

OSRODEK BADAWCZO - ROZWOJOWY INFORMATYKI
ZAKŁAD METODYKI I EKONOMIKI SYSTEMÓW

729_B

Temat Nr 105/0000/2/T.1/77

TECHNIKA HIPO W DOKUMENTOWANIU SYSTEMU

Autor:

Kierownik Zakładu:

mgr T. Falborska

mgr inż. H. Zygiel

527

Warszawa, kwiecień 1977

3/160



Proj. 729
Egz. B

Spis treści

1. Wprowadzenie	3
2. Technika HIPO w poszczególnych etapach tworzenia systemu	4
2.1 Opracowanie wstępnego pakietu projektowego	4
2.1.1 Tworzenie diagramu VTOC	5
2.1.2 Tworzenie diagramów IPO	6
2.1.3 Weryfikacja pakietu	7
2.2 Opracowanie szczegółowego pakietu projektowego	8
2.2.1 Rozbudowywanie diagramu VTOC	8
2.2.2 Rozbudowywanie diagramów IPO	9
2.2.3 Weryfikacja pakietu	11
2.3 Tworzenie pakietu konserwacyjnego	12
2.4 Konserwacja	12
3. Technika sporządzania diagramu VTOC	14
4. Technika sporządzania diagramów IPO	18
4.1 Uwagi ogólne	18
4.2 Sekcje danych wejściowych i wyjściowych	21
4.3 Sekcja przetwarzania	24
4.4 Użycie strzałek	27
4.5 Sekcja opisu rozszerzającego	34
Bibliografia	35

1. WPROWADZENIE

Metodę dokumentowania systemu techniką HIPO można i należy traktować nie tylko jako sposób graficznego obrazowania poczyniń projektowych. Jest to metoda czynnie zaangażowana w proces tworzenia systemu informatycznego, przy czym nie ogranicza twórczej działalności analityków, projektantów, a także programistów. Przeciwnie, pozwala na "szeroki rozmach" nawet w graficznym obrazowaniu systemu. Należy więc ją rozpatrywać w dwóch aspektach. Z jednej strony jako narzędzie w tworzeniu logiczno-funkcyjnej struktury systemu /metodą hierarchicznego tworzenia systemu/, z drugiej zaś jako zbiór technik graficznego obrazowania wyników tych prac.

Rola i zadania metody HIPO w procesie tworzenia systemu zostały omówione w pracy pt: "HIPO jako technika dokumentowania hierarchicznego tworzenia systemu". Niniejsze opracowanie należy traktować jako jej kontynuację. Dlatego też wyjaśnienia wielu pojęć i zagadnień zostały pominięte.

W niniejszym opracowaniu zostanie omówiona metoda dokumentowania poszczególnych etapów tworzenia systemu, jak również jedna z konwencji używania symboli graficznych. Nie jest to jedyna możliwa konwencja i może nawet nie najlepsza. Wydaje się być jednak dosyć prosta, łatwa w użyciu i zapamiętaniu, a także nie stwarza kłopotów z powielaniem dokumentacji.

2. TECHNIKA HIPO W POSZCZEGÓLNYCH ETAPACH TWORZENIA SYSTEMU

W świetle metody HIPO prace nad tworzeniem systemu informatycznego prowadzone są w trzech etapach, a mianowicie:

- określanie /budowanie/ logiczno-funkcyjnej struktury systemu i wiązanie jej z danymi wejściowymi i wyjściowymi; wynikiem tych prac jest wstępny pakiet projektowy,
- uszczegółowienie struktury funkcyjnej i jej związków z danymi, przystosowanie do potrzeb technologii przetwarzania; wynikiem tych prac jest szczegółowy pakiet projektowy,
- oprogramowanie, przetestowanie, wdrożenie i konserwacja systemu, w wyniku czego powstaje pakiet konserwacyjny.

Dokumentacja systemu wykonanego metodą HIPO składa się z trzech części: wstępnego pakietu projektowego, szczegółowego pakietu projektowego i pakietu konserwacyjnego. Na ogół żaden z nich nie stanowi samodzielnej części dokumentacji, lecz uzupełniają się wzajemnie. Każdy jednak ma inne przeznaczenie i innego odbiorcę.

Główną część każdego pakietu projektowego stanowią diagram WFCO oraz diagramy IPO ogólne i szczegółowe /pozostałych elementów dokumentacji technika HIPO nie obejmuje/. Pomimo, iż w każdym pakiecie mają one tę samą formę, różnią się jednak poziomem szczegółowości zawartych informacji i stopniem rozbudowania szaty graficznej. Należy więc rozpatrywać rolę i zadania, a także budowę poszczególnych diagramów w zależności od pakietu, w którym występują.

2.1 Opracowanie wstępnego pakietu projektowego

Wstępny pakiet projektowy ma służyć przede wszystkim:

- użytkownikom systemu do zapoznania się z ogólnymi rozwiązaniami problemu,
- kierownictwu komórki informatycznej do określenia rozmiaru prac prowadzonych przez zespół,

- specjalistom z zakresu dziedzin objętych systemem do weryfikacji przedstawionych rozwiązań,
- projektantom i programistom do dalszych prac nad rozbudowywaniem i uszczegóławianiem projektu,
- osobom nowoprzyjętym w trakcie poszerzania zespołu projektowo-programowego do zapoznania się z projektowanym systemem.

Ponieważ z dokumentacji tej często korzystają nie-informatycy, winna być napisana językiem zrozumiałym /unikając należy "żargonu" informatycznego/, w sposób jasny, zwięzły i przejrzysty. Nie powinno używać się skrótów, a w przypadkach koniecznych wyjaśniać je w miejscu pierwszego użycia lub w specjalnej tablicy skrótów. .

2.1.1 Tworzenie diagramu VTOC

W tym pakiecie diagram VTOC obrazuje logiczną strukturę funkcji systemu, które są z reguły funkcjami złożonymi. Jego najwyższy "prostokąt" /zob.rys.1/ określa funkcję ogólną, natomiast następny poziom rozbija ją na logiczne podfunkcje /funkcje główne/. Nie należy w tym czasie zastanawiać się nad "pogrupowaniem" tych funkcji dla celów przetwarzania, lecz nad ich związkami czysto logicznymi. Każdy kolejny niższy poziom rozbija funkcję wyższego poziomu na jej logiczne podfunkcje. Najczęściej na tym etapie wystarcza 4 - 5 poziomów. Generalną jednak zasadą jest kontynuowanie rozwijania funkcji ogólnej do momentu, gdy stanie się ona zrozumiała zarówno dla projektantów jak i użytkowników. Pierwszy narysowany diagram VTOC najczęściej nie jest wersją ostateczną. Jego budowa zależy przecież od "biegu myśli" projektanta, a w konsekwencji również od rozwiązań przyjętych i przedstawionych w diagramach IPO. Tak więc pierwsze "przymiarki" do rozwiązania zagadnienia podlegają wielu modyfikacjom, by w fazie końcowej dać strukturę logiczną proponowanego systemu. Każdy "prostokąt" w diagramie VTOC przypisuje sobie diagram IPO, którego tytuł i numer identyfikacyjny jest taki sam, jak tytuł i numer umieszczony

w tym "prostokącie".

2.1.2 Tworzenie diagramów IPO

Jednocześnie z tworzeniem diagramu VTOC powstają /najczęściej/ również diagramy IPO, które obrazują powiązania poszczególnych funkcji z odpowiednimi danymi wejściowymi i wyjściowymi. Funkcje w sekcji przetwarzania /zob.rys.5a/ diagramu IPO zapisane są jako ciąg ponumerowanych kroków, przy czym kolejność ta nie wyznacza dokładnej kolejności ich realizacji.

Dla potrzeb wstępnego pakietu projektowego najczęściej sporządza się jeden diagram ogólny IPO opisujący ogólną funkcję systemu /związany z "najwyższym prostokątem" diagramu VTOC/. W jego sekcji przetwarzania umieszczone są główne funkcje systemu /ale nie tylko - np: funkcje sortowania, łączenia zbiorów itp./. Dla pozostałych funkcji wykonuje się diagramy szczegółowe.

Dane wejściowe i wyjściowe można i należy określać ogólnie, nie podając nośników, na jakich będą się w przyszłości znajdowały. Niemniej istotną sprawą jest, aby dane wejściowe zaspakajały potrzeby wszystkich funkcji, jak również, aby określone funkcje gwarantowały otrzymanie wszystkich danych wyjściowych. Należy również zwrócić uwagę, by dane wejściowe/wyjściowe określone w diagramie niższego poziomu były zawarte w danych wejściowych/wyjściowych odpowiedniego diagramu poziomu wyższego.

Analogiczny związek w logicznych relacjach funkcji gwarantuje metoda hierarchicznego tworzenia systemu zdefiniowanego jego funkcjami /wykorzystywana w trakcie budowania diagramu VTOC/.

Uwaga

Dla ułatwienia i usprawnienia pracy projektowej wskazane jest zachowanie następującej kolejności wykonywania diagramu IPO.

1. Wypisanie /w sekcji wyjścia/ wszystkich wyjść, które można jednoznacznie i dobrze określić.
2. Równoczesne rozwijanie sekcji wejścia i przetwarzania wraz z uzupełnieniem sekcji wyjścia wyjściami pośrednimi.

3. Połączenie funkcji z odpowiednimi danymi wejściowymi i wyjściowymi /nie należy zbyt martwić się powstałym "bałaganem";. wszak to tylko diagram roboczy/.
4. Uporządkowanie i "kosmetyka" diagramu /poprawienie kolejności oraz odpowiednie pogrupowanie danych i funkcji; można oczywiście "prostokąty" zamykać w "prostokąty" większe, ale wykonanie tego jest trudniejsze graficznie/.

Po tych zabiegach powinien powstać przejrzysty i czytelny diagram IPO.

Wskazane jest również opracowanie każdego diagramu IPO tak, aby mieścił się /bez opisu rozszerzającego/ na jednej stronie. Czasami jest to sprawa dosyć trudna, niemniej ze względu na podniesienie czytelności warto do tego dążyć.

2.1.3 Weryfikacja pakietu

W trakcie hierarchicznego tworzenia systemu i wykonywania dokumentacji techniką HIPO weryfikacja pakietu następuje niejako automatycznie-poprzez sam proces projektowania. Niemniej jednak wskazane jest wykonanie pełnego "przeglądu" pakietu przez połączoną grupę informatyków /projektantów, analityków/, użytkowników, a także specjalistów z danej dziedziny. Każdy z nich ma inne spojrzenie na projektowane zagadnienie i tym cenniejszy jest ich udział w pracach weryfikacyjnych. Do tego celu nie musi służyć ostateczna postać dokumentacji. Musi być tylko czytelna i zrozumiała. Pierwszy pełny "przegląd" powinien być robiony ze względu na:

- strukturę logiczną i zawartość dokumentacji
/czy diagram VTOC jest strukturą hierarchiczną systemu?
czy diagram najwyższego poziomu jest dokładnie opisany przez odpowiadającą mu grupę diagramów niższego poziomu?
czy dane wejściowe/wyjściowe w diagramach każdego poziomu są podzbiorami danych wejściowych/wyjściowych w odpowiednich diagramach wyższego poziomu?/,
- dokładność technologiczną
/czy, zrozumiale i poprawnie określone jest każde wyjście z danego kroku?
czy poprawnie opisana jest każda funkcja?

czy nie ma funkcji nieokreślonych?
czy brakuje danych wejściowych lub wyjściowych?/,

- wymogi formalne

/czy poprawne są symbole, strzałki i techniki graficzne?
czy zrozumiała i konsekwentna jest terminologia?
czy zdefiniowane są użyte symbole?/.

Przeglądy takie należy powtarzać aż do otrzymania poprawnej, zwartej, przejrzystej i czytelnej dokumentacji. Wstępny pakiet projektowy zatwierdzony przez przyszłego użytkownika projektowanego systemu jest materiałem wyjściowym do dalszych prac nad rozwijaniem systemu.

2.2 Opracowanie szczegółowego pakietu projektowego

Główne zadania tego etapu prac to:

- rozwijanie i grupowanie funkcji do poziomu modułu/programowego/ - rozbudowywanie diagramu VTOC,
- przystosowanie struktury funkcyjnej systemu do potrzeb technologii przetwarzania,
- określenie elementów danych wejściowych i wyjściowych dla poszczególnych modułów - rozbudowywanie diagramów IPO.

Szczegółowy pakiet projektowy jest udokumentowaniem prac tego etapu i służy przede wszystkim informatykom. Niemniej jednak w trakcie jego wykonywania pamiętać należy, że nie tylko autorzy będą z niego korzystać. Niezależnie od zachowania użytej uprzednio we wstępnym pakiecie projektowym terminologii i symboliki, winna być powtórnie opisana i wyjaśniona; często lista używanych symboli jest z konieczności rozbudowywana i dokumentowanie tego wymaga szczególnej uwagi.

2.2.1 Rozbudowywanie diagramu VTOC

Rola diagramu VTOC w szczegółowym pakiecie projektowym jest taka sama jak we wstępnym pakiecie projektowym. Winien on spełniać zadania niejako przewodnika po pakiecie oraz przedstawiać rozbudowaną i pełną strukturę hierarchiczną funkcji systemu. Zatem istnieje potrzeba zachowania jego logiczno-funkcyjnej budowy.

Jeśli główne funkcje systemu zostały dobrze określone i jeśli są w pełni zrozumiałe, uszczegółowienie diagramu VTOC polega na przybliżeniu i powiązaniu funkcji logicznych z potrzebami technologicznymi przetwarzania. Raz zdefiniowane funkcje mogą być zarówno rozdzielone, jak i grupowane tak, aby w efekcie końcowym można było wydzielić w miarę optymalny układ programów i modułów programowych nadających się do bezpośredniego kodowania. Praca ta należy prowadzić do momentu, gdy moduły programowe dadzą się zakodować w około 50 instrukcjach /dla języka wyższego rzędu/. Wszystkie użyte w projektowanym systemie funkcje standardowe oraz procedury usługowe winny być umieszczone w diagramie VTOC w "prostokacie" odpowiedniej postaci /zob.rys.4/. W przypadku wystąpienia kilku funkcji w kilku miejscach w diagramie, zaleca się wydzielenie ich jako odrębnej funkcji i opisanie w jednym miejscu. Takie postępowanie jest wygodne również ze względu na zmniejszenie "zatkoczenia" w diagramie VTOC. Jeśli diagram VTOC nie mieści się na jednej stronie /co zdarza się najczęściej/, można go rozrysowywać na kilku stronach, pamiętając jednak, aby zachować jego czytelność umieszczając odpowiednie wyłączenia i odwołania na pierwszej stronie diagramu /zob.rys.1/.

Rozwinięcia funkcji głównej 3 należy szukać w Diagramie VTOC część 2, natomiast rozwinięcia funkcji głównej 4 w Diagramie VTOC część 3.

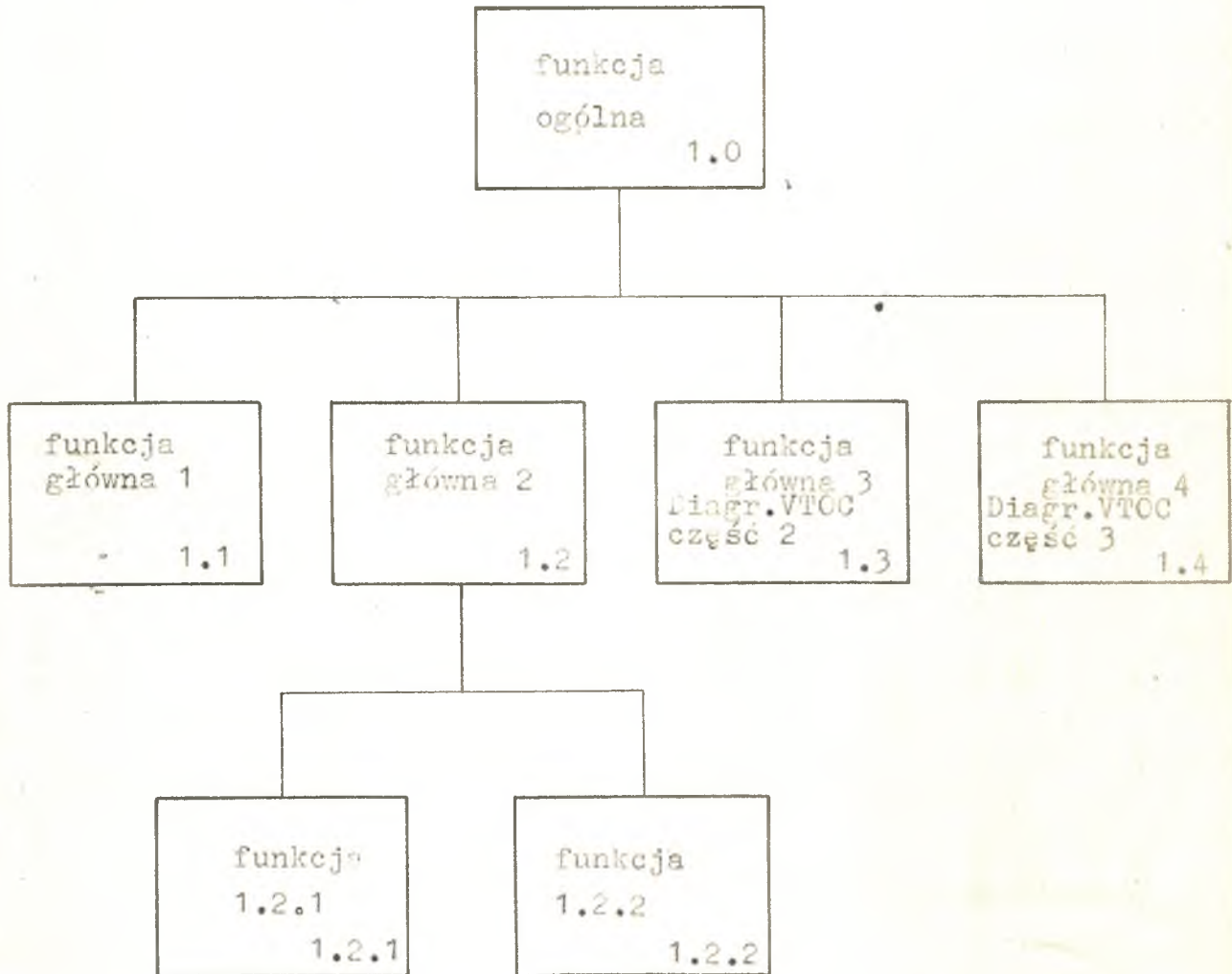
2.2.2 Rozbudowywanie diagramów IPO

W czasie uszczegóławiania projektu diagramy IPO, zarówno ogólne jak i szczegółowe, budowane są w taki sposób, aby przedstawiały odpowiednią kolejność wykonania makroinstrukcji^{1/} i operacji^{2/}. Diagramy ogólne, które

1/ makroinstrukcja - pojedynczy instruktor wykonujący funkcję najmniejszą pewną

2/ operacja - wyrażenie strukturalne wyrażające funkcję realizowaną przez jedną lub więcej instrukcji, które mogą być wywołane z poziomu

Diagram VTCC
część 1



Rys.1 Sposób podziału diagramu VTCC

najczęściej w tym miejscu opisują logikę działania programów winny być skonstruowane tak, aby programiści /w przyszłości operatorzy systemu/ wiedzieli, w jaki sposób wiązać poszczególne moduły programowe w logiczną całość. Dla tego celu w kolumnach "Procedury", "Etykiety" i "Odwołania" /zob.rys.5b/ umieszcza się odpowiednie informacje /użyte skróty i symbole winny być uprzednio wyjaśnione/. Nazwy procedur, etykiet, a także symbole wszystkich diagramów powinny być zachowane w całej dokumentacji. Diagramy szczegółowe, opisujące funkcje poszczególnych modułów programowych, należy tak budować, by na ich podstawie można było bezpośrednio zakodować te moduły. Wskazane jest również ustalenie nazw poszczególnych elementów danych, używanych tablic, przełączników, pól roboczych itp.

Ponieważ diagramy IPO powinny w zasadzie mieścić się na jednej stronie, opisane w nich operacje często muszą być formułowane zbyt zwięźle, aby można je było dokładnie zrozumieć. Miejscem szczegółowego wyjaśnienia jest opis rozszerzający /zawierający dowolną ilość stron/, który ma za zadanie:

- dostarczenie dodatkowych informacji o operacjach i makroinstrukcjach,
- dostarczenie opisu stosowanych algorytmów obliczeniowych,
- ukazanie powiązań danego diagramu z innymi elementami dokumentacji.

Uwaga

Jeśli w trakcie uszczegóławiania funkcji systemu część diagramów IPO /najczęściej opisujących funkcje wysokich poziomów/ nie ulega modyfikacji, n_i_e_n_a_l_e_ż_y ich umieszczać po raz drugi w szczegółowym pakiecie projektowym.

2.2.3 Weryfikacja pakietu

"Przeglądy" szczegółowego pakietu projektowego są kontynuacją "przeglądów" wykonywanych na wstępnym pakiecie projektowym. Nie muszą w nich jednak uczestniczyć specjaliści nie-informatycy i użytkownicy /choć uczestnictwo

użytkowników jest niekiedy wskazane/, ponieważ te "prze-glądy" częściej dotyczą spraw związanych ściśle z technologią przetwarzania. Głównie chodzi o sprawdzenie poprawności uszczegółowienia wszystkich funkcji i odpowiedniego ich przystosowania do potrzeb przetwarzania. Poza tym należy tu zwrócić szczególną uwagę na:

- konsekwencję w używaniu nazw programów, zbiorów, danych i ich elementów, tablic, itp,
- powiązanie elementów danych wejściowych i wyjściowych z operacjami,
- wielkości poszczególnych modułów programowych i łatwość ich kodowania,
- odpowiednie wyjaśnienie wszystkich użytych nazw.

2.3 Tworzenie pakietu konserwacyjnego

Pakiet konserwacyjny, pomimo iż jest elementem dokumentacji wykonanej metodą HIPO, w przeważającej części zawiera elementy z techniką HIPO nie związane. Jest on wynikiem prac prowadzonych w fazach programowania, testowania i wdrażania systemu. Główną więc jego część stanowią wydruki zakodowanych modułów i programów, dane i wyniki testowania, instrukcje operatorskie, instrukcje dla użytkownika itp. Elementy dokumentacji wykonane techniką HIPO pojawiają się tylko w przypadku wprowadzania modyfikacji do zaprojektowanego systemu.

2.4 Konserwacja

W czasie użytkowej eksploatacji systemu jego konserwacją najczęściej zajmują się programiści, nie-autorzy systemu. Uzyskanie konsultacji i wyjaśnień od autorów jest często zupełnie niemożliwe. Grupy konserwujące system, aby wnieść jakiegokolwiek poprawki, muszą niemalże "nauczyć się" całego systemu, co dla rozbudowanych zastosowań jest rzeczą trudną i czasochłonną. Wynika stąd potrzeba szybkiego zlokalizowania ujawnionego błędu i poprawienia go bez konieczności znania całego systemu w szczególności i bez zachwiania jego struktury. Taką możliwość daje dokumentacja wykonana techniką HIPO - szczegółowy pakiet projektowy. Już na podstawie diagramu

VTOC można określić drogę, jaką należy dokładnie prześledzić, celem zlokalizowania błędu. Diagram ten bowiem pokazuje logiczne związki a także fizyczną zależność między programami i modułami programowymi^{1/}. Programy zaś swą strukturę logiczną mają opisaną w odpowiednich diagramach IPO. Ze względu na powiązania między poszczególnymi diagramami IPO, zlokalizowanie błędu jest rzeczą prostą, natomiast poprawienie go wymaga zapoznania się zwykle z budową jednego modułu. Poprawki wprowadzane do modułu /poza błędami kodowania/ najczęściej sprowadzają się do poprawienia opisu rozszerzającego, a to już nie nastrocza żadnych trudności. Na marginesie warto zauważyć, że wniesienie poprawki do modułu, który występuje w wielu programach i umieszczony jest w wielu miejscach diagramu VTOC, nie wymaga zmian w żadnym programie. Poprawia się tylko umieszczony w jednym miejscu w pakiecie diagram IPO dla tego modułu /no i oczywiście jego postać zakodowaną/.

Podobnie łatwo i prosto wprowadza się modyfikacje polegające na dodaniu nowych funkcji do systemu. Należy wtedy "podwiesić" dodatkową gałąź funkcyjną na diagramie VTOC, rozbudować ją, pokazać powiązania z już opisanymi funkcjami, a także dobudować odpowiednie diagramy IPO.

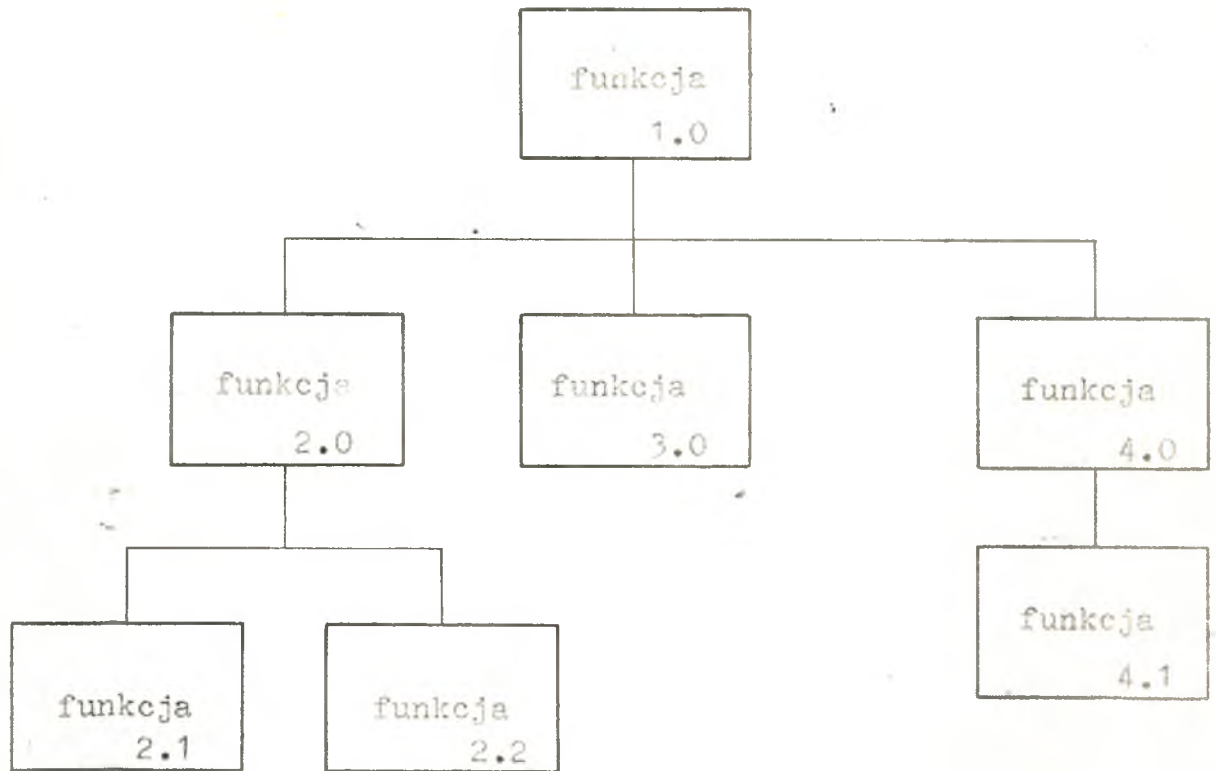
Nieco więcej trudności bywa w przypadku usuwania pewnych modułów lub grupowania modułów zbyt małych. Należy wtedy dokładnie sprawdzić we wszystkich miejscach systemu, w których one występują /odnalezienie poprzez diagram VTOC/, czy taka operacja jest możliwa.

1/ Poziom modułu, programu, systemu wyjaśniono w opracowaniu [1]

3. TECHNIKA SPORZĄDZANIA DIAGRAMU VTOC

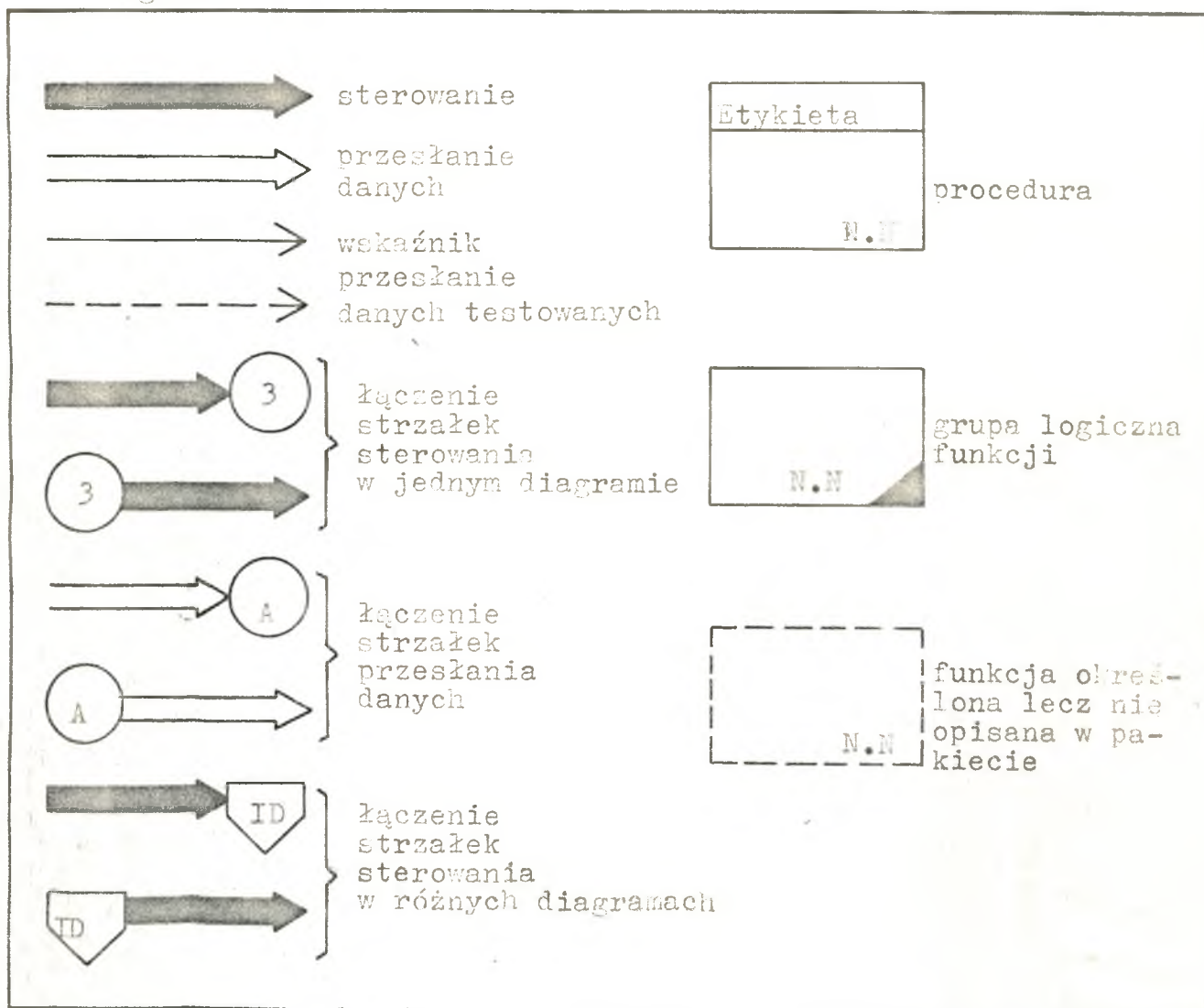
W punktach 3 i 4 przedstawiona jest jedna z częściej używanych konwencji stosowania symboli graficznych w dokumentacji systemu informatycznego wykonanej techniką HIPO. Odnosnie diagramu VTOC wyróżnić można następujące problemy:

- Diagram VTOC powinien być pierwszą pozycją w każdym pakiecie projektowym.
- Każdy "prostokąt" w diagramie VTOC powinien zawierać swój numer, który jest zarazem identyfikacyjny dla diagramu IPO opisującego daną funkcję; dodatkowo może zawierać także numer strony, na której znajduje się ten diagram.
- Funkcje i diagramy IPO należy numerować jak na rys. 2; takie numerowanie ułatwia dodawanie nowych funkcji bez potrzeby zmiany numeracji funkcji już istniejących.
- Używając metody numeracji dziesiętnej, inne elementy dokumentacji /np: projekty zapisów, projekty tabulogramów, tablice decyzyjne itp./ mogą być numerowane poprzez dodanie odpowiedniej litery do numeru diagramu IPO, który się do nich odwołuje, jak np: 4.2.1A, 4.1C.
- Każdy diagram VTOC powinien zawierać legendę /zob.rys.3/ wyjaśniającą użyte symbole. Jeśli pakiet HIPO ma być czytany przez ludzi zupełnie nie znających techniki HIPO lub/i problemów przetwarzania danych, do diagramu VTOC winna być dołączona obszerna legenda wyjaśniająca:
 - symbole graficzne używane we wszystkich sekcjach diagramów IPO,
 - używane skróty, których informatykom nie trzeba wyjaśniać, szczególnie używane w sekcji opisu rozszerzającego.
- Funkcje opisane w pakiecie umieszczają się w prostokątach narysowanych linią ciągłą /zob.rys.4a/.
- Dla przedstawienia funkcji określonej, lecz której opis nie został zawarty w pakiecie, używa się prostokąta narysowanego linią przerywaną; tak samo oznaczać można również funkcję, która jest spełniana przez inny system /zob.rys.4b/.



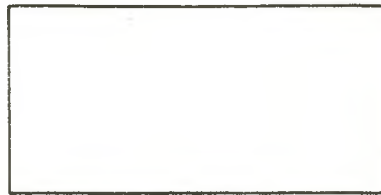
Rys. 2 Sposób numeracji funkcji w diagramie VTOC

Legenda

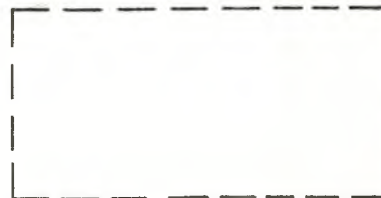


Rys. 3. Legenda

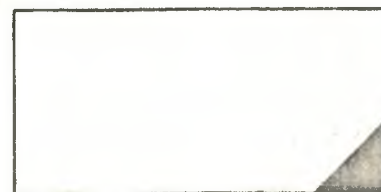
- Funkcje będące grupą logiczną kilku funkcji umieszcza się w prostokątach z zaciemnionym prawym dolnym rogiem /zob.rys.4c/.



a/ Symbol funkcji opisanej w pakiecie



b/ Symbol funkcji zdefiniowanej, lecz nie opisanej w pakiecie



c/ Symbol grupy logicznej funkcji

Rys. 4 Użycie symboli graficznych w diagramie VTOC

4. TECHNIKA SPORZĄDZANIA DIAGRAMÓW IPO

4.1 Uwagi ogólne

- . Każda sekcja w diagramie IPO /wejście, przetwarzanie, wyjście/ powinna być ujęta w ramki i opisana nad górną jej linią.
- . Rozmiary poszczególnych sekcji zależne są od ilości znaków graficznych i tekstu w nich zawartego; mogą być różnicowane w miarę potrzeb.
- . Wskazane jest ograniczenie rozmiarów diagramu IPO do jednej strony. Jeśli jednak używa się diagramów wielostronnicowych, operacje winny być numerowane kolejno przez cały diagram /co już zostało wspomniane, jednak w związku z istotą problemu celowym wydaje się zwrócić nań szczególnej uwagi/.
- . Każdy diagram IPO winien zawierać pewne informacje identyfikacyjne, między innymi:
 - nazwisko autora diagramu,
 - nazwę systemu/programu, którego dotyczy,
 - nr strony /w przypadku diagramu wielostronnicowego należy wyraźnie zaznaczyć, że jest to n-ta strona diagramu m-stronnicowego/,
 - nr identyfikacyjny diagramu /jak w diagramie VTOC/,
 - skróconą nazwę funkcji, którą opisuje /jeśli jest to diagram opisujący program, należy użyć w tym miejscu symbolu programu/,
 - opis funkcji /musi to być ten sam opis, który znajduje się w diagramie VTOC/,

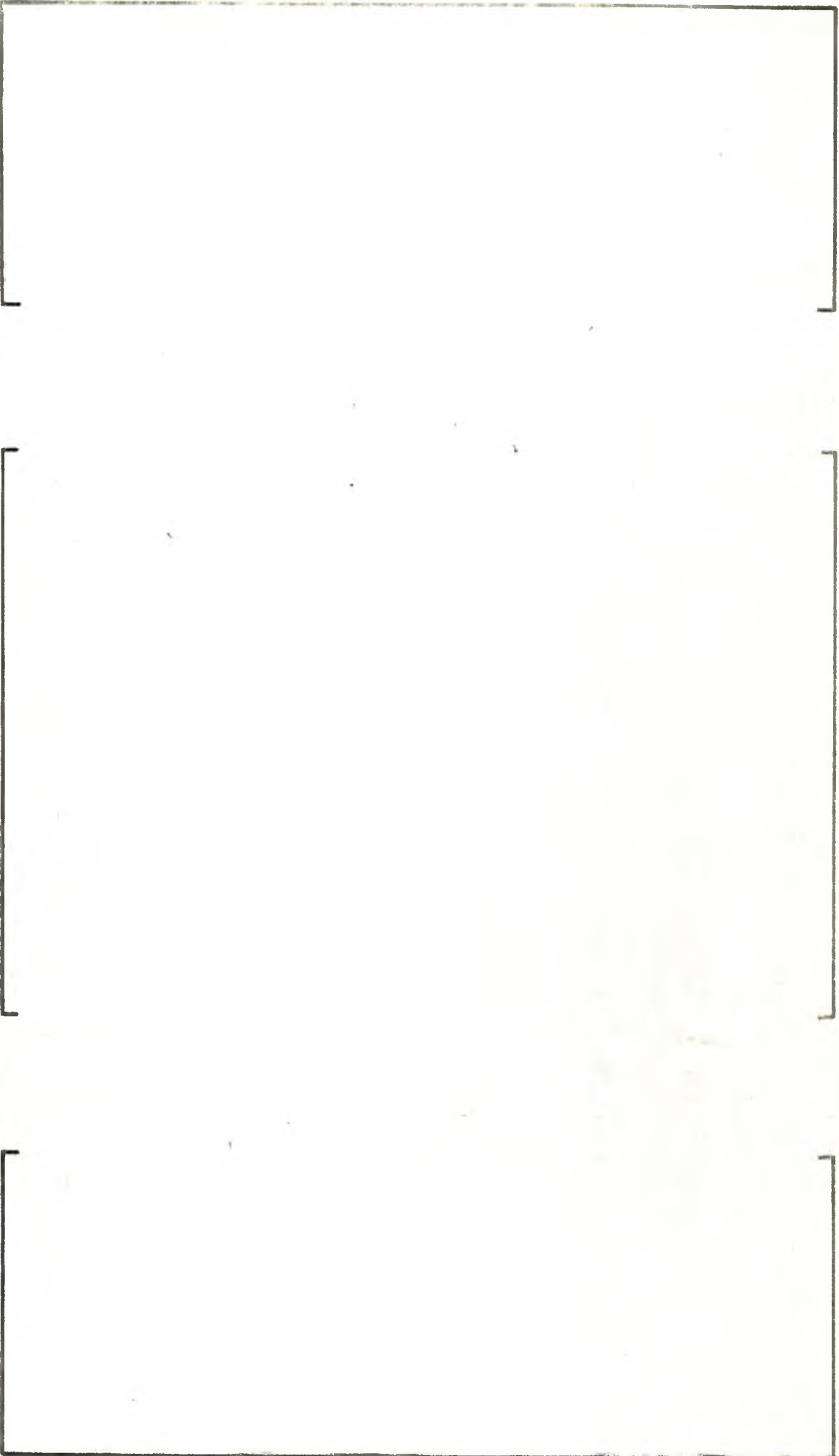
Przykład formularza, na którym można budować diagram IPO pokazano na rys. 5a i 5b.

Autor: _____ System/Program: _____ Data: _____ Strona: _____
ID diagramu: _____ Nazwa: _____ Opis: _____

Ważenie

Przetwarzanie

Ważenie



Rys. 5a Formularz do rysowania diagramu IPO

Opis rozszerzający

Uwagi	Proced.	Etykiet	Odwoł.

Opis rozszerzający

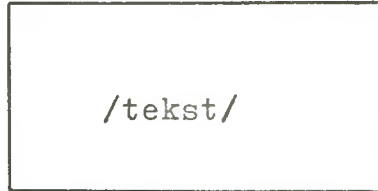
Uwagi	Proced.	Etykiet	Odwoł.

Rys. 5b Formularz opisu rozszerzającego

4.2 Sekcje danych wejściowych i wyjściowych

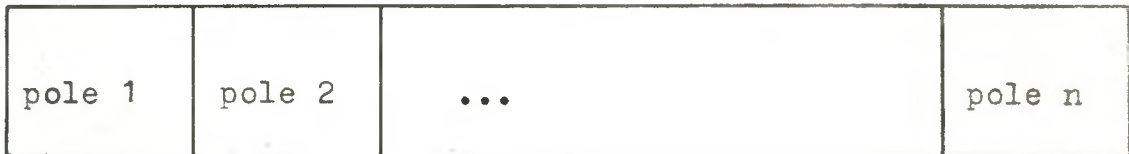
- Zasady budowy sekcji wejścia i sekcji wyjścia są takie same.
- Symbolem pojedynczego elementu danych /pola, rejestru, zapisu, obszaru pamięci operacyjnej, itp./ jest prostokąt opatrzony najczęściej etykietą /zob.rys.6a/. Wielkość tego prostokąta nie ma znaczenia funkcyjnego. Zależy tylko od ilości informacji, które będą weń wpisane. Etykieta może być:
 - nazwą zbioru, w którym ta dana występuje /wtedy tekst wewnątrz prostokąta powinien identyfikować daną/,
 - nazwą pola lub rejestru,
 - nazwą obszaru pamięci operacyjnej /wtedy tekst może być nazwą odpowiedniej komórki/.Tekst wewnątrz prostokąta może słownie opisywać element danych.
- Jeśli diagram dotyczy programu/modułu programowego zarówno etykieta jak i tekst winny być nazwami używanymi w programie/module programowym.
- Sposób przedstawiania zapisu wraz z planem pól wewnątrz obrazuje rys. 6b. Ze względu na rysowanie strzałek wygodniejsza jest jednak postać pionowa. Ilość pozostawionego wolnego miejsca między polami nie jest zależna od ilości pól pominiętych. Także kolejność wypisania nazw pól nie musi być zgodna z aktualną ich kolejnością w zapisie, tablicy itp. /jeśli jest to opis programu, należy to wyjaśnić w opisie rozszerzającym/. Taka możliwość manewrowania kolejnością pól jest wygodna ze względu na przedstawianie przesłania wartości tych pól i rysowanie strzałek.
- Dla zobrazowania zewnętrznych nośników danych, takich jak karty dziurkowane, taśmy, dyski, tabulogramy, itp. używa się powszechnie znanych i stosowanych symboli graficznych. Jeśli natomiast określenie nośnika, nie jest potrzebne lub nie jest możliwe, można użyć symbolu elementu danych.

Etykieta



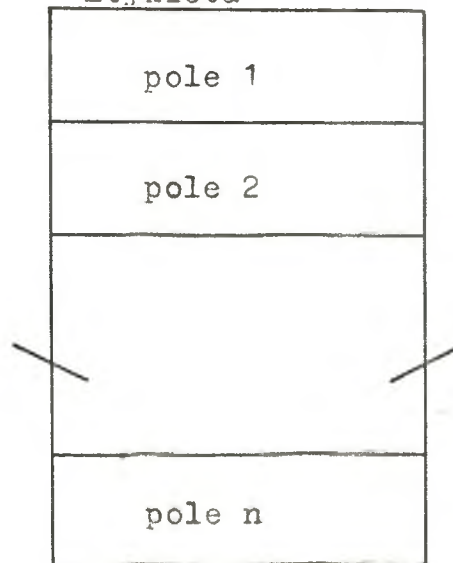
a/ Sposób przedstawienia pojedynczego elementu danych

Etykieta



lub

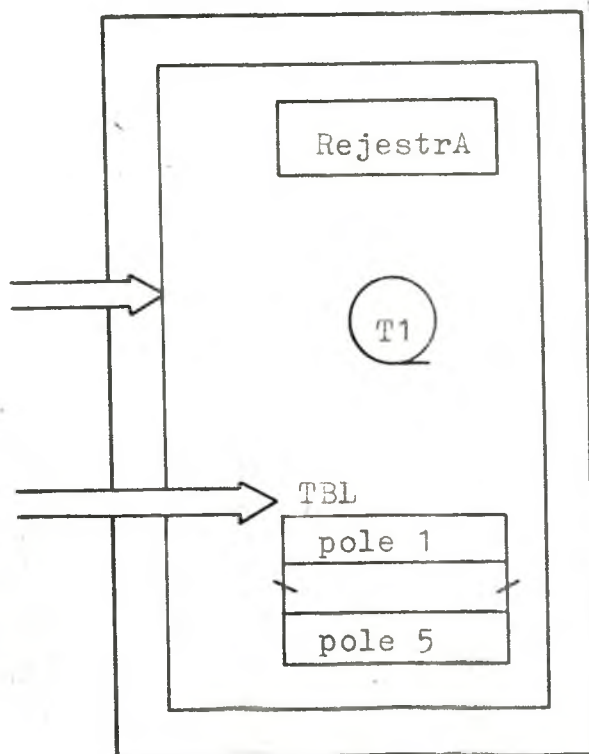
Etykieta



b/ Sposób przedstawienia zapisów lub obszarów
pamięci operacyjnej

Rys. 6 Sposoby przedstawiania danych

- Dla uproszczenia strony graficznej i redukcji ilości strzałek wprowadza się grupowanie danych. Grupa danych to kilka elementów danych obwiedzionych "dużym" prostokątem. Strzałki dochodzące do tego prostokąta pokazują, że wszystkie elementy danych znajdujące się wewnątrz biorą udział w przepływie danych /zob.rys.7/.



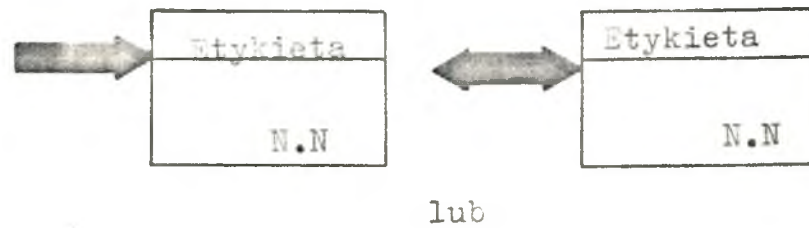
Rys. 7 Sposób oznaczania grupy danych

Uwaga

W sekcji wejścia /wyjścia/ winny być umieszczone tylko te elementy danych, które są wykorzystywane /tworzone lub modyfikowane/.

4.3 Sekcja przetwarzania

- . Sekcja ta jest kolumną operacji ponumerowanych poczynając od 1 jako numeru pierwszej operacji na pierwszej stronie diagramu IPO.
- . Operacje winny być opisane krótko i zrozumiale, a także reprezentować raczej funkcje aniżeli instrukcje kodowania. Dlatego wskazane jest rozpoczęcie określenia ich /szczególnie na poziomie programu czy modułu programowego/ czasownikiem w trybie rozkazującym, jak np: czytaj, sprawdź, wylicz, wyznacz, utwórz, prześlij, zapisz, itp.
- . Nie określa się maksymalnej ilości operacji w jednym diagramie; niemniej ilość przekraczająca 7 - 10 operacji utrudnia szybkie zrozumienie opisanej funkcji. Nie wskazane jest również umieszczanie zbyt wielu szczegółów w sekcji przetwarzania. Problemy te można rozwiązać przenosząc część informacji do opisu rozszerzającego lub podzielenie funkcji na diagramy niższego poziomu.
- . W sekcji przetwarzania, ani w ogóle w diagramach IPO nie powinny być zawarte symbole decyzji.
- . Sytuację, że dana operacja jest opisana szczegółowo w innym diagramie IPO można zobrazować dwoma alternatywnymi sposobami:
 - umieszczając numer identyfikacyjny odpowiedniego diagramu niższego poziomu w kolumnie odwołań przy numerze tej operacji w sekcji opisu rozszerzającego,
 - rysując prostokąt wokół opisu tej operacji w sekcji przetwarzania i umieszczając numer odpowiedniego diagramu niższego poziomu w prawym dolnym rogu tego prostokąta.
- . Operacje umieszczone w sekcji przetwarzania mogą korzystać z procedur. Sposób przedstawienia procedury pokazuje rys. 8, gdzie Etykieta to nazwa procedury, N.N to numer identyfikacyjny odpowiadającego jej diagramu IPO, a strzałki pokazują skąd i dokąd przekazywane jest sterowanie.

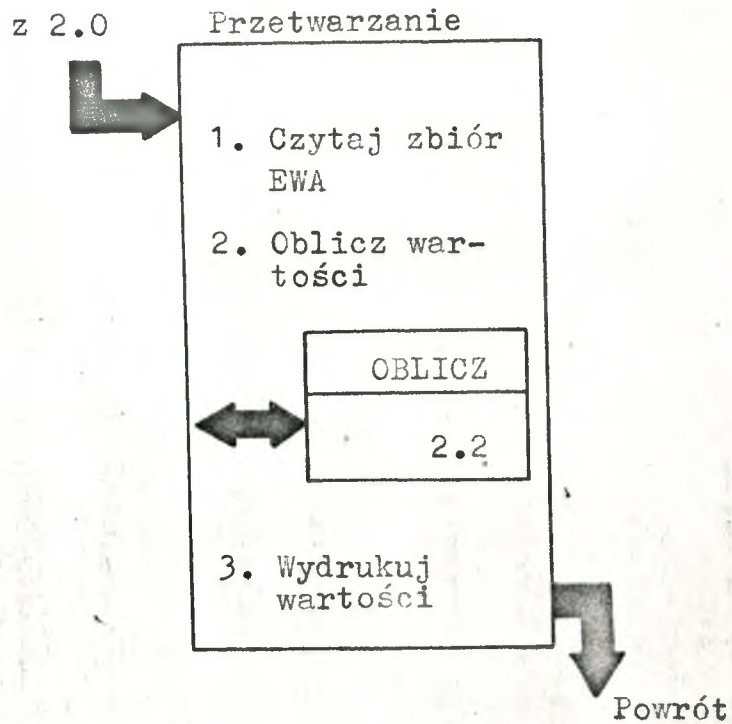


Rys. 8 Sposób przedstawienia bloku procedury

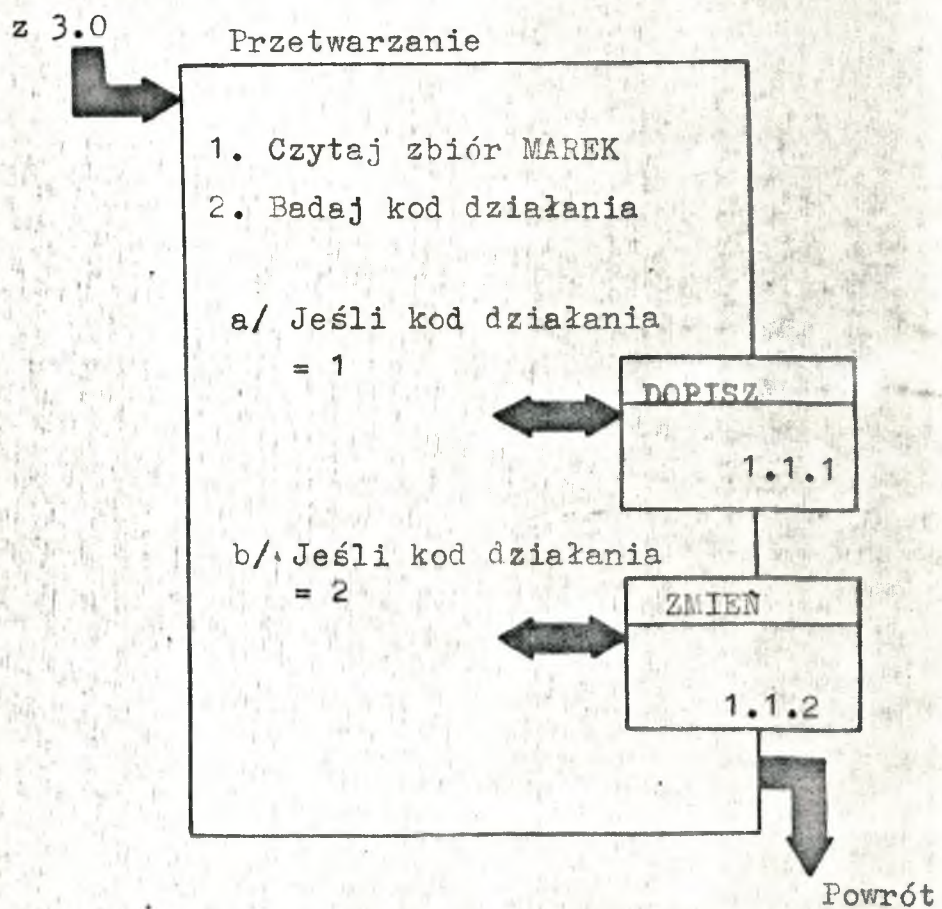
- Użycie procedur wewnętrznych i zewnętrznych należy rozróżnić i odpowiednio przedstawić graficznie. Sposób przedstawienia procedury wewnętrznej pokazany jest na rys. 9, a procedury zewnętrznej na rys. 10.
- Funkcje iteracyjne można przedstawiać "zamykając" je w prostokąt /zob.rys.11/.

Uwaga

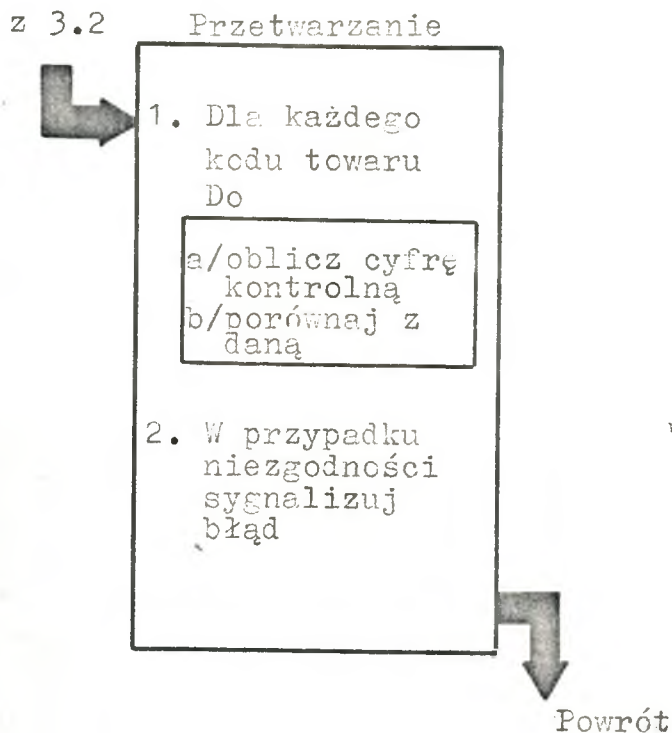
Dla podniesienia czytelności diagramu wskazane jest ograniczenie ilości poziomów "zagnieżdżenia" /2 - 3/. Jeśli w diagramie występuje ich więcej, korzystniejsze jest "rozbicie" diagramu do poziomu niższego lub umieszczenie dodatkowych informacji w sekcji opisu rozszerzającego.



Rys. 9 Sposób przedstawienia procedury wewnętrznej



Rys. 10 Sposób przedstawienia procedury zewnętrznej



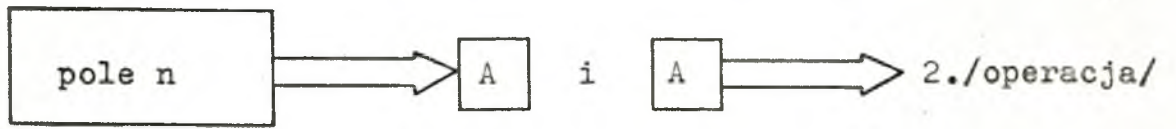
Rys. 11 Sposób przedstawienia funkcji iteracyjnej

4.4. Użycie strzałek

- Szerokość używanych strzałek ogólnie nie ma znaczenia funkcyjnego. Ponieważ jednak szablon do wykonywania dokumentacji techniką HIPO posiada strzałki dwóch szerokości, przyjęte jest stosowanie:
 - szerokiej strzałki, gdy jest ona jedna od sekcji wejścia lub do sekcji wyjścia,
 - wąskiej strzałki we wszystkich pozostałych sytuacjach.Każde szczególne użycie strzałek innej szerokości należy wyjaśnić w legendzie.
- Strzałka przesłania danych /zob.rys.12a/ wskazuje, jakie elementy danych lub grupy elementów danych są przesyłane, przetwarzane, tworzone lub modyfikowane przez odpowiednią operację. W części wejścia powinna zaczynać się od prawej linii prostokąta elementu danych lub grupy danych i kończyć przy numerze operacji, która z tych danych korzysta.

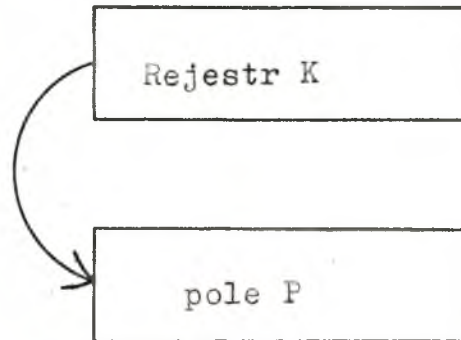


a/ Strzałki przesłania danych



b/ Sposób łączenia przesłania danych

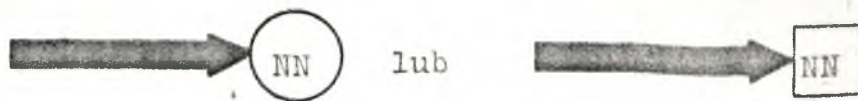
/litera A może być również otoczona kołem/



c/ Strzałka wskaźnika danych



d/ Strzałka sterowania



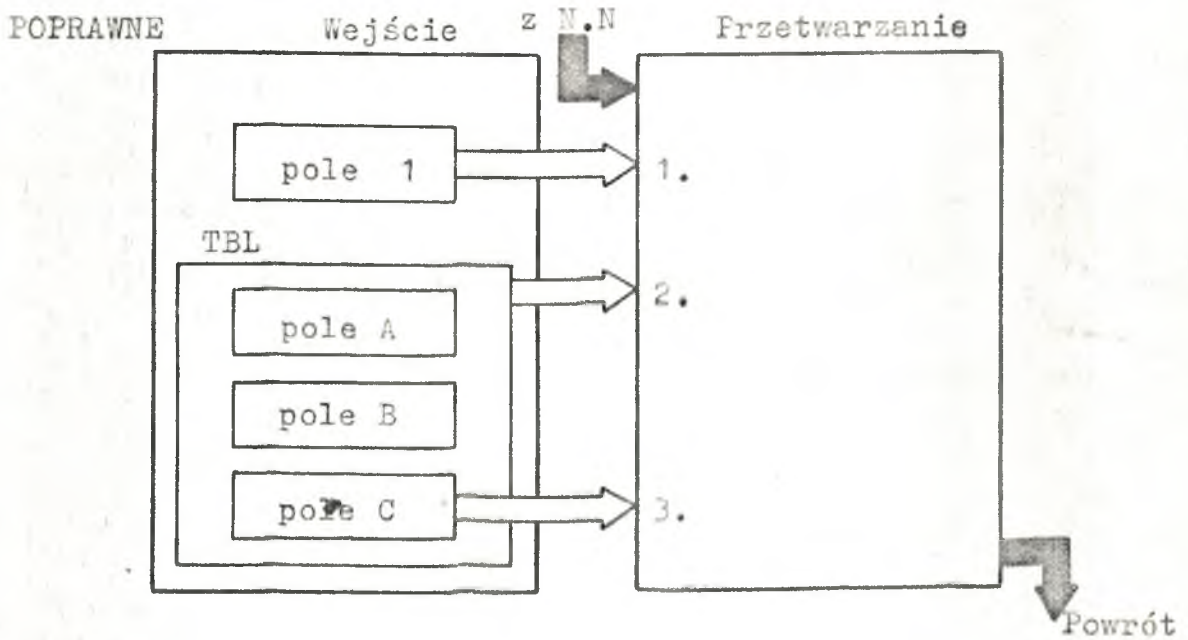
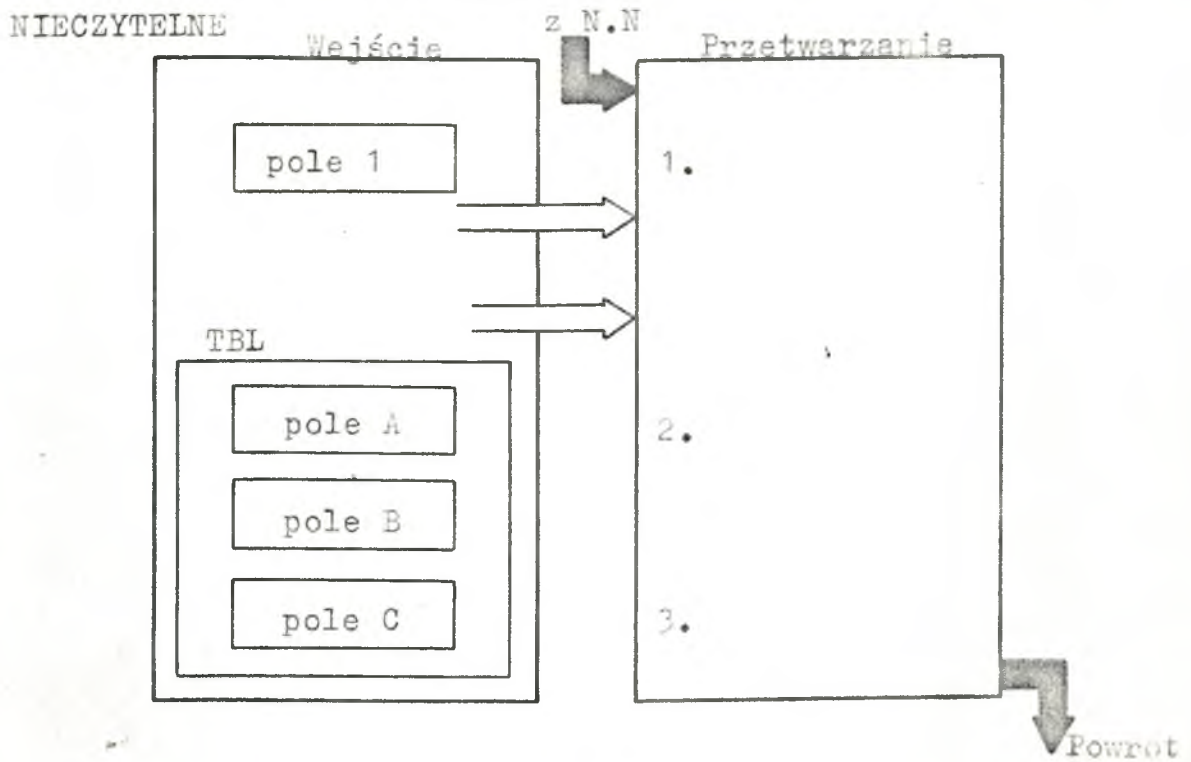
e/ Przekazanie sterowania w jednym diagramie



f/ Przekazanie sterowania do innego diagramu

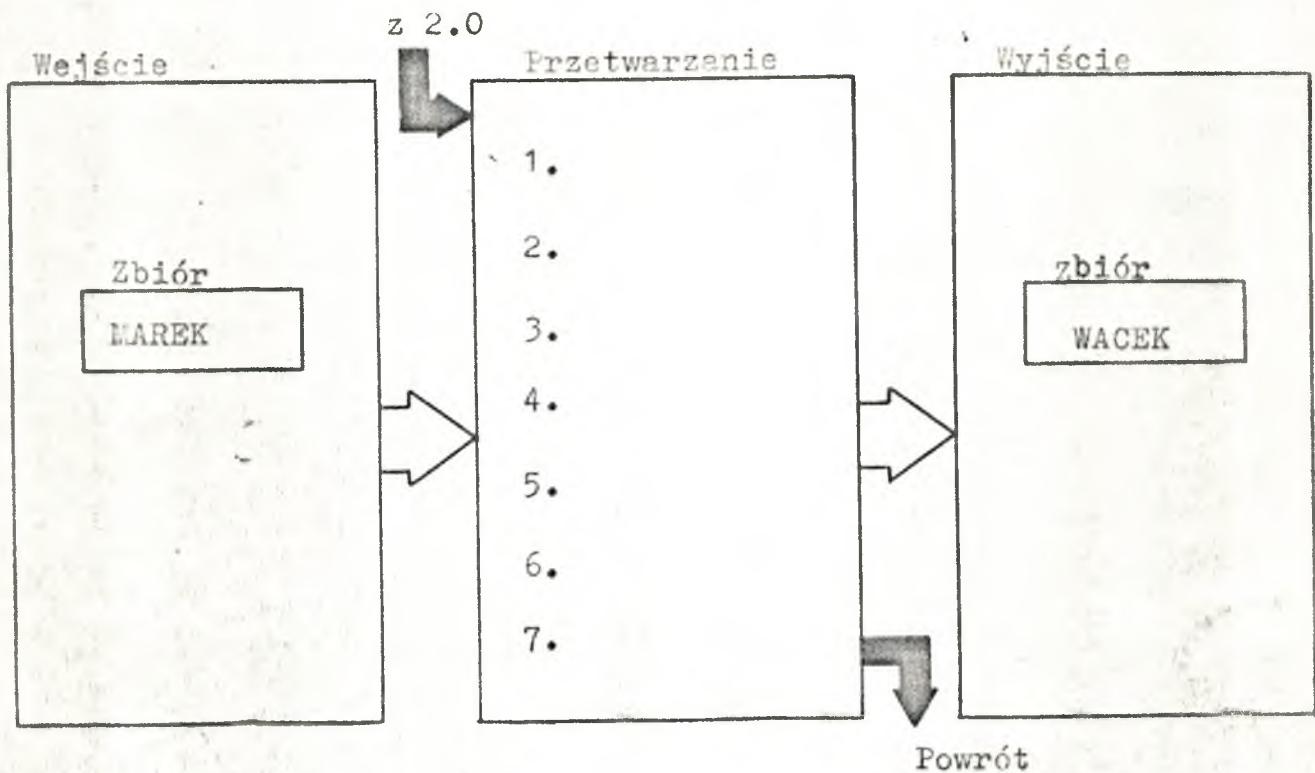
Rys. 12 Rodzaje strzałek

W części wyjścia powinna zaczynać się na wysokości opisu odpowiedniej operacji i kończyć przy lewej linii prostokąta danych w sekcji wejścia /zob.rys.13/.



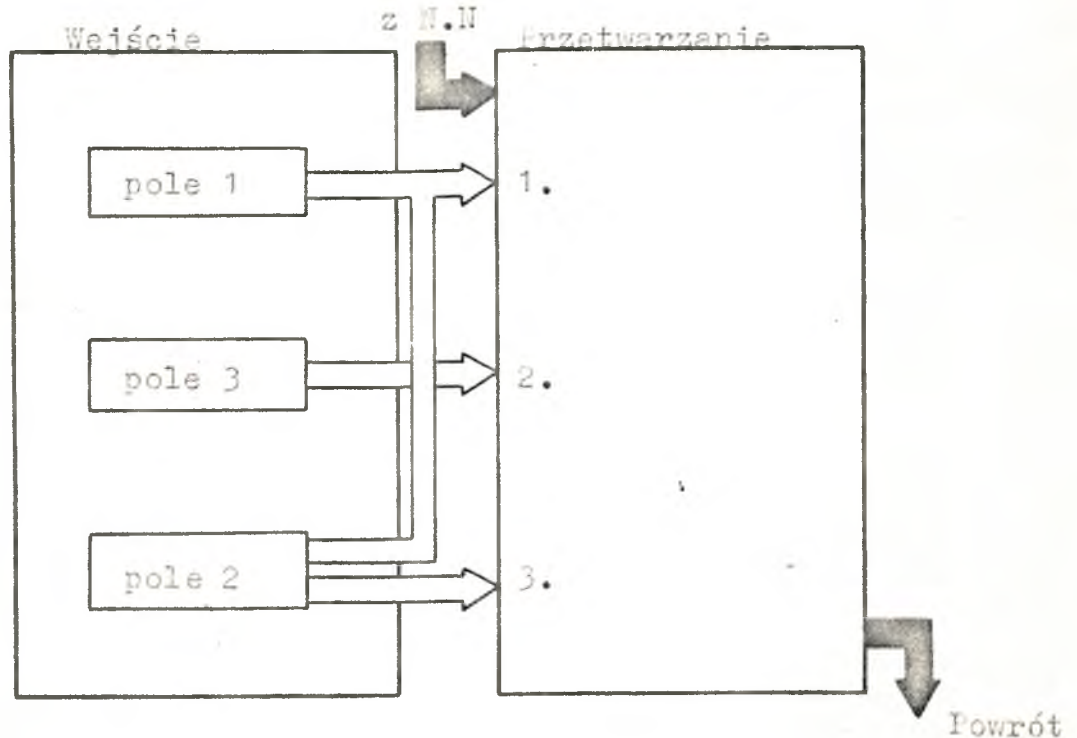
Rys. 13 Nieprawidłowe i poprawne użycie strzałek przesłania danych

- Jeśli jest tylko jedna strzałka przesłania danych w diagramie, winna ona zaczynać się od "ramki" sekcji wejścia i kończyć przy "ramce" sekcji przetwarzania, oraz zaczynać od "ramki" sekcji przetwarzania i kończyć przy "ramce" sekcji wyjścia /zob.rys.14/. Jest to najczęściej stosowane w diagramach ogólnych.



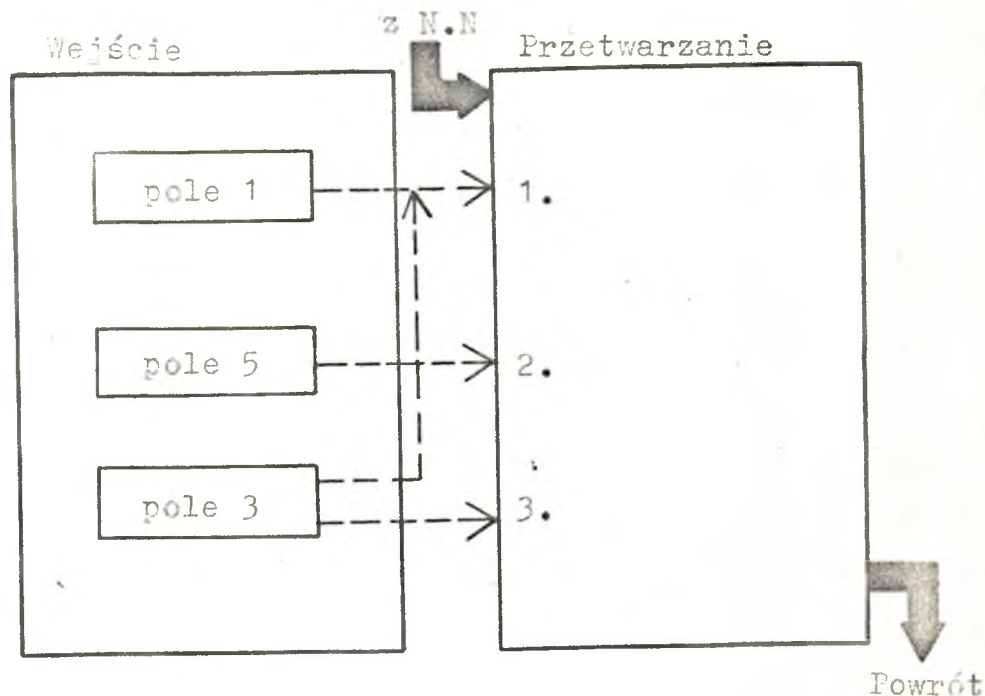
Rys. 14 Użycie jednej strzałki przesłania danych

- Strzałki przesłania danych mogą się łączyć i przecinać /zob.rys.15/, lecz ze względu na trudne śledzenie ich "biegu" wskazane jest minimalizowanie takiego ich użycia. Dla zmniejszenia ilości strzałek przecinających się można stosować odpowiednie łączenie przesłania danych przy pomocy znaku graficznego /zob.rys.12b/.



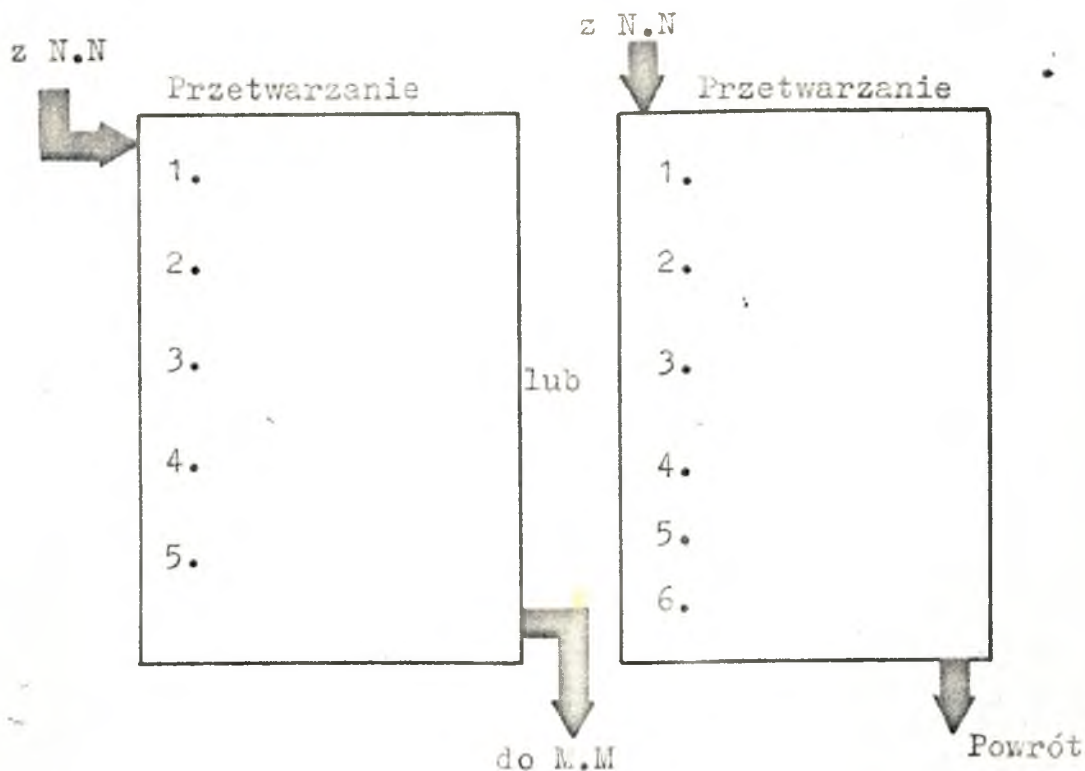
Rys. 15 Użycie strzałek łączących się i przecinających

- Nieco trudniejsze jest użycie łączących i przecinających się strzałek danych testowanych /zob.rys.3/. Dlatego należy pamiętać o dwóch zasadach:
 - jeśli jakaś strzałka łączy się z inną, łącząca się strzałka powinna kończyć się grotem /zob.rys.16/,
 - jeśli jedna strzałka przecina inną, nie powinna mieć grotu w obszarze między sekcjami.
- W przypadku wielokrotnego używania danych /wkilku operacjach/ można zastosować:
 - strzałkę wielokrotną od danych do każdego kroku,
 - łączenie przesłania danych,
 - powtórzenie elementu danych przy każdym kroku.
- Jeśli element danych jest wyjściem z jednej operacji i zarazem wejściem do innej można:
 - użyć strzałki odwróconej od danych wyjściowych do dalszej operacji,
 - powtórzyć element danych w sekcji wejścia z uwagą /np: "utworzone w operacji 2" lub " z operacji 2"/.



Rys. 16 Użycie /łączenie/ strzałek danych testowanych

- Jeśli dane w sekcji wyjścia używane są jako wejścia i wyjścia w tej samej operacji musi być użyta strzałka z grotem na obydwu końcach.
- Strzałka wskaźnika danych pokazuje adres lub inny element danych. Może ona łączyć elementy danych /zob.rys.12c/.
- Strzałka sterowania pokazuje skąd i dokąd przekazywane jest sterowanie w realizacji danej funkcji. Symbol strzałki sterowania przedstawiony jest na rys.12d.
- Punkty wejścia do diagramu i wyjścia z diagramu mogą być pokazane na dwa sposoby /zob.rys.17/. Nad strzałką górną umieszcza się numery diagramów, z których sterowanie zostało przejęte, a pod strzałką dolną numery diagramów, do których sterowanie będzie przekazane.
- Punkty przekazania sterowania z diagramu do procedury wewnętrznej /lub zewnętrznej/ przedstawia się jak na rys. 9 i 10. To użycie strzałki sterowania oznacza, że po wykonaniu procedury wykonywana będzie następna operacja w tym diagramie.



Rys. 17 Sposoby pokazania punktów wejścia do diagramu i wyjścia z diagramu

- W przypadku diagramów wielostronicowych używa się łączników strzałek sterowania /zob.rys.12e/ do pokazania, że sterowanie jest przekazywane z jednej operacji do innej wewnątrz tego samego diagramu; N.N to numer operacji, do której jest przekazywane sterowanie.
- W sytuacji, gdy sterowanie jest przekazywane z operacji w jednym diagramie do operacji w diagramie innym, używa się łącznika jak na rys.12f. "Nazwa" to nazwa funkcji, do której przekazuje się sterowanie, a ID to numer identyfikacyjny diagramu opisującego tę funkcję. Jeśli sterowanie przekazywane jest nie do pierwszej operacji, należy podać również jej numer.

4,5 Sekcja opisu rozszerzającego

Sekcja opisu rozszerzającego ma za zadanie dostarczenie dodatkowych informacji o powiązaniach poszczególnych diagramów. Może ona zawierać takie kolumny, jak opis, procedury, etykiety programowe, komunikaty, odwołania, itp.

- W kolumnie opisu można umieszczać dodatkowe informacje o poszczególnych operacjach /z sekcji przetwarzania/, a także o elementach danych.
- Informacje w sekcji opisu rozszerzającego winny mieć ten sam numer, co operacja, której dotyczą.
- Dane w kolumnie odwołań pokazują, w którym diagramie, lub w którym elemencie dokumentacji nie-HIPO można znaleźć dodatkowe informacje o danej operacji /funkcji/.
- Szczególne znaczenie posiada sekcja opisu rozszerzającego w diagramach dotyczących programów /lub modułów programowych/. Można w niej umieścić wszystkie nazwy etykiet, pól, procedur, itp, z których program będzie korzystał. W obszarze opisu można również przedstawić wybrane bloki programu, zapisane w pseudokodzie.

Bibliografia

1. T.Falborska, HIPO jako technika dokumentowania hierarchicznego tworzenia systemu, OBRI, Wyrzawa 1977
2. HIPO - A Design Aid and Documentation Technique, Installation Management IBM GC20-1851-1
3. IBM, Materiały szkoleniowe
4. Stevens, W.P. and Myers, G.J. and Constantine, L.L., Structured Design, IBM Systems Journal, Vol. 13 No. 2 1974



Proj. 729
Egz. B