

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

440
Główny wykonawca mgr inż. Andrzej Aderek

Wykonawcy mgr inż. Krzysztof Czarnomski

Konsultant dr inż. Andrzej Syrczyński

Nr zlecenia
9436
Etap 1

Analiza możliwości realizacji
protokołów MIR-PROWAY na dwuprocesso-
rówym układzie 8080-8035. oraz
na procesorze 8086.
Etap 1. Analiza możliwości realizacji
protokołów MIR-PROWAY na dwu-
processorowym układzie 8080-8035.

Zleceniodawca praca własna

Pracę rozpoczęto dnia 08.83r.
Kierownik Pracowni

mgr inż. A. Aderek

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

dr inż. T. Gałązka

zakończono dnia 30.09.83
Kierownik Ośrodka

prof. dr inż. T. Missala

Praca zawiera:

stron 25

rysunków 4

fotografii -

tabel -

tablic -

załączników -

Rozdzielnik - ilość egz:

Egz. 1 BOINTE

Egz. 2 OAE-8

Egz. 3 OBN

Egz. 4 OAE-83

Egz. 5

Egz. 6

Nr rejestr. 5082

Analiza deskryptorowa

OPROGRAMOWANIE + MIKROPROCESOR + MIR-PROWAY.

Analiza dokumentacyjna

Opracowanie zawiera opis programu kontrolera komunikacyjnego stacji PROWAY zbudowanego na dwuprocessorowym układzie 18080- 18035. Oszacowano czas wykonania programu i porównano go z limitem czasu określonym przez protokół MIR-PROWAY.

Tytuły poprzednich sprawozdań

UKD

MERA-PIAP/TW 331/78 5000

2

SPIS TRESCI

1. Wstęp.....	2
2. Założenia na kontroler komunikacyjny.....	2
3. Opis i schemat blokowy programów realizujących krytyczny fragment protokołu PROWAY.....	4
4. Listingi.....	16
5. Oszacowanie czasu wykonania krytycznego fragmentu protokołu PROWAY.....	23
6. Podsumowanie.....	25
7. Literatura.....	25

1. Wstęp

Niniejsze sprawozdanie jest kontynuacją przygotowań do skonstruowania i wdrożenia systemu wymiany informacji w rozłożonych systemach sterowania.

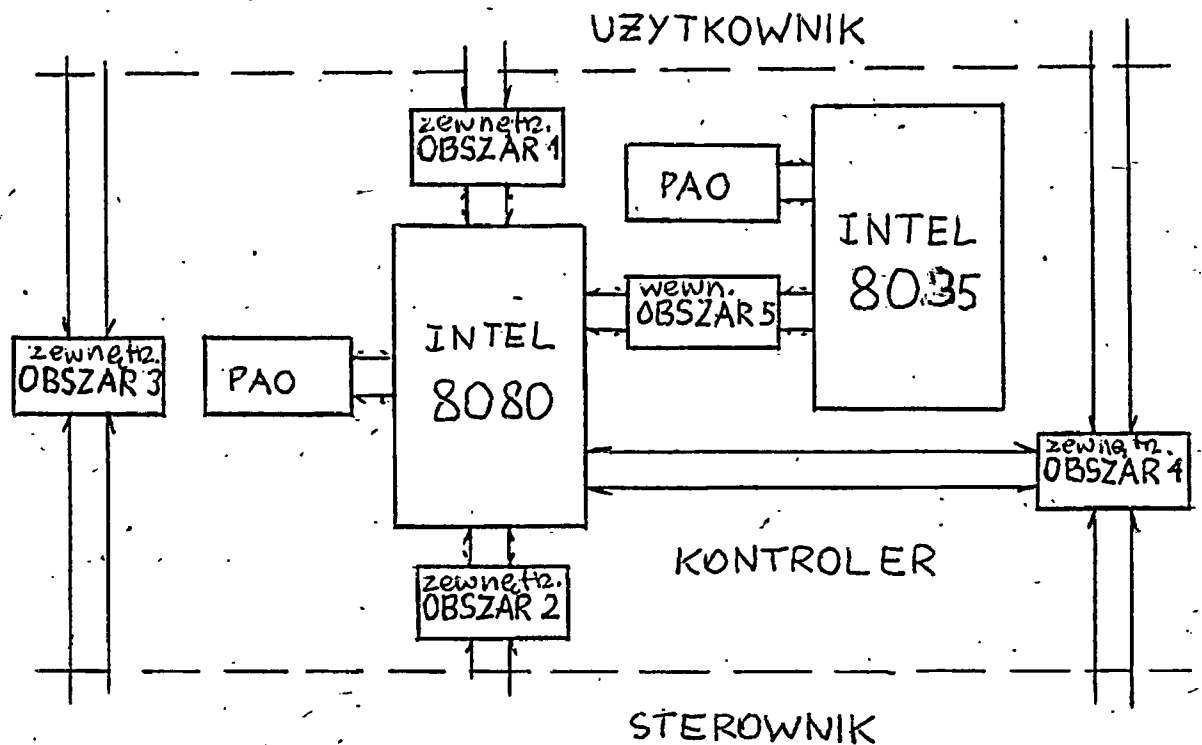
Efektom poprzednio wykonanych prac [1] było stwierdzenie, że na mikroprocesorze INTEL 8080 nie jest możliwe wykonanie oprogramowania realizującego protokół systemu PROWAY. Czasy wykonania programów przekraczałyby limity stawiane przez projekt normy PROWAY-a. Celem niniejszego etapu jest przeanalizowanie możliwości wykonania powyższego oprogramowania na dwuprocessorowym układzie INTEL 8080 - INTEL 8035.

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

Założenia dotyczące zasady działania kontrolera komunikacyjnego stacji PROWAY zostały przedstawione w pracy [1]. Zaproponowany tam sposób współpracy kontrolera z innymi elementami stacji został przyjęty bez zmian do przeprowadzonej w bieżącym etapie analizy. Jedyne różnice dotyczą wewnętrznej struktury kontrolera. Tam był on pojedynczym mikroprocesorem z pamięcią mającą wspólne ^{zewnetrzne} obszary z otoczeniem, przez które była realizowana łączność z sąsiednimi elementami stacji. Tu zaś jest układem dwuprocessorowym. Wszystkie obszary wspólne pamięci opisane w pracy [1] pozostają zachowane. Dodany zostaje wewnętrzny obszar pamięci wspólny dla dwóch procesorów kontrolera, służący do komunikacji między nimi. Oprócz tego każdy z mikroprocesorów dysponuje własną, dostępną tylko dla niego, pamięcią. Ten podział pamięci zapewnić na możliwie jaknajszybszy dostęp do niej. Czas dostępu do pamięci wspólnej dla wszystkich urządzeń byłby znacznie dłuższy niż jest w proponowanym układzie, co jest nie do przyjęcia ze względu na konieczność zachowania limitów czasu.

Rysunek 1 przedstawia schematycznie układ procesorów i pamięci kontrolera komunikacyjnego.



Rys.1. Schemat funkcjonalny kontrolera komunikacyjnego.

Bufory zawarte we wspólnych ^{zewnętrznych} obszarach pamięci 1-4 i sposoby komunikacji pomiędzy użytkownikiem a kontrolerem oraz kontrolerem a sterownikiem są opisane w pracy [1]. Zakładając dwuprocessorowy układ kontrolera, należało przede wszystkim wykonać podział zadań kontrolera pomiędzy mikroprocesory INTEL 8080 i 8035 tak, aby mogły pracować współbieżnie. Liczba przekazywanych pomiędzy obu mikroprocesorami danych powinna być możliwie mała. Pozwala to uniknąć opóźnień związanych z korzystaniem ze wspólnej ^{wewnętrznej} pamięci oraz uniknąć sytuacji związanych z "wyścigiem" urządzeń i konieczności synchronizacji. Przy podziale zadań brano pod uwagę tylko działania wykonywane przez kontroler w czasie tzw. "ścieżki krytycznej" - tj. fragmentu programu kontrolera realizującego elementy protokołu PROWAY-a stawiające najwyższe i najtrudniejsze do spełnienia wymagania czasowe.

Ścieżka krytyczna została określona w ramach analizy możliwości wykonania oprogramowania kontrolera na mikroprocesorze 8080 /p.praca [1] /. Stanowi ją fragment protokołu PROWAY-a obejmując

- analizę odbieranej ze sterownika ramki,
- aktualizację Live List,
- określenie Next Gap Address i Next Live Address /patrz [2] /,
- kontrolę zleceń użytkownika,
- przesłanie ramki do Next Gap lub Next Live Address.

Powyższe działania dzielą się na dwie grupy: odczyt i prostą analizę odbieranych z zewnątrz informacji oraz opracowywanie Live List. Operacje należące do pierwszej z wymienionych grup wymagają częstego dostępu do wspólnych obszarów pamięci.

Opracowywanie Live List jest to natomiast stosunkowo skomplikowana obróbka niewielkiej ilości danych.

Jej realizacja trwa mniej więcej połowę całego czasu wykonania programu ścieżki krytycznej kontrolera zrealizowanego na jednym mikroprocesorze 8080.

Biorąc to wszystko pod uwagę dokonano następującego wyboru : procesor 8080 będzie realizował działania z wymienionej wyżej pierwszej grupy tzn. będzie analizował odebraną ze sterownika ramkę, badał zlecenia użytkownika i inicjował nadawanie ramki przez sterownik. Procesor ten będzie komunikował się z urządzeniami zewnętrznymi przez ^{zewnętrzne} obszary wspólne pamięci nr 1,2,4.

Mikroprocesor 8035 będzie aktualizował Live List oraz przygotowywał Next live Address i Next Gap Address. Potrzebne do pracy dane układ 8035 będzie otrzymywał z procesora 8080 przez ^{wewnętrzny} wspólny obszar pamięci nr 5 /p. rys.1/.

3. Opis i schemat blokowy programów realizujących krytyczny fragment protokołu PROWAY.

Opis omawianych programów został przedstawiony w pracy [1]. Pozostaje on aktualny w odniesieniu do rozpatrywanego tu kontrolera dwuprocessorowego z następującymi zmianami:

- aktualizacja Live List jest wykonywana przez drugi mikroprocesor I8035 pozostałe działania wykonuje I8080,
- pierwszy procesor I8080 przekazuje drugiemu /I8035/ adres źródłowy odbieranej ramki zawierającej "baton" bezpośrednio po otrzymaniu go ze sterownika, w ten sposób inicjując przygotowanie aktualizacji Live List,

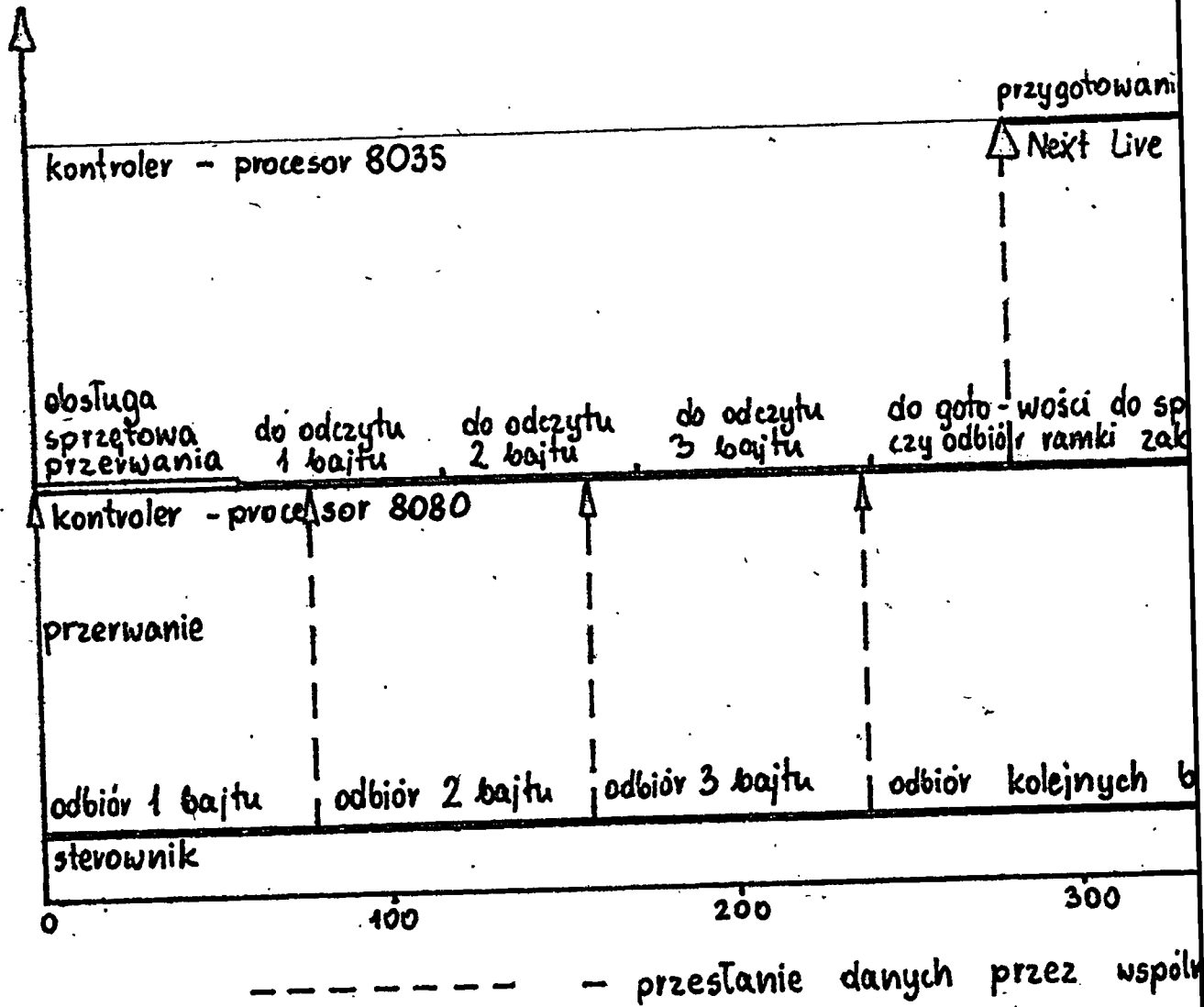
- drugi procesor w stanie jałowym cyklicznie sprawdza, czy został mu podany we wspólnym ^{wewnętrznym} obszarze pamięci adres do aktualizacji /jeżeli nie - sprawdzone słowo we wspólnym ^{wewnętrznym} obszarze pamięci ma wartość charakterystyczną = 0/.
- w trakcie odbioru czwartego i następnych bajtów drugi procesor przygotowuje aktualizację Live List, a pierwszy w tym samym czasie sprawdza żądanie użytkownika,
- po zakończeniu odbioru ramki i kontroli poprawności transmisji pierwszy procesor odczytuje Next Live i Next Gap Address ^{wewnętrznym} przygotowane już przez drugi procesor i zapisane we wspólnym obszarze pamięci i zleca sterownikowi rozpoczęcie transmisji przesyłki według sprawdzonego uprzednio żądania użytkownika,
- opisana w pracy [1] organizacja Live List pozostaje bez zmian, tylko jest realizowana przez drugi procesor,
- aktualizacja Live List odbywa się w dwóch etapach - przygotowania i dokończenia. Pierwszy z nich jest wykonywany w trakcie odbioru ramki. Po zakończeniu odbioru i sprawdzeniu jego poprawności następuje dokończenie aktualizacji /gdy odbiór był prawidłowy/ lub powrót do stanu sprzed jej rozpoczęcia /gdy odbiór był nieprawidłowy/.

Prawidłowość współpracy obu procesorów kontrolera komunikacyjnego jest zapewniona dzięki temu, że procesor pierwszy INTEL 8080 inicjuje aktualizację w czasie, kiedy procesor drugi INTEL 8035 jest zawieszony w stanie "jałowym" - wszystkie czynności związane z poprzednią aktualizacją zostały zakończone wcześniej /ich realizacja trwa 130 μ s od momentu zakończenia poprzedniej transmisji, natomiast minimalny czas jaki upływa od tego momentu do chwili ponownej inicjacji równy jest sumie czasów przerwy w linii i transmisji pierwszych czterech bajtów i wynosi $100\mu s + 320\mu s = 420\mu s$. Z kolei procesor drugi wykonuje pierwszy etap /przygotowanie/ aktualizacji i wysyła Next Live oraz Next Gap Address do pamięci dostępnej dla pierwszego procesora w czasie 213 μ s. Pierwszy procesor odczyta te adresy po minimalnie $240 + 53 - 43 = 250\mu s$, od inicjacji aktualizacji Live List, gdzie: 240 μ s - czas transmisji 3 bajtów - minimalnej długości części ramki odbieranej, kiedy już trwa aktualizacja,

53 μ s - czas wykonywania programu pierwszego procesora od zakończenia transmisji do odczytu z pamięci przygotowanych przez drugi procesor adresów,

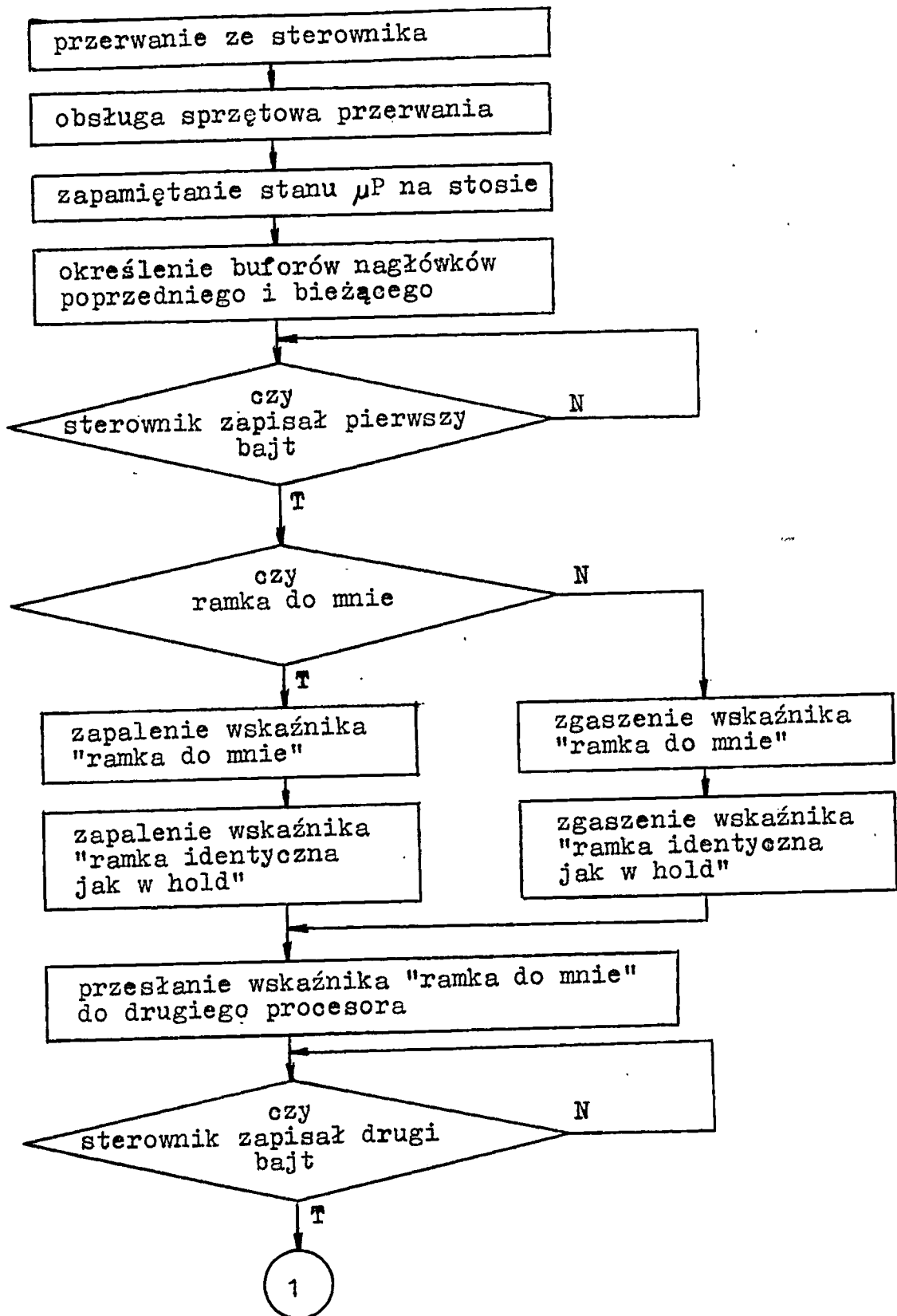
43 μ s - czas wykonywania programu pierwszego procesora od zakończenia odbioru bajtu z adresem źródła odbieranej ramki do inicjacji drugiego procesora.

Z porównania obu czasów /213 μ s i 250 μ s/ wynika, że drugi procesor zdąży przygotować żądane dane przed odczytaniem ich przez procesor pierwszy. W ten sposób jest zapewniona poprawna jednoczesna praca obu procesorów. Przebieg czasowy ich współpracy przedstawia rys.2. Działania wykonywane przez ich programy przedstawiają schematy blokowe na rys. 3 i 4. Na schemacie programu procesora pierwszego INTEL 8080 /rys.3/ zaznaczone zostały odgałęzienia "donikąd", będące potencjalnymi połączeniami napisanego programu z całością oprogramowania kontrolera komunikacyjnego. Są one oznaczone dużymi literami.

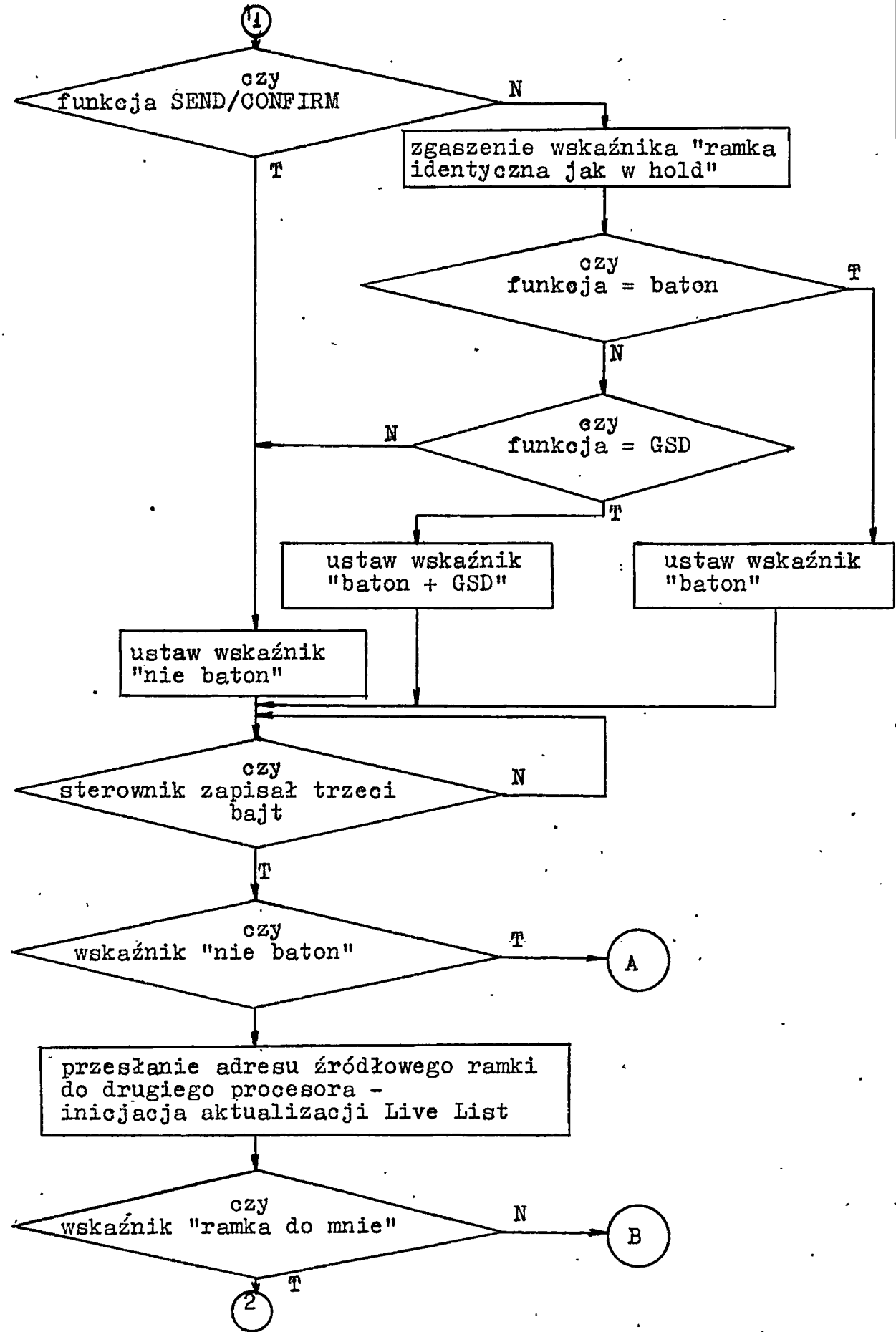


Rys 2. Przebieg współpracy mikroprocesorów kontrolera k

SCHEMAT BLOKOWY PROGRAMU PIERWSZEGO PROCESORA INTEL 8080

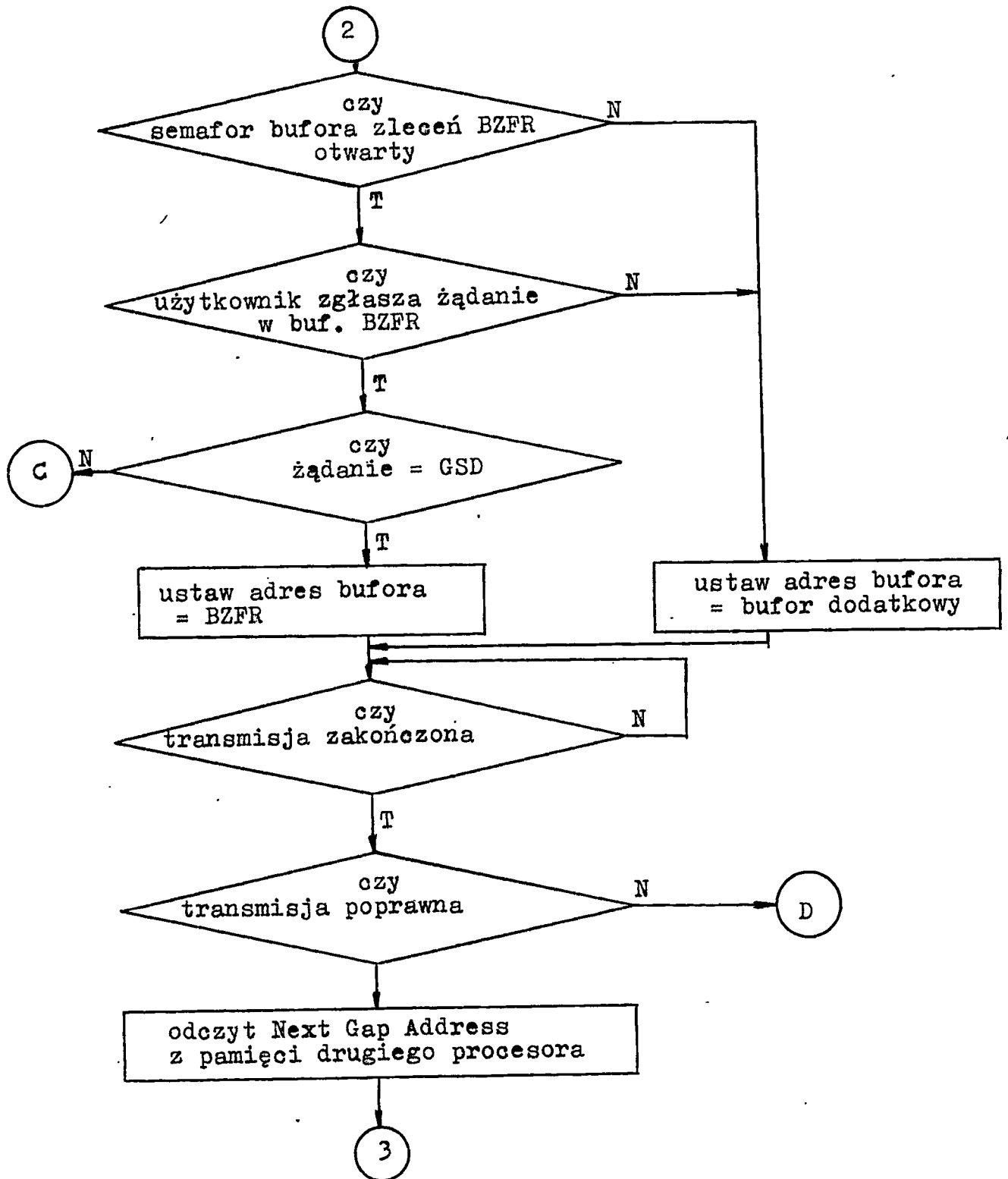


SCHEMAT BLOKOWY PROGRAMU PIERWSZEGO PROCESORA INTEL 8080

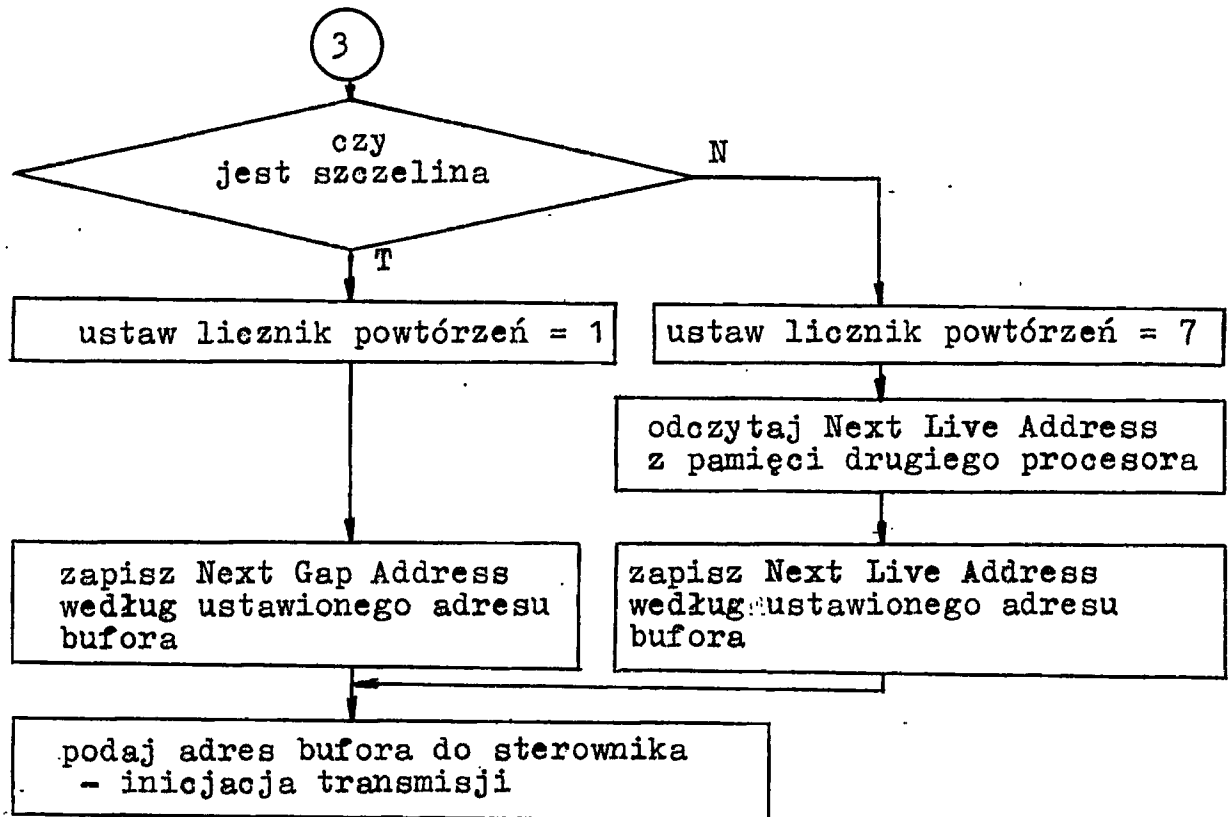


11

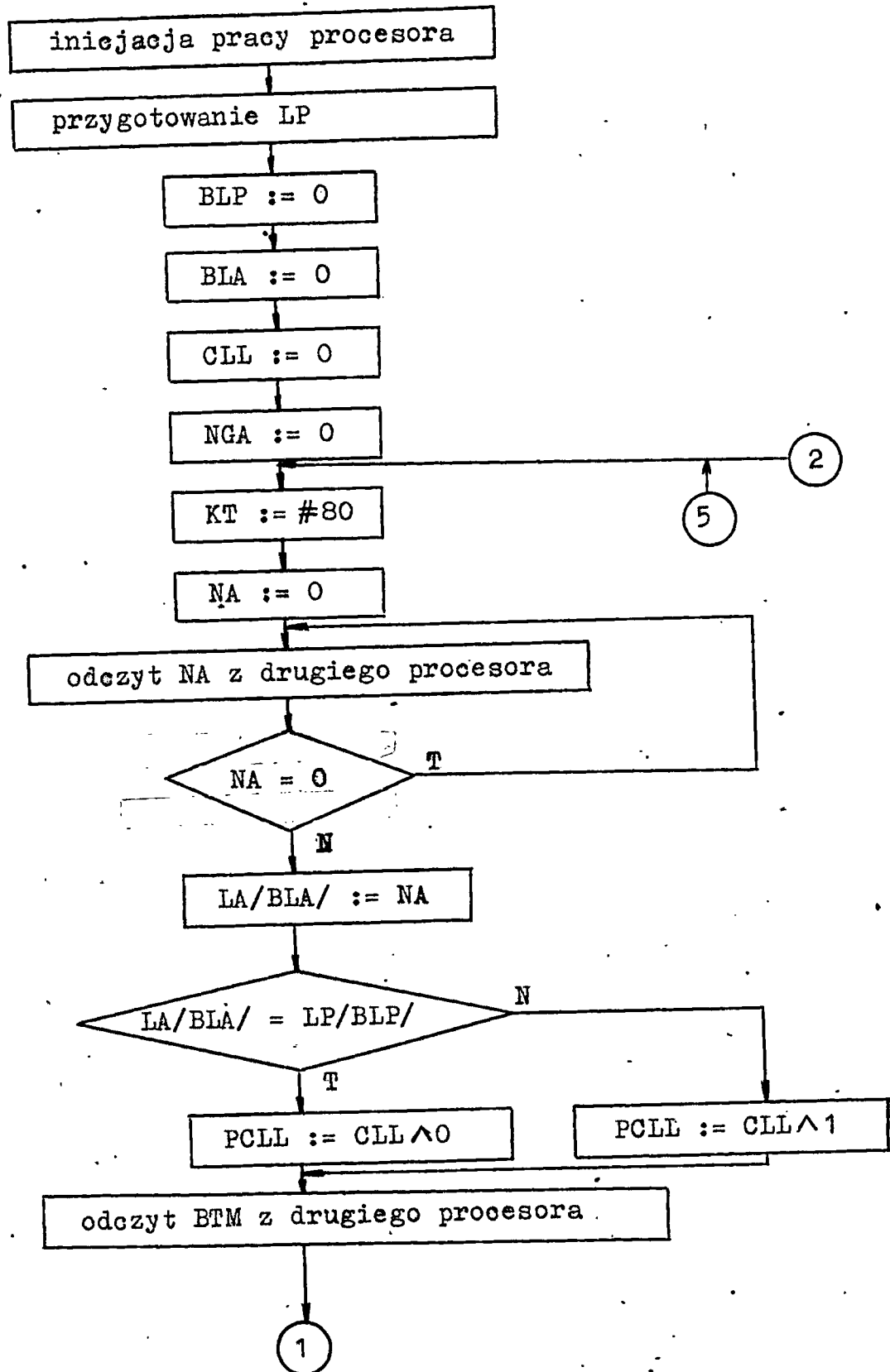
SCHEMAT BLOKOWY PROGRAMU PIERWSZEGO PROCESORA INTEL 8080



SCHEMAT BLOKOWY PROGRAMU PIERWSZEGO PROCESORA INTEL 8080

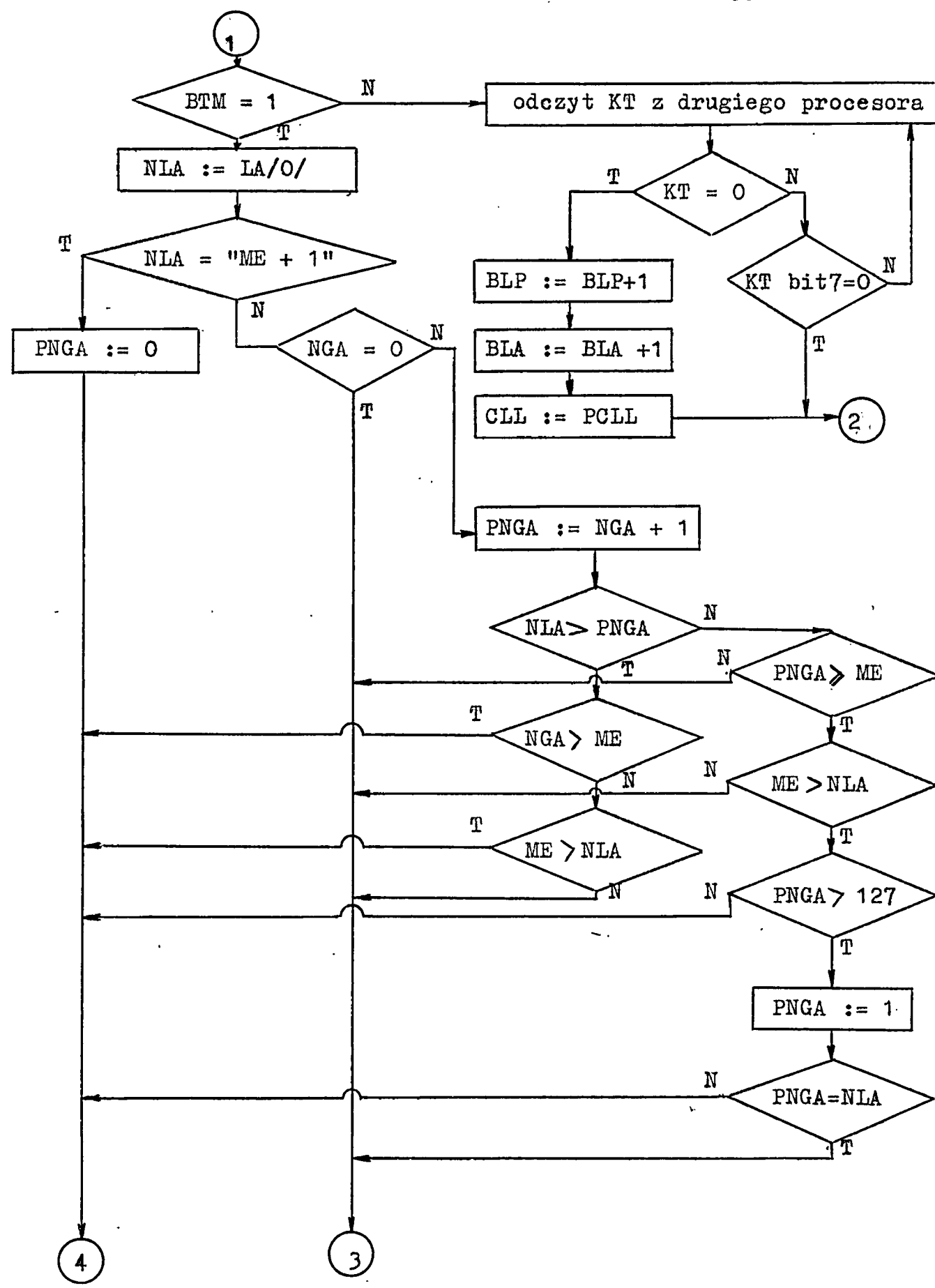


SCHEMAT BŁOKOWY PROGRAMU DRUGIEGO PROCESORA INTEL 8035

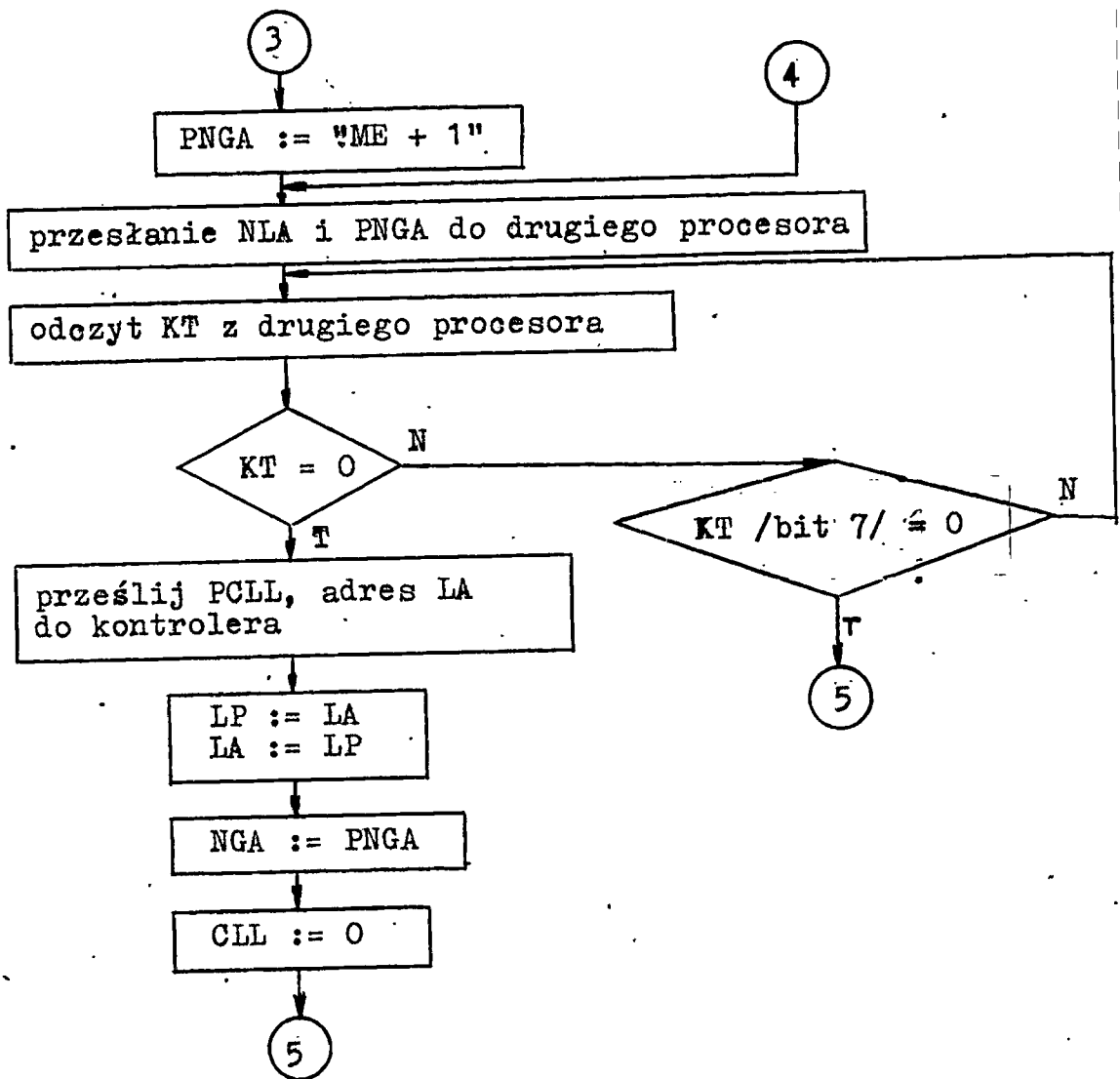


14

SCHEMAT BLOKOWY PROGRAMU DRUGIEGO PROCESORA INTEL 8035



SCHEMAT BLOKOWY PROGRAMU DRUGIEGO PROCESORA INTEL 8035



Oznaczenia na schemacie blokowym programu procesora INTEL 8035:

- LA - aktualnie tworzona Live List,
LP - poprzednia Live List
BLA - numer bieżącej pozycji w LA
BLP - numer bieżącej pozycji w LP
LA/BLA/ - zawartość pozycji w LA wskazywanej przez BLA
LP/BLP/ - zawartość pozycji w LP wskazywanej przez BLP
ME - mój adres /numer "mojej" stacji PROWAY/
"ME+1" - kolejno następujący po moim adres stacji PROWAY.
"ME+1" nie zawsze jest równy ME+1 jeżeli ME=127 do "ME+1"=1
NLA - Next Live Address
NGA - Next Gap Address
PNGA - Next Gap Address przygotowywany w pierwszej fazie aktualizacji Live List
CLL - wskaźnik zmiany Live List
=0 gdy nie ma zmian
=1 gdy są zmiany
PCLL - CLL przygotowywany w pierwszej fazie aktualizacji Live List,
BTM - wskaźnik "baton do mnie" ustawiony przez pierwszy procesor INTEL 8035 we wspólnym obszarze pamięci
=0 gdy baton nie do mnie
=1 gdy baton do mnie
KT - wskaźnik zakończenia transmisji, we wspólnym obszarze pamięci przed oczekiwaniem na koniec transmisji, drugi procesor nada mu wartość ≠80, po zakończeniu transmisji pierwszy procesor przesyła do niego odczytane ze sterownika słowo stanu,
NA - nowy adres, wskaźnik uruchamiający przygotowanie aktualizacji Live List; we wspólnym obszarze pamięci, jego wartość początkowa wynosi 0, po odebraniu czwartego bajtu pamki typu "baton" z linii pierwszy procesor przesyła do NA otrzymany adres źródłowy.

4. Listingi

Listing programu pierwszego procesora INTEL 8080

```
;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    ABPR
        SHLD   LHD
        LHL    ABAR
        SHLD   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
E3      MOV    A,D      ;PRZESLIJ WSKAZNIK "RAMKA DO MNIE"
        STA    BTM     ;DO PROCESORA L.L.
;CZY STEROWNIK ZAPISAL DRUGI BAJT
        MVI    B,1
E3A     LDA    LBAR
        CMP    B
        JZ     E3A
;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
;PRZYGOTOWANIE DO ODBIORU TRZECIEGO BAJTU
E5      INX      H
;CZY STEROWNIK ZAPISAL TRZECI BAJT
E8      LDA      LBAR
        CPI      2
        JZ       E8
;CZY FUNKCJA ZAWIERA BATON
        XRA      A
        CMP      C
        JC       E9          ;WYJSCIE "DO NIKAD"
;JEZELI FUNKCJA ZAWIERA BATON - AKTUALIZACJA LIVE LIST
        MOV      A,M          ;PRZESLIJ SOURCE ADRESS
        STA      NA          ;DO PROCESORA L.L.
;CZY BATON DO MNIE
        XRA      A          ;ZERUJ A
        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        ;GDY ZADANIE=GSD

```

```
;UZYTKOWNIK NIE ZGLASZA ZADANIA LUB SEMAF. ZAMKNIETY
E13   LXI   H, BRB
;CZY ODBIOR RAMKI ZAKONCZONY
;SLOWO STANU STEROWNIKA - SST
E15   LDA   SST
      RRC
      JC    E15
;CZY ODBIOR ZAKONCZONY POPRAWNIE
      STA   MHSW
      ANI   #07
      STA   KT      ;PRZESLIJ KT DO DRUGIEGO PROCESORA
      CPI   0
      JNZ   E11     ;GDY TRANSMISJA BLEDNA
;CZY JEST SZCZELINA
      LDA   NGA     ;ODCZYT NGA Z PROCESORA L.L.
      CPI   0
      JZ    E16
;GDY JEST SZCZELINA
      MOV   M,A
;USTAW LICZNIK POWTORZEN (REJ D)
      MVI   D,1
      JMP   E17
;GDY NIE MA SZCZELINY
E16   LDA   NLA     ODCZYT NLA Z PROCESORA L.L.
      MOV   M,A
;USTAW LICZNIK POWTORZEN
      MVI   D,7
;PODAJ ADRES BUFORA DO STEROWNIKA
E17   SHLD  ABF
;ABF ADRES SLOWA NA ADRES BUFORA PRZEKAZYWANEGO STEROWNIKOWI
;W TYM MIEJSCU POWINIEN BYC FRAGMENT PROGRAMU POWODUJACY
;ZWOLNIENIE Z "HOLD" INFORMACJI DLA UZYTKOWNIKA
;.....
;SPRAWDZ CZY BATON+GSD I OBSLUGA GSD
;CZY DANE PRZYJETE
;W TYM MIEJSCU POWINIEN BYC FRAGMENT PROGRAMU POWODUJACY
;ZWOLNIENIE INFORMACJI "GLOBAL SEND DATA" (GSD)
;.....
;ZMNIEJSZ LICZNIK POWTORZEN, TRANSMISJI
;DALSZE FRAGMENTY PROGRAMU SA NIEZDEFINIOWANE
;.....
```

Listing programu drugiego procesora INTEL 8035

```

;PRZYGOTOWANIE WSTEPNE
  SEL RB1      ;WYBIERZ GRUPE REJESTROW 1
  MOV R7,LIST2 ;R7:=ADRES LISTY POPRZEDNIEJ
;PRZYGOTOWANIE LISTY POPRZEDNIEJ DO INICJACJI
;PO INICJACJI ZAKLADAMY ZE WSZYSTKIE STACJE SA AKTYWNE
;KOLEJNE KOMORKI ZAWIERAJA KOLEJNE ADRESY OD ME+1 DO ME
  MOV A,R7
  MOV R0,A     R0:=ADRES PRZYGOTOWYWANEJ LISTY POPRZ
  MOV R4,#1    ;R4:=1 DO ZWIEKSZANIA ADRESOW
  MOV R5,#7F   ;R5:=127 LICZNIK POWTORZEN PETLI
  MOV R1,ME
  MOVX A,@R1   ;A:=MOJ ADRES
;W PETLI - ZAPISANIE KOLEJNYCH ADRESOW DO LISTY
ET2  ADD A,R4   ;ZWIEKSZ ADRES ZAPISYWANY
     JB7 ET1A  ;SKOK GDY ADRES GT 127
     JMP ET1
ET1A CLR A     ;GDY ADRES GT 127
     INC A    ;A:=1
ET1  MOVX @R0,A ;ZAPISZ ADRES W LISCIE
     INC R0   ;ZWIEKSZ ADRES DO ZAPISYWANIA
     DJNZ R5,ET2 ;LICZNIK POWTORZEN PETLI
;KONIEC PRZYGOTOWYWANIA LISTY POPRZ PO INICJACJI
;USTAWIENIE WARTOSCI POCZATKOWYCH REJESTROW
  MOV R1,LIST1 ;R1:=ADRES LISTY BIEZACEJ
  SEL RBO
  MOV R2,A     ;R2:=MOJ ADRES
  SEL RB1
  MOV R2,A     ;R2(GR 1):=MOJ ADRES
  MOV A,R7
  SEL RBO
  MOV R0,A     ;R0:=ADRES LISTY POPRZEDNIEJ
  SEL RB1
  MOV A,R1
  SEL RBO
  MOV R1,A     ;R1:=ADRES LISTY BIEZACEJ
  CLR A
  MOV R5,A     ;R5:=0 TWORZONY WSKAZNIK ZMIANY L.L.
  MOV R4,A     ;R4:=0 NEXT GAP ADDRESS "Z POPRZ OBIEGU"
  SEL RB1
  MOV R4,A     ;R4(GR 1):=NEXT GAP ADDRESS "Z POPRZ OBIEGU"
  SEL RBO
  MOVX A,@R0
  MOV R3,A     ;R3:=NASTEPNIK PO MOIM ADRESIE
  SEL RB1
  MOV R3,A     ;R3(GR 1):=NASTEPNIK PO MOIM ADRESIE
  SEL RBO
;KONIEC PRZYGOTOWANIA WSTEPNEGO (INICJACJI)

```

```

;OCZEKIWANIE NA POJAWIENIE SIE NOWEGO ADRESU Z KONTROLERA
ET7  MOVX A,@R0
     MOV R6,A           ;R6:=NEXT LIVE ADDRESS PO OSTATNIO ZAPISANYM ADRESIE
     SEL RB1           ;WYBIERZ GRUPE REJESTROW 1
     MOV R0,KT         ;KT-WSKAZNIK ZAKONCZENIA TRANSMISJI
     MOV A,#80
     MOVX @R0,A        ;(KT):=10000000 (BIT 7 =1)
     MOV R0,NA         ;NA-ADRES GDZIE KONTROLER ZAPISUJE NOWY ADRES
     CLR A
     MOVX @R0,A        ;(NA):=0 ZERUJ MIEJSCE NA NOWY ADRES
ET3  MOVX A,@R0        ;ODCZYT ZAPISYWANEGO NA
     JZ ET3           ;SKOK GDY NIC NIE ZAPISANO
;GDY POJAWIL SIE NOWY ADRES
;ZAPISANIE GO DO BIEZACEJ LISTY
     SEL RB0           ;WYBIERZ GRUPE REJESTROW 0
     MOVX @R1,A        ;ZAPISZ NOWY ADRES DO TWORZONEJ LISTY
;POROWNANIE NOWEGO ADRESU ZE STARYM
     MOV R7,#0
     CPL A             ;NEGACJA NOWEGO ADRESU
     ANL A,R6         ;POROWNANIE
     JZ ET4
     INC R7
ET4  MOV A,R7           ;GDY ADRESY NIEROWNE A:=1, INACZEJ A:=0
     ORL A,R5
     SEL RB1
     MOV R5,A         ;PRZYGOTUJ NOWY WSKAZNIK ZMIANY L.L.
;CZY BADANY BATON JEST DO MNIE
     MOV R0,BTH        ;BTM-WSKAZNIK "BATON DO MNIE"
     MOVX A,@R0        ;BTM=0-NIE DO MNIE, 1-DO MNIE
     JNZ ET5
;GDY BATON NIE DO MNIE
;CZY TRANSMISJA ZAKONCZONA POPRAWNIE
     MOV R0,KT
ET8  MOVX A,@R0        ;A:=KT
     JZ ET8A          ;SKOK GDY TRANSM ZAKONCZ POPRAWNIE
     JB7 ET8          ;SKOK GDY TRANSM NIE ZAKONCZONA
     SEL RB0
     JMP ET7          ;SKOK GDY TRANSMISJA BLEDNA
;GDY TRANSMISJA ZAKONCZONA POPRAWNIE
ET8A MOV A,R5
     SEL RB0
     MOV R5,A
     INC R0
     INC R1
     JMP ET7
;GDY BATON DO MNIE
;OKRESLENIE NEXT GAP ADDRESS I NEXT LIVE ADDRESS
ET5  MOVX A,@R1
     MOV R6,A         ;R6:=PRZYGOTOWYWANY NEXT LIVE ADDRESS
;SPRAWDZ CZY NLA JEST ADRESEM NASTEPNYM PO MOIM
     CPL A             ;NEGACJA NLA
     ANL A,R3         ;POROWNANIE Z NASTEPNIKIEM
     JNZ ET6         ;SKOK GDY NLA NEQ ME+1 (JEST GAP)

```

```

;GDY NIE MA GAP
  CLR A
  MOV R4,A          ;R4:=0 NEXT GAP ADDRESS
  JMP ET12
;GDY JEST GAP
ET6   MOV A,R4
      JZ ET11      ;SKOK GDY POPRZEDNIO NIE BYLO GAP
;GDY POPRZEDNIO BYL GAP
      INC R4       ;ZWIEKSZ NGA
;CZY ZWIEKSZONY NGA MIESCI SIE W GAP
;CZY NLA GT NGA
      MOV A,R6     ;A:=NLA
      CPL A
      INC A        ;A:=-NLA
      ADD A,R4
      JB7 ET10     ;SKOK GDY NLA GT NGA
;CZY NGA GT ME
      MOV A,R4     ;A:=NGA

      CPL A
      INC A
      ADD A,R2
      CPL A
      JB7 ET11     ;SKOK GDY NGA LE ME
;CZY ME GT NLA
      MOV A,R6     ;A:=NLA
      CPL A
      INC A
      ADD A,R2
      JB7 ET11     ;SKOK GDY ME LT NLA
;CZY NGA GT 127
      MOV A,R4     ;A:=NGA
      CPL A
      JB7 ET12     ;SKOK GDY NGA LE 127
      MOV R4,#1   ;NGA:=1
;CZY NGA=1=NLA
      MOV A,R6     ;A:=NLA
      DEC A        ;A:=A-1:=NLA-1
      JZ ET11     ;SKOK GDY NLA=1=NGA
      JMP ET12    ;SKOK GDY NLA NE NGA
;GDY NLA GT NGA
;CZY NGA GT ME
ET10  MOV A,R4     ;A:=NGA
      CPL A
      INC A
      ADD A,R2
      JB7 ET12     ;SKOK GDY NGA GT ME
;CZY ME GT NLA
      MOV A,R2     ;A:=ME
      CPL A
      INC A
      ADD A,R6
      JB7 ET12     ;SKOK GDY ME GT NLA

```

```
;GDY NGA NIE MIESCI SIE W GAP
;NASTEPNIK (ME+1)STAJE SIE NGA
ET11  MOV A,R3
      MOV R4,A          ;R4:=R3 (NASTEPNIK)
      MOV RO,NGA
      MOVX BR0,A        ;PRZESLIJ NGA DO KONTROLERA
      MOV A,R6
      MOV RO,NLA
      MOVX BR0,A        ;PRZESLIJ NLA DO KONTROLERA
;CZY TRANSMISJA ZAKONCZONA POPRAWNIE
ET12  MOV RO,KT
ET9   MOVX A,BR0        ;ODCZYT KT Z KONTROLERA
      JZ ET9A          ;SKOK GDY TRANSMISJA ZAKONCZONA
      JB7 ET9          ;SKOK GDY TRANSMISJA NIE ZAKONCZONA
      JMP ET13         ;SKOK GDY TRANSMISJA BLEDNA
;GDY TRANSMISJA ZAKONCZONA POPRAWNIE
ET9A  MOV A,R1
      MOV RO,ALL
      MOVX BR0,A        ;PRZESLIJ ADRES LISTY DO KONTROLERA
      MOV A,R5
      MOV RO,CLL
      MOVX BR0,A        ;PRZESLIJ WSK ZMIANY L.L. DO KONTROLERA
;ZAMIANA LIST I USTAWIENIE POCZATKOWE
      MOV A,R1          ;A:=ADRES LISTY OSTATNIO BIEZACEJ
      XCH A,R7          ;ZAMIANA A Z R7 LISTY POPRZ I BIEZACEJ
      MOV R1,A
;PRZESLANIE REJESTROW DO GRUPY 0
      SEL RBO
      MOV R1,A          ;R1:=ADRES NOWEJ LISTY BIEZACEJ
      SEL RB1
      MOV A,R7
      SEL RBO
      MOV RO,A          ;RO:=ADRES NOWEJ LISTY POPRZEDNIEJ
      SEL RB1
      MOV A,R4
      SEL RBO
      MOV R4,A          ;R4:=NGA
      SEL RB1
      MOV A,R6
      SEL RBO
      MOV R6,A          ;R6:=NLA
      CLR A
      MOV R5,A          ;R5:=0 TWORZONY WSKAZNIK ZMIANY L.L.
      JMP ET7
;GDY TRANSMISJA DO MNIE ZAKONCZONA BLEDNIE
ET13  SEL RBO
      MOV A,R4
      SEL RB1
      MOV R4,A          ;ODTWORZ POPRZEDNI NGA
      JMP ET7          ;SKOK DO OCZEKIWANIA NA NOWY ADRES
```


5. Oszacowanie czasu wykonania krytycznego fragmentu protokołu
PROWAY

Programy realizujące krytyczny fragment protokołu PROWAY nie były uruchamiane. Czasy potrzebne na ich wykonanie zostały określone teoretycznie. Przy obliczeniach wykorzystano podane przez producenta mikroprocesorów INTEL 8080 i INTEL 8035 długości wykonania instrukcji w cyklach zegarowych.

Założono maksymalne dostępne częstotliwości zegarów mikroprocesorów, wynoszące dla obu mikroprocesorów 2 MHz. Nie uwzględniono czasów dostępu do pamięci.

Mikroprocesor INTEL 8035 nie ma stanu WAIT i jego pamięć zewnętrzna musi nadążać za cyklami zegarowymi. W wypadku pamięci użytkowanej tylko przez dany mikroprocesor nie stanowi to trudności, gdyż INTEL 8035 jest względnie powolny. Problemy mogą wystąpić przy pamięci wielodostępnej /tu: ^{wewnętrznej} wspólnej dla obu mikroprocesorów/. Przyjęto, że problemy te zostałyby rozwiązane na drodze sprzętowej.

Czasy wykonania kolejnych części programu pierwszego mikroprocesora - INTEL 8080 - wynoszą:

- 1/ od otrzymania przerwania do gotowości do odczytu pierwszego bajtu potrzebny jest czas równy 58 μ sek + czas sprzętowej obsługi przerwania,
- 2/ od zapisania przez sterownik pierwszego bajtu do gotowości do odczytu drugiego bajtu czas wykonania programu wynosi 59 μ sek,
- 3/ od zapisania przez sterownik drugiego bajtu do gotowości do odczytu trzeciego bajtu wymagany jest czas 65 μ sek,
- 4/ od zapisania przez sterownik trzeciego bajtu do sprawdzenia, czy odbiór ramki zakończony - 94 μ sek, w tym:
 - 4a/ od zapisania przez sterownik trzeciego bajtu do wysłania adresu źródłowego odbieranej ramki do drugiego procesora - 43 μ sek,
- 5/ od zakończenia odbioru ramki do wysłania zlecenia nadawania do sterownika - 72 μ sek, w tym:
 - 5a/ od zakończenia odbioru ramki do odczytu Next Live Address i Next Gap Address przygotowanych przez drugi procesor - 53 μ sek.

Czas realizacji programu drugiego procesora INTEL 8035 - od otrzymania nowego adresu źródłowego odbieranej ramki zawierającej "baton" do zapisania obliczonych Next Gap Address i Next Live Address do wspólnej pamięci, wynosi maksymalnie 213 μ sek.

Przebieg pracy obu programów przedstawia rys.2. Pierwszy procesor wykonuje swoje zadanie w czasie jaki ma do dyspozycji i ma pewne rezerwy - zwłaszcza we fragmencie programu wymienionym w pkt-cie 4. Na wykonanie tej części programu pierwszy procesor ma conajmniej 240 μ sek, zaś wykonuje go w 94 μ sek, czyli w ciągu 39% dysponowanego czasu.

Pierwszy procesor wykonuje segmenty programu wymienione w punktach 1-4 /tzn. te, których realizacja odbywa się w czasie odbioru ramki/ w czasie 276 μ sek mając do dyspozycji czas conajmniej 480 μ sek /czas transmisji najkrótszej ramki od zakończenia bajtu synchronizacyjnego do końca/, czyli w ciągu 58% czasu.

Po zakończeniu odbioru ramki procesor wykonuje program w czasie 72 μ sek, to jest w czasie 72% czasu do dyspozycji.

W bardziej krytycznych warunkach czasowych pracuje drugi procesor kontrolera komunikacyjnego. Na przygotowanie aktualizacji Live List i przesłanie adresów do ^{wewnętrznej} pamięci wspólnej /co zajmuje 213 μ sek ma on 250 μ sek. Wykorzystanie posiadanego czasu wynosi tu 85%. Zapas czasu jest więc stosunkowo nie-duży, a na jego istnieniu oparta jest zasada współpracy obu procesorów. Wynika stąd konieczność - w wypadku gdyby omawiane rozwiązanie kontrolera komunikacyjnego miało być realizowane - przetestowania go doświadczalnie i prawdopodobnie dodanie programowych lub sprzętowych mechanizmów zabezpieczających prawidłową współpracę procesorów. Wykonanie ich przekraczało ramy i możliwości mniejszej pracy. Gdyby w wyniku przeprowadzonych badań okazało się, że kontroler nie spełnia wymagań protokołu PROWAY, to jest jeszcze inna możliwość wydłużenia czasu na realizację programów. Można to osiągnąć przez wcześniejsze wysłanie bajtu synchronizacyjnego ramki nadawanej-przed zakończeniem przygotowania całej ramki. Daje dodatkowo 80 μ sek. Wymagałoby to wprowadzenia pewnych zmian w komunikacji kontroler - sterownik w stosunku do założeń podanych w rozdziale 2.

6. Podsumowanie

Tematem pracy było znalezienie odpowiedzi na pytanie czy można skonstruować kontroler komunikacyjny stacji systemu PROWAY w oparciu o dwuprocésorowy układ INTEL 8080 - INTEL 8035. Na podstawie przeprowadzonej analizy stwierdzono, że przy wykorzystaniu tego układu można zrealizować działania przewidziane w protokole PROWAY, przy szybkości transmisji 100 kB/s. Jednak, że wobec dużego obciążenia układu, teoretycznie znacznie przekraczającego 50% jego maksymalnych możliwości, do wyniku pracy należy podejść z pewną ostrożnością. Do otrzymania definitywnej odpowiedzi konieczne byłyby przede wszystkim próby ze sprzętem, których przeprowadzenie nie mieściło się w ramach niniejszej pracy. Należy także stwierdzić, że kontroler z omawianym układem byłby nie-rozwojowy - nie dawałby możliwości zwiększenia szybkości transmisji i istotnego skrócenia przerw w linii między transmisjami.

7. Literatura

1. "Analiza protokółów MIR-PROWAY pod kątem możliwości ich realizacji na dostępnej w kraju bazie elementowej" - sprawozdanie PIAP 1983r., nr rej.5057.
2. 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.

24
27