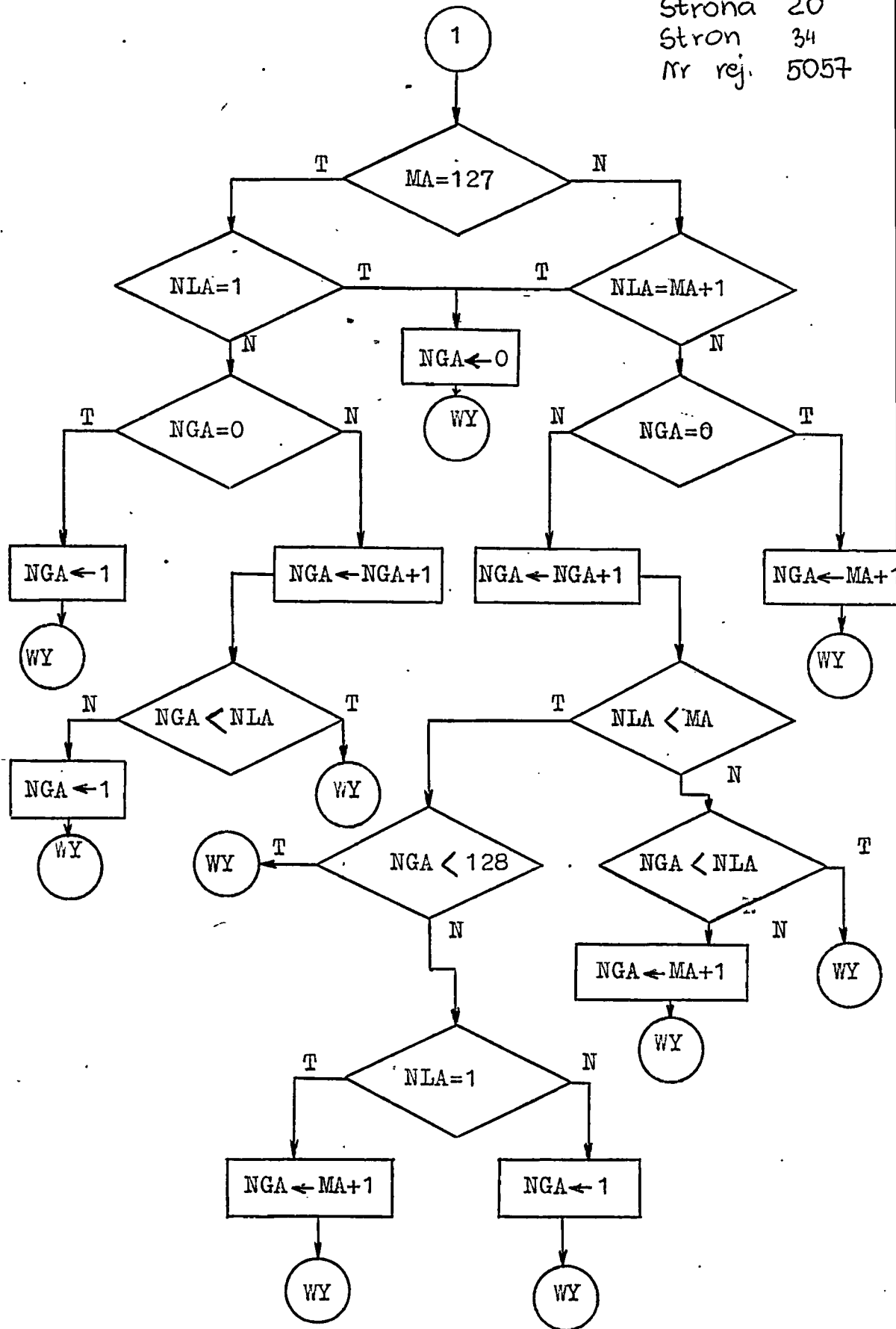
Jest także pamiętany wskaźnik określający miejsce ostatnio zapisanego adresu. Kontroler po stwierdzeniu wystąpienia stacji aktywnej dopisuje jej adres jako następny do tworzonej Live List. Aktualizowana w ten sposób Live List zostaje zakończona wtedy gdy kontroler odbierze baton zaadresowany do siebie 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 ukończoną.

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

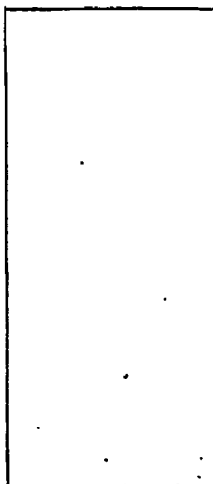
SA - Source Address  
DA - Destination Address  
MA - My Address





Dane do aktualizacji Live List, wyszukiwania Next Live Address i Next Gap Address

LL



- Live List zawierająca 2 x 127 bajtów

WSLL - 2 bajty na adres pierwszego wolnego miejsca w aktualnie zapełnianej Live List,

WZLL - 1 bajt na wskaźnik zmian w Live List od chwili odczytania jej ostatnia funkcja Management /0 - brak zmian, 1 - były zmiany/.

POP - wskaźnik aktualnie zapełnianej Live List  
/POP =  $\pm 127$ / - 1 byte

POP = 127 - aktualnie zapełniana Live List jest od adresu LL

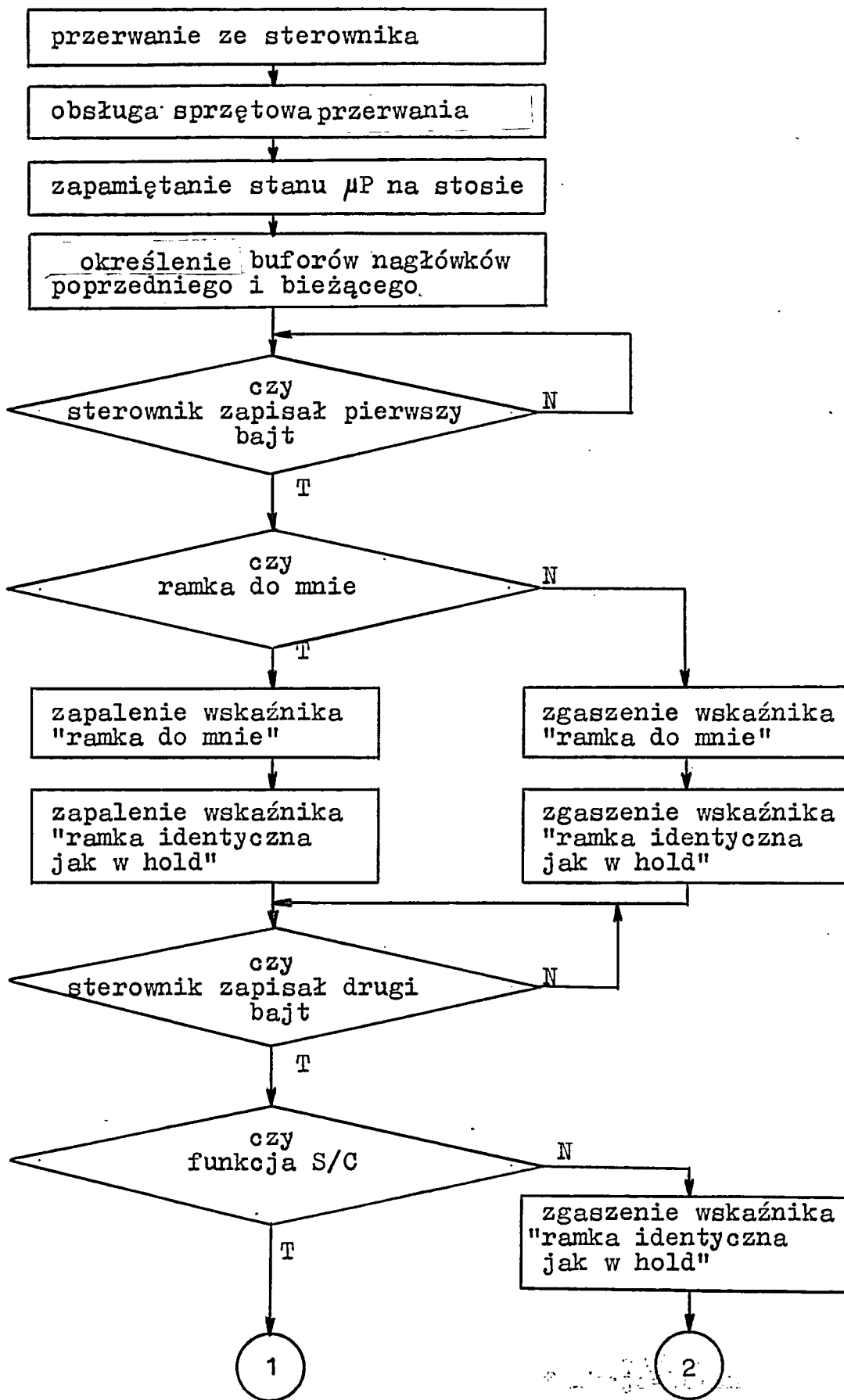
POP = -127- aktualnie zapełniana Live List jest od adresu LL + 127

SAL - adres stacji która jako ostatnia wysłała "baton"  
/1 byte/

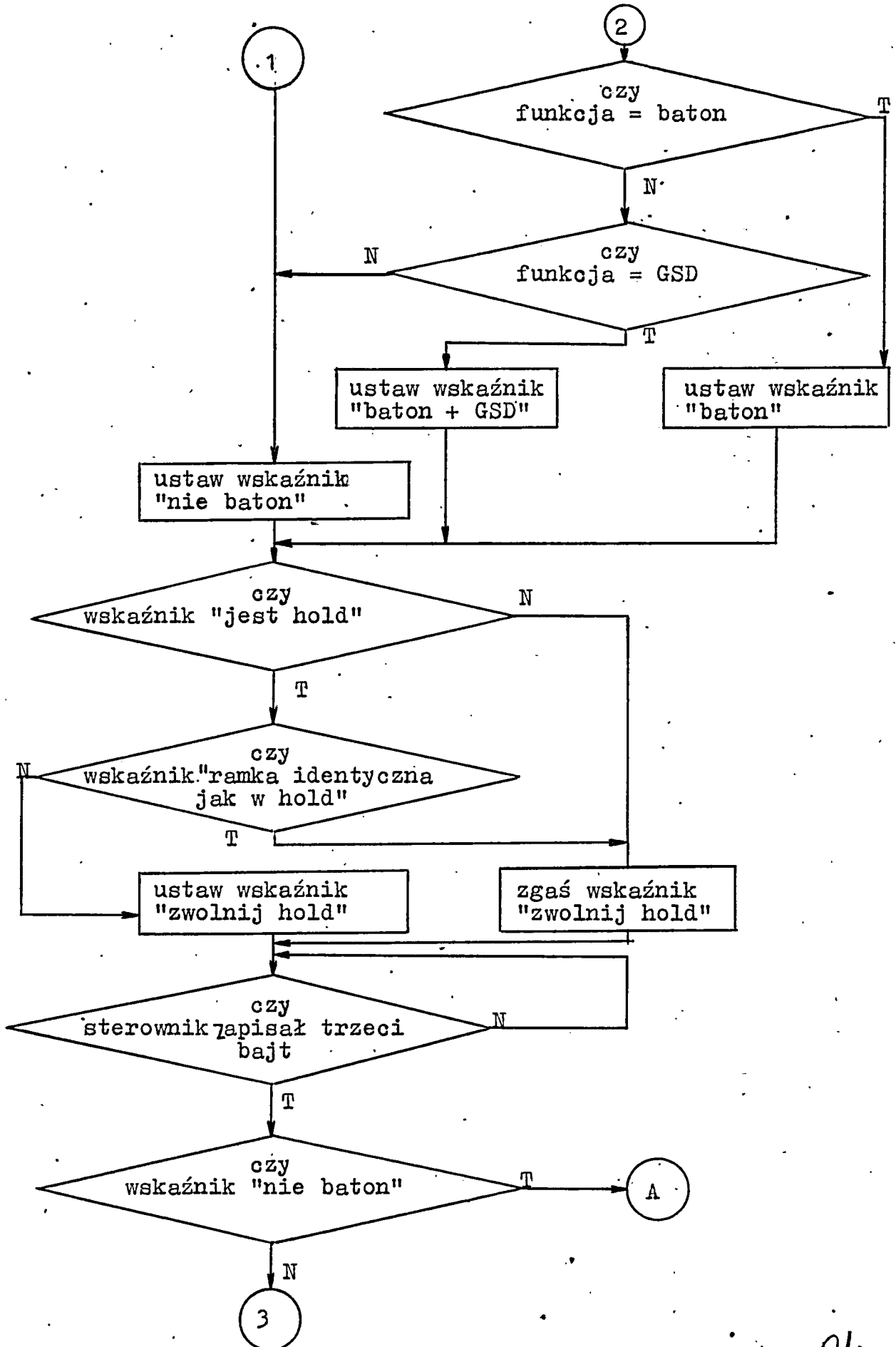
NGA - 1 byte zawierający adres stacji która ostatnio próbowaliśmy ożywić /Next Gap Address/

$\left. \begin{array}{l} DE \\ HL \end{array} \right\}$  - 2 x 2 bajty na przechowanie zawartości rejestrów.  
D, E, H, L.

SCHEMAT BLOKOWY PROGRAMU ŚCIEŻKI KRYTYCZNEJ 104

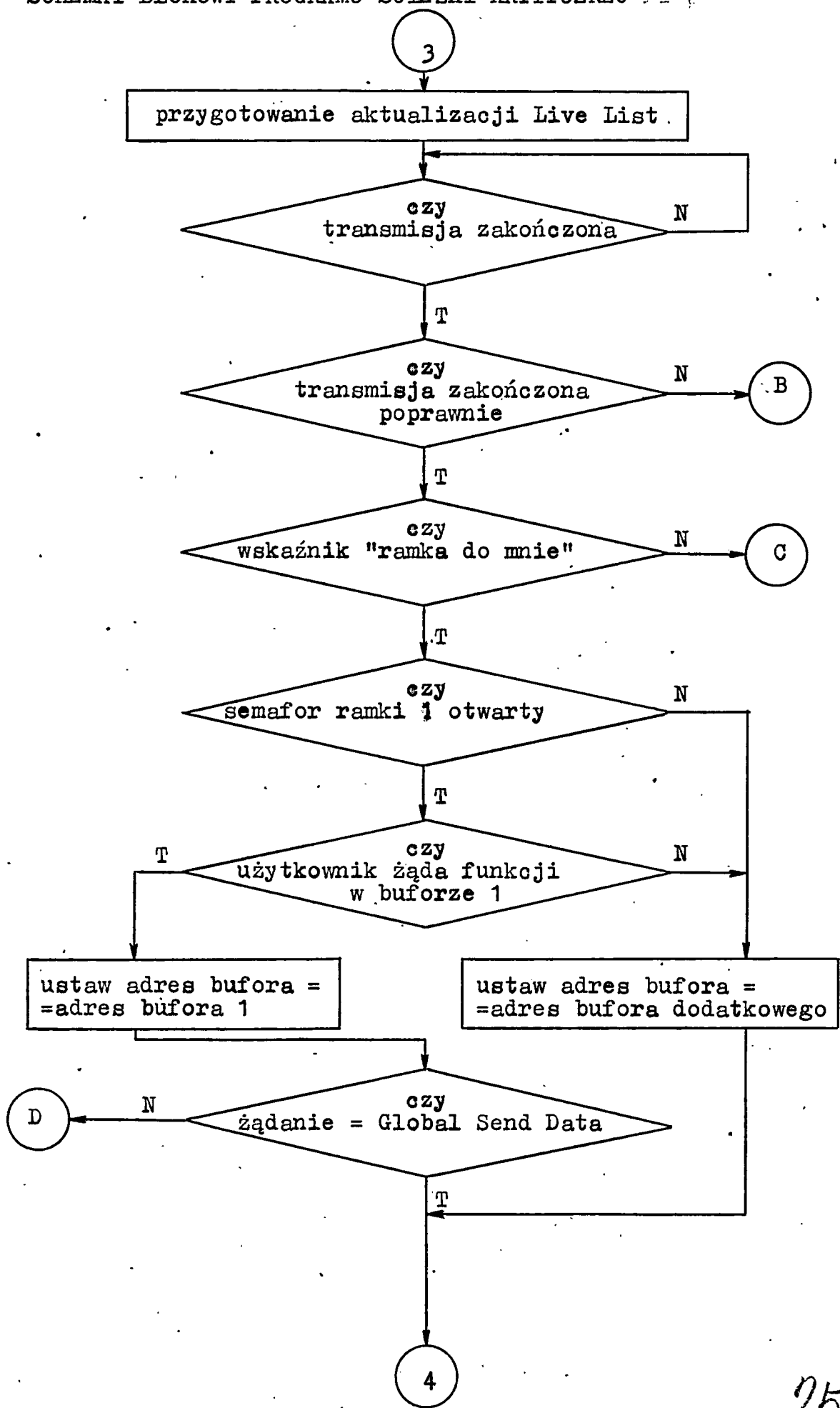


SCHEMAT BLOKOWY PROGRAMU ŚCIEŻKI KRYTYCZNEJ

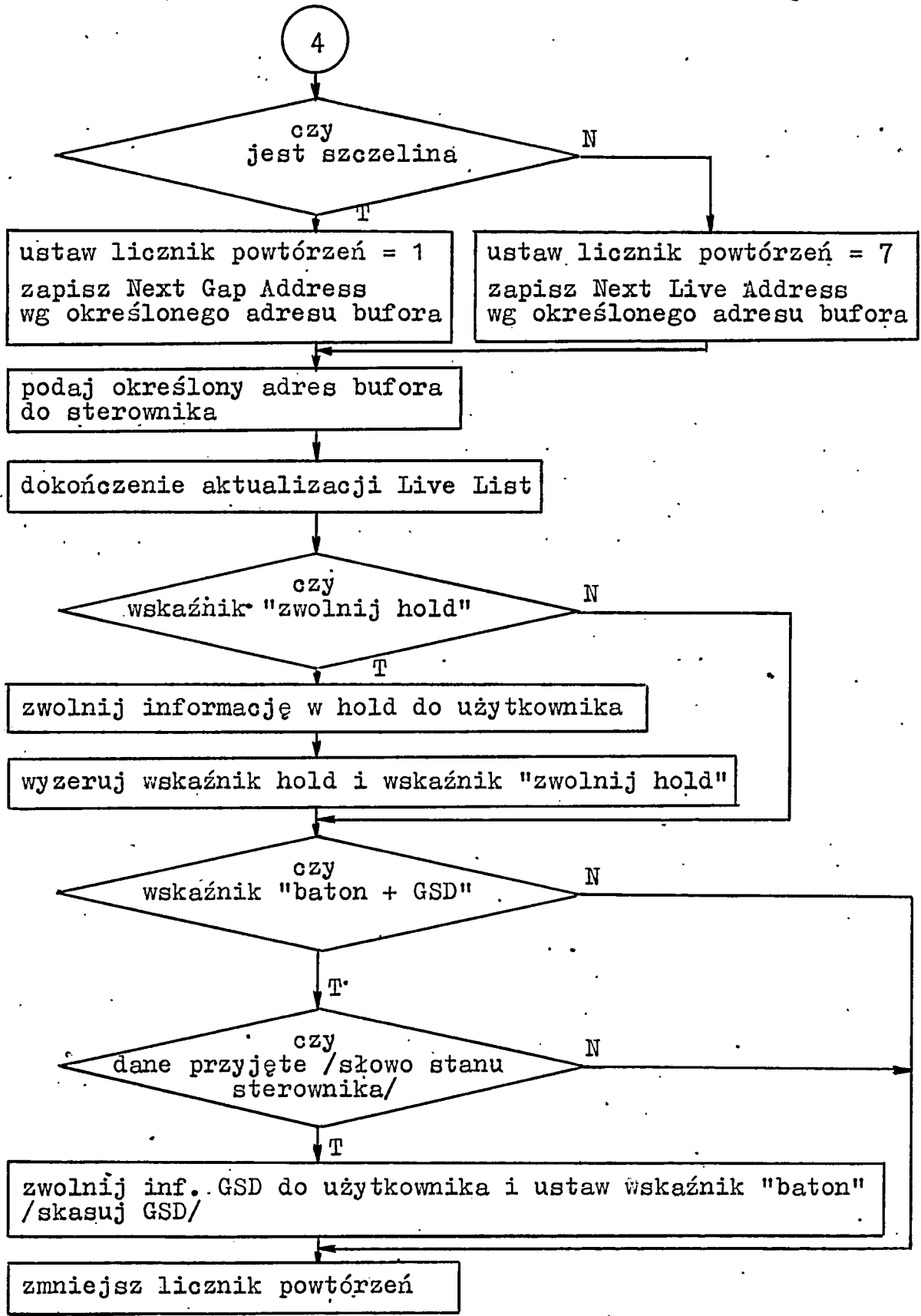




SCHEMAT BLOKOWY PROGRAMU ŚCIEŻKI KRYTYCZNEJ



SCHEMAT BŁOKOWY PROGRAMU ŚCIEŻKI KRYTYCZNEJ



## 5. Listing programu

```
;ZAPAMIETANIE STANU MIKROPROCESORA PO ZGLOSZENIU PRZERWANIA
;ZE STEROWNIKA
HWY      PUSH    PSW
        PUSH    B
        PUSH    D
        PUSH    H
;ZAPAMIETANIE ADRESOW NAGLOWKOW POPRZEDNIEGO I BIEZACEGO
;S(AHD) - ADRES AKTUALNIE LADOWANEGO BUFORA
;S(LHD) - ADRES POPRZEDNIO LADOWANEGO BUFORA
        LHL D    ABPR
        SHL D    LHD
        LHL D    ABAR
        SHL D    AHD
;CZY STEROWNIK ZAPISAL PIERWSZY BAJT
;ILOSC BAJTOW ZAPISANYCH = S(LBAR)
        MVI     B,0
E1      LDA     LBAR
        CMP     B
        JZ      E1
;CZY RAMKA DO MNIE.S(MYAD)-MOJ ADRES (NR STACJI)
        MOV     B,M
        LDA     MYAD
        CMP     B
        JNZ     E2
;JEZELI RAMKA JEST DO MNIE
;REJ D - WSK. RAMKA DO MNIE
;REJ E - WSK. RAMKA IDENTYCZNA JAK W HOLD
;WARTOSCI WSKAZNIKOW: 0 - NIE,1 - TAK
        MVI     D,1
        MOV     E,D
        JMP     E3
;JEZELI RAMKA JEST NIE DO MNIE
E2      MVI     D,0
        MOV     E,D
;CZY STEROWNIK ZAPISAL DRUGI BAJT
E3      MVI     B,1
        LDA     LBAR
        CMP     B
        JZ      E3
;CZY FUNKCJA SEND/CONFIRM
        INX     H
        MVI     B,#81
        MOV     A,M
        CMP     B
        JNZ     E4
;JEZELI FUNKCJA SEND/CONFIRM
;REJ C - WSK. BATON - NIE BATON - BATON+GLOBAL SEND
; NIE BATON C>0
; BATON C=0
; BATON+GSD C<0
```

```

E7      MVI      C,1
        JMP      E5
;JEZELI NIE SEND/CONFIRM
;ZGASZONY WSKAZNIK W REJ E
E4      MVI      E,0
;CZY FUNKCJA BATON
        CMP      E
        JZ      E6
;JEZELI NIE BATON - CZY GLOBAL SEND
        DCR      B
        CMP      B
        JNZ     E7
        MOV      C,B
        JMP      E5
;JEZELI FUNKCJA BATON
E6      MOV      C,E
;JEZELI RAMKA ROZNA NIZ W HOLD - CZY W HOLD JEST RAMKA
;REJ B - WSK. ZWOLNIJ HOLD
; B=1 NIE ZWALNIAJ
; B=0 ZWALNIAJ
;HOLD - WSK "JEST HOLD"
; HOLD=0 JEST
; HOLD=1 NIE MA
E5      LDA      HOLD
        ORA      E
        MOV      B,A
;PRZYGOTOWANIE DO ODBIORU TRZECIEGO BAJTU
        INX      H
;CZY STEROWNIK ZAPISAL TRZECI BAJT
E8      LDA      LBAR
        CPI      2
        JZ      E9
;CZY FUNKCJA ZAWIERA BATON
        MVI      A,0
        CMP      C
        JC      E9          ;WYJSCIE "DO NIKAD"
;JEZELI FUNKCJA ZAWIERA BATON - AKTUALIZACJA LIVE LIST
;W H,L ADRESA, W ADRES-2 JEST DA
        PUSH     B          ;PRZECHOWANIE B,C,D,E
        PUSH     D
        MVI      D,0          ;D=0
        LDA      POP
        MOV      E,A          ;E=POP
        MOV      B,M          ;B=SA
        DCX      H
        DCX      H
        MOV      C,M          ;C=DA
        LHLD     WSL
        LDA      SAL
        CMP      B
        JZ      SAME          ;SKOK GDY SA=SAL
        MOV      M,B
        DAD      D
        CMP      M
        JZ      EET1
        INR      D
EET1    LHLD     WSL
        INX      H          ;WSL=WSL+1
        MOV      A,C
        CPI      MA
  
```

```

EET11  JNZ      WYJ      ;SKOK GDY DA=/MA
        MOV     M,A    ;S(WLL+1)=DA
        MOV     A,E
        CPI     0
        JM      EET2   ;SKOK GDY POP<0
        LXI     H,LL+127
        LDA     LL
        JMP     EET3
EET2   LXI     H,LL
        LDA     LL+127
EET3   MOV     B,A    ;B=NLA
        MOV     A,E
        CMA
        INR     A
        MOV     E,A    ;E=-POP
        MVI     A,MA   ;WYSZUKANIE NGA
        CPI     127
        JZ      EET4   ;SKOK GDY MA=127
        INR     A
        CMP     B
        JZ      EET5   ;SKOK GDY NLA=MA+1
        LDA     NGA
        CPI     0
        JZ      EET6   ;SKOK GDY NGA=0
        INR     A
        MOV     C,A    ;C=NGA+1
        MOV     A,B
        CPI     MA
        JP      EET8   ;SKOK GDY NLA<MA
        MOV     A,C
        CPI     128
        JM      WYJ    ;SKOK GDY NGA<128
        MVI     A,1
        CMP     B
        JNZ     EET7   ;SKOK GDY NLA=/1
EET7   ADI     MA
        MOV     C,A    ;C=NGA
        JMP     WYJ
EET5   MVI     C,0
        JMP     WYJ
EET6   LDA     MA+1
        JMP     EET7
EET8   MOV     A,C
        CMP     B
        JM      WYJ
        LDA     MA+1   ;NGA=MA+1
        JMP     EET7
EET4   MOV     A,B
        CPI     1
        JZ      EET5
        LDA     NGA
        CPI     0
        JZ      EET9   ;SKOK GDY NGA=0
        INR     A
        MOV     C,A    ;C=NGA+1
        CMP     B
        JM      WYJ
EET9   MVI     C,1    ;C=1
        JM      WYJ

```

```
SAME  MOV    A,C
      CPI    MA
      JNZ    WYJ          ;SKOK GDY DA=/MA
      MVI    B,0
      MOV    C,E
      DAD    B
      CMP    M
      JZ     EET11       ;S(WSL)=DA
      MVI    D,1         ;WZLL=1
WYJ    SHLD  HL          ;PRZECHOWANIE D,E,H,L
      XCHG                ;DO AKTUALIZACJI LL
      SHLD  DE
;CZY ODBIOR RAMKI ZAKONCZONY
;SLOWO STANU STEROWNIKA - SS
      LXI    H,SS
E10    MOV    A,M
      RRC
      JC     E10
;CZY ODBIOR RAMKI ZAKONCZONY POPRAWNIE
      STA    MHSW
      ANI    #F7
      CPI    0
      JNZ    E11         ;WYJSCIE "DO NIKAD"
      POP    D
;CZY BATON DO MNIE
      CMP    D
      JZ     E12         ;WYJSCIE "DO NIKAD"
;SPRAWDZENIE SEMAFORA BUFORA ZLECEN BZFR
      LXI    H,BZFR
      INR    M
      JNZ    E13
;CZY UZYTKOWNIK ZGLASZA ZADANIE
      INX    H
      CMP    M
      JZ     E13
;USTAW ADES BUFORA
      INR    A
      INR    A
      CMP    M
      JNZ    E14         ;WYJSCIE "DO NIKAD"
      JMP    E15
;UZYTKOWNIK NIE ZGLASZA ZADANIA LUB SEMAF. ZAMKNIETY
E13    LXI    H,BRB
;CZY JEST SZCZELINA
E15    XRA    A
      CMP    C
      JZ     E16
;GDY JEST SZCZELINA
      MOV    M,C
;USTAW LICZNIK POWTORZEN (REJ D)
      MVI    D,1
      JMP    E17
;GDY NIE MA SZCZELINY
E16    MOV    M,B
;USTAW LICZNIK POWTORZEN
      MVI    D,7
;PODAJ ADRES BUFORA DO STEROWNIKA
E17    SHLD  ABF
;ABF ADRES SLOWA NA ADRES BUFORA PRZEKAZYWANEGO STEROWNIKOWI
```

```

;AKTUALIZACJA LIVE LIST PO WYSLANIU RAMKI
LHLD   AHD           ;ADR. AKTUAL. LADOW. RAMKI
MOV    D,M
INX    H
INX    H
MOV    E,M
LDA    SAL
MOV    B,A
MOV    A,E
STA    SAL
CMP    B
JNZ    ET20         ;SKOK GDY SA=/SAL
MOV    A,D
CPI    MA           ;MA - MOJ ADRES
JNZ    CDN
JMP    ET21
ET20   MOV    A,D
CPI    MA
JNZ    ET22
ET21   MOV    A,C
STA    NGA         ;NGA - NEXT GAP ADRES
ET22   LHLD   HL
SHLD   WSLL        ;WSLL - WSKAZNIK LL
LXI    H,DE
MOV    A,M
STA    POP         ;POP - WSKAZ. AKTUAL. LL
INX    H
LDA    WZLL        ;WZLL - WSKAZNIK ZMIAN W LL
ORA    M
STA    WZLL
;ODTWORZENIE ZE STOSU REJ B,C
CDN    POP    B
MVI    A,0
CMP    B
JNZ    E18
;W TYM MIEJSCU POWINIEN BYC FRAGMENT PROGRAMU POWODUJACY
;ZWOLNIENIE Z "HOLD" INFORMACJI DLA UZYTKOWNIKA
;.....
;SKASUJ WSKAZNIK "HOLD"
STA    HOLD
INR    B
;SPRAWDZ CZY BATON+GSD I OBSLUGA GSD
E18    CMP    C
JZ     E19
;CZY DANE PRZYJETE
PUSH   H
LXI    H,MHSW
CMP    M
JNZ    E50
;W TYM MIEJSCU POWINIEN BYC FRAGMENT PROGRAMU POWODUJACY
;ZWOLNIENIE INFORMACJI "GLOBAL SEND DATA" (GSD)
;.....
;SKASUJ WSKAZNIK GSD
MOV    C,A
POP    H
;ZMNIJSZ LICZNIK POWTORZEN TRANSMISJI
E50    DCR    D
;DALSZE FRAGMENTY PROGRAMU SA NIEZDEFINIOWANE
;.....

```

6. Określenie czasu wykonania krytycznego fragmentu  
protokołu PROWAY

Program realizujący krytyczny fragment protokołu PROWAY zamieszczony w rozdz.5 nie był uruchamiany. Czas potrzebny na jego wykonanie został określony teoretycznie. Przy obliczeniach wykorzystano podane przez producenta mikroprocesora INTEL 8080 długości wykonania instrukcji w cyklach zegarowych.

Jednocześnie założono maksymalną dostępną częstotliwość zegara wewnętrznego mikroprocesora wynoszącą 2 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 bufony dostępne dla innych urządzeń.

Czasy wykonania kolejnych części programu określono poniżej.

- 1/ Od otrzymania przerwania do gotowości do odczytu pierwszego bajtu potrzebny jest czas równy  $64\mu s$  plus czas sprzętowej obsługi przerwania. Jeżeli czas obsługi przerwania będzie większy od  $16\mu s$  to będzie przekroczony czas odbioru pierwszego bajtu i kontroler w tym fragmencie programu przestanie nadać za informacją przychodzącą z linii.
- 2/ Od odczytu pierwszego bajtu do gotowości do odczytu drugiego bajtu potrzebny jest czas  $59\mu s$ . Daje to rezerwę  $21\mu s$ , która pozwala na "dogonienie" transmisji po opóźnieniu wynikłym przy obsłudze przerwania.
- 3/ Od odczytu drugiego bajtu do gotowości do odczytu trzeciego bajtu wymagany jest czas max.  $88\mu s$ . Kontroler znowu przestaje nadać za transmisją.
- 4/ Od odczytu trzeciego bajtu do stwierdzenia faktu zakończenia transmisji ramki potrzebny jest czas  $272\mu s$ . W przypadku, gdy przesyłka zawiera tylko "baton do



nie" zostaje przekroczony czas odbioru końcowego fragmentu ramki wynoszący  $240\mu s$  /3 bajty/.

5/ Od zakończenia odbioru ramki do rozpoczęcia nadawania wymagany jest czas  $119\mu s$ .

W najbardziej niekorzystnym przypadku limit czasu równy 100 s zostanie przekroczony o różnicę czasów wynikającą z punktów 3,4 i 5 równe  $= 88-80+272 - 240 + 119-100=59\mu s$ . Rozważano również możliwość wcześniejszego wysłania bajtu synchronizacyjnego /przed zakończeniem przygotowania całej ramki/ co dałoby dodatkowe  $80\mu s$ .

Wymagałoby to wprowadzenia pewnych zmian w komunikacji kontroler - sterownik w stosunku do założeń podanych w rozdziale 2. Zostałby również wydłużony fragment programu od zakończenia odbioru ramki do rozpoczęcia nadawania /do zakończenia przygotowania ramki do wyśłania - już w trakcie transmisji bajtu synchronizacyjnego/. Uzyskany czas dodatkowy wynosiłby  $80\mu s$ . Oszacowano konieczne wydłużenie programu na 3 rozkazy po 8 cykli zegarowych tj.  $12\mu s$ .

W ten sposób analizowany program zdążyłby się wykonać w limicie czasu z rezerwą wynoszącą  $80-59-12 = 9\mu s$ .

Czas wykonania całego analizowanego programu jest równy  $64 + 59 + 88 + 272 + 119 + 12 = 614\mu s$ .

Otrzymana rezerwa wynosi więc ok. 1,5 % całkowitego czasu wykonania programu. Uwzględniając wszystkie wymienione wyżej zastrzeżenia wpływające na dokładność metody, należy przyjąć, że rezerwa ta jest zbyt mała aby analizowany program mógł się w rzeczywistości wykonać w zadanym limicie czasu,

## 7. Podsumowanie

Tematem pracy było znalezienie odpowiedzi na pytanie czy mikroprocesor INTEL 8080 może być zastosowany jako procesor kontrolera komunikacyjnego stacji systemu PROWAY.

Na podstawie przeprowadzonej analizy stwierdzono, że przy wykorzystaniu tego mikroprocesora nie mogą być zrealizowane działania przewidziane w protokole PROWAY w zadanych limitach czasu.

Odpowiedź na pytanie postawione w temacie pracy jest więc negatywna.