



2

OŚRODEK BADAWCZO - ROZWOJOWY INFORMATYKI
ZAKŁAD METODYKI I EKONOMIKI SYSTEMÓW



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

Proj. 727
Egz. B

HIPO JAKO TECHNIKA DOKUMENTOWANIA
HIERARCHICZNEGO TWORZENIA SYSTEMU

Autor:

Kierownik Zakładu:

mgr T. Falborska

mgr inż. H. Zygiel

Warszawa, luty 1977



Spis treści

1. Wprowadzenie	3
2. Technika HIPO	5
3. Rodzaje diagramów HIFC	6
3.1 Diagram VTOC	6
3.2 Diagramy IPO	10
4. Warunki stosowania techniki HIPO	15
5. Rodzaje pakietów HIPO	17
5.1 Wstępny pakiet projektowy	17
5.2 Szczegółowy pakiet projektowy	17
5.3 Pakiet konserwacyjny	20
Bibliografia	22
Załącznik	



1. WPROWADZENIE

Od dosyć dawna informatycy zaczęli zastanawiać się nad podniesieniem "elegancji" i zmniejszeniem kosztów programowania. Z wielu doświadczeń wynikało, iż najwięcej zmian w strukturze systemu powstaje na etapie programowania, testowania i wdrażania, a nierzadko - eksploatacji. Modyfikacja zaś na tych etapach jest bardzo kosztowna, ponieważ zbyt późne wykrycie błędów projektowych zmusza często do wnoszenia poprawek począwszy nawet od analizy zagadnienia. Stąd potrzeba większego skoncentrowania się na projektowaniu, nie tylko poprzez poświęcenie mu większej ilości czasu, ale stworzenie takich technik projektowania i dokumentowania, które pozwoliłyby na szybką i łatwą weryfikację projektowanego systemu oraz zrozumienie jego funkcji przez nieinformatyków. Tak postawiony problem powoduje również zmianę podziału czasu przeznaczanego na kolejne etapy pracy nad systemem.

Zakładając, że całość prac związanych z tworzeniem systemu informatycznego można /w przypadku ogólnym/ podzielić na trzy etapy: projektowanie, programowanie oraz testowanie i wdrożenie, to przy tradycyjnych metodach pracy czas dzielony był odpowiednio /orientacyjnie/: 30%, 40%, 30%. Stosując IPT^{1/} podział ten ulega zmianie na układ: 40%, 30%, 30%, a nawet 50%, 30%, 20% [3]. Pozwala to na uzyskanie istotnych efektów prac w zakresie tworzenia oprogramowania /głównie użytkowego/, począwszy od sformułowania zadania projektowego, poprzez analizę, opracowanie założeń, projektowanie, programowanie, aż po wdrożenie i konserwację.

Pod pojęciem IPT rozumie się zbiór metod, technik i procedur, wśród których można wymienić [1] sześć głównych

1/ IPT - IMPROVED PROGRAMMING TECHNOLOGY

S. Gałka w Informatyce nr 10/76r używa skrótu UTO /Ulepszoa Technologia Oprogramowania/.

W niniejszym opracowaniu zostaną zachowane skróty w wersji oryginalnej.



elementów, a mianowicie:

- . hierarchiczne tworzenie systemu /TOP-DOWN DEVELOPMENT/,
- . dokumentacja HIPO /HIERARCHY + INPUT-PROCESS-OUTPUT/,
- . weryfikacja strukturalna /STRUCTURED WALK-THROUGH/,
- . programowanie strukturalne /STRUCTURED PROGRAMMING/,
- . organizacja zespołów wykonawczych,
- . biblioteka programów i dokumentacji /DEVELOPMENT SUPPORT LIBRARY/.

Stosowanie powyższych elementów jest w pewnym sensie dowolne; można je stosować łącznie lub tylko wybrane spośród nich. Zaleca się jednak stosowanie IPT w całości, szczególnie zaś trzy pierwsze elementy powinny być traktowane łącznie jako metoda tworzenia i dokumentowania oprogramowania. W tym sensie HIPO jest techniką dokumentowania stosowaną dla metody hierarchicznego tworzenia systemu i przystosowaną do potrzeb weryfikacji strukturalnej.

Niniejsze opracowanie nie zawiera dokładnego omówienia wszystkich elementów IPT. Zajmuje się tylko wybranymi jego aspektami, niezbędnymi dla dokładniejszego poznania i zrozumienia zagadnień związanych bezpośrednio z HIPO.



2. TECHNIKA HIPO

Hierarchiczne tworzenie systemu /nazywane również projektowaniem hierarchii systemu zdeterminowanej jego funkcjami/ to powtarzająca się działalność "od ogółu do szczegółu" na drodze rozwijania i przybliżania poszczególnych funkcji. Począwszy od głównych funkcji systemu /które z reguły są funkcjami złożonymi/ poprzez kolejne poziomy dochodzi się do funkcji prostych.

HIPO zaś jest techniką graficznego obrazowania tych poczynań i to techniką czynnie zaangażowaną w proces projektowania.

Ponieważ w dalszych rozważaniach jednym z podstawowych pojęć jest pojęcie funkcji, konieczne jest sprecyzowanie jego znaczenia.

Funkcja może być zdefiniowana jako działanie na obiekcie, a dla potrzeb tego opracowania, jako przekształcanie dowolnych danych wejściowych na jakieś dane wynikowe. Tak rozumiana funkcja /każdego poziomu/ opisuje raczej co trzeba zrobić, niż jak to zrobić i nie należy jej utożsamiać z pojęciem algorytmu.

Proces iteracji głównych funkcji systemu wymaga zręczności analitycznej i wyobraźni. Każdy z projektantów prace te może prowadzić w nieco inny sposób, a to powoduje możliwość wyboru i otrzymanie rozwiązanie najkorzystniejszego. Naturalnym efektem tak prowadzonych prac projektowych jest rysowanie "drzewka" funkcyjnego, które doprowadzone do ostatecznej postaci w dokumentacji HIPO przyjmuje kształt odpowiedniego diagramu.



3. RODZAJE DIAGRAMÓW IIFO

Podstawą dokumentacji projektu sporządzonego metodą IIFO są dwa rodzaje diagramów: VTOC^{1/} i diagramy IPO^{2/} /ogólne i szczegółowe/. Ich zakres tematyczny i stopień szczegółowości jest różny w zależności od etapu prac, na którym są wykonywane. Sprawa ta dokładniej zostanie przedstawiona przy okazji omawiania poszczególnych pakietów dokumentacyjnych.

3.1 Diagram VTOC

Diagram VTOC /patrz rys.1/ zawierać musi całą strukturę /architekturę/ systemu wraz z nazwami i numerami identyfikacyjnymi wszystkich diagramów ogólnych i szczegółowych. Obrazuje graficznie zależności między funkcjami w ich hierarchii. Ważnym jego elementem jest legenda, czyli wykaz stosowanych w pakiecie symboli wraz z ich dokładnym wyjaśnieniem. VTOC może również zawierać bliższe określenie głównych funkcji, jeśli ich nazwy są mało zrozumiałe, oraz ogólny wykaz używanych w systemie danych. Poprzez VTOC czytelnik może łatwo odnaleźć interesujące go funkcje na dowolnym poziomie, tj. dotrzeć do informacji o odpowiednim stopniu szczegółowości bez przeglądania całego pakietu.

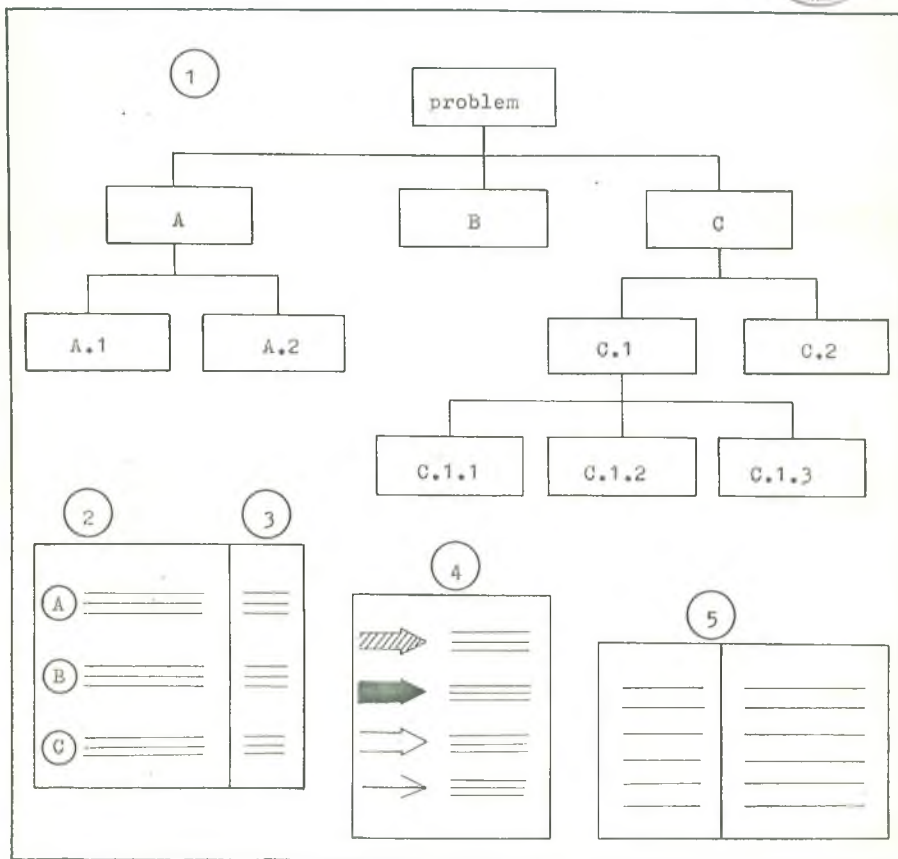
Ponieważ wszystkie poziomy systemu są opisane jako funkcje, mogą być więc zgrupowane w trzy kategorie, poczynając od poziomu najwyższego - ogólnego do poziomów niższych - szczegółowych, jak następuje:

- a/ poziom systemu,
- b/ poziom programu,
- c/ poziom modułu /programowego/.

1/ VTOC - VISUAL TABLE OF CONTENTS

W dowolnym tłumaczeniu "poglądowa tablica zawartości".

2/ IPO - INPUT-PROCESS-OUTPUT



- 1 logiczna struktura problemu
- 2 tablica zawartości - szerszy opis funkcji najwyższego poziomu
- 3 uwagi, odwołania do innych dokumentów
- 4 legenda - wykaz symboli używanych w pakiecie
- 5 ogólny wykaz danych używanych w systemie

Rys. 1 .Diagram VTCC



Są to oczywiście poziomy "pojęciowe", a nie fizyczne poziomy funkcji. Każdy poziom "pojęciowy" może być reprezentowany przez wiele poziomów funkcji w strukturze logicznej, jak również każdy "prostokąt" na wykresie może być "dołem" poziomu wyższego, a także "góram" poziomu niższego.

a/ Poziom systemu

Poziom systemu w strukturze logicznej /hierarchii funkcji/ może zawierać wybrane lub wszystkie główne funkcje systemu i jest takim spojrzeniem na problem, jakie chce mieć użytkownik. Pomimo, że terminologia stosowana przez użytkownika może różnić się nieco od terminologii funkcji, w dokumentacji systemu należy posługiwać się tą ostatnią, ponieważ jest ona bardziej precyzyjna. W ogólnym przypadku poziom systemu nie zawiera reprezentacji jakiejś funkcji nadającej się do bezpośredniego zaprogramowania. Przedstawia on ogólną koncepcję systemu /ogólne spojrzenie na rozwiązanie problemu/. Jeśli w trakcie dalszych prac nad systemem okaże się, że obraz struktury systemu jest komunikatywny dla użytkownika lecz nie daje możliwości projektowania programów, należy wymienić odpowiednie elementy struktury w ten sposób, aby spełniały one wymagania grup projektowo-programowych i jednocześnie zachowały komunikatywność.

b/ Poziom programu

Poziom programu jest rezultatem rozwinięcia ogólnej koncepcji systemu. Może być charakteryzowany jako ostateczny poziom dla użytkownika. "Prostokąty" reprezentujące poziom programu zawierają funkcje zazwyczaj nadające się do bezpośredniego zakodowania i dlatego często nazywa się je programami. Poziom programu, wstępnie określony w pierwszej fazie projektowania, może być w dalszym ciągu prac rozwijany lub modyfikowany, w zależności od potrzeb programistów, jednakże zawsze z uważnym śledzeniem, czy poczynione zmiany w pełni gwarantują spełnienie wszystkich zadanych funkcji.



c/ Poziom modułu

Do poziomu modułu dochodzimy uszczegóławiając poziom programu. Ponieważ projektowanie strukturalne jest procesem iteracji, kolejne uszczegółowienie zmusza często do ponownego przeglądu całej struktury. Oznacza to, że rozbudowywanie poziomu programu powoduje jeszcze jedną kontrolę poprawności wykonanej pracy. Diagram VTOC jest więc weryfikowany poprzez sam proces projektowania.

Poziom modułu reprezentuje segment wykonawczy programu, będący zazwyczaj samodzielnym elementem. Element ten nazywany jest OBJECT MODULE /moduł wynikowy/ dla odróżnienia go od LOAD MODULE /moduł załadowczy/, który może być konstruowany poprzez łączenie kilku modułów wynikowych. Poprawne uszczegółowienie projektu na szczeblu poziomu modułu pozwala na kodowanie programów bezpośrednio z dokumentacji HIPO. W fazie projektu technicznego poziom modułu należy rozbudowywać do takiego stopnia, aby poszczególne moduły po zakodowaniu w języku wyższego rzędu nie zawierały więcej, jak 50 instrukcji.

Jak wynika z powyższego diagram VTOC jest dokumentem "uniwersalnym" służącym wszystkim uczestnikom procesu projektowania systemu:

- poziom systemu ma sens i znaczenie przede wszystkim dla użytkownika, dzięki odpowiedniej budowie i właściwej terminologii;
- poziom modułu pozwala programiście na bezpośrednie kodowanie;
- poziom programu dostarcza użytkownikowi szczegółów, programiście spojrzenia bardziej ogólnego. Jednocześnie zaś w procesie uszczegóławiania funkcji łączy poziom systemu z poziomem modułu.

Należy jednak podkreślić, że stopień realizacji tych zadań zależy od zakresu tematycznego i stopnia szczegółowości diagramu, tj od etapu prac, na którym jest on wykonywany /o czym już wyżej wspomniano/.



3.2 Diagramy IFO

W trakcie projektowania jednocześnie ze zdefiniowaniem funkcji określone są dane, których system ma dostarczać, oraz dane, których będzie potrzebował jako wejść. Związki między poszczególnymi danymi wejściowo/wyjściowymi a funkcjami systemu przedstawiane są na diagramach dwóch rodzajów:

- a/ diagramach ogólnych IFO /patrz rys.2/,
- b/ diagramach szczegółowych IFO /patrz rys.3/.

Diagramy ogólne wykonuje się dla wszystkich funkcji, które posiadają dalsze rozwinięcia. Dla funkcji nie posiadających dalszych rozwinięć, tj dla funkcji prostych, wykonuje się diagramy szczegółowe.

a/ Diagramy ogólne IFO

Opisują one główne funkcje systemu oraz obrazują związki między tymi funkcjami a danymi wejściowymi i wyjściowymi.

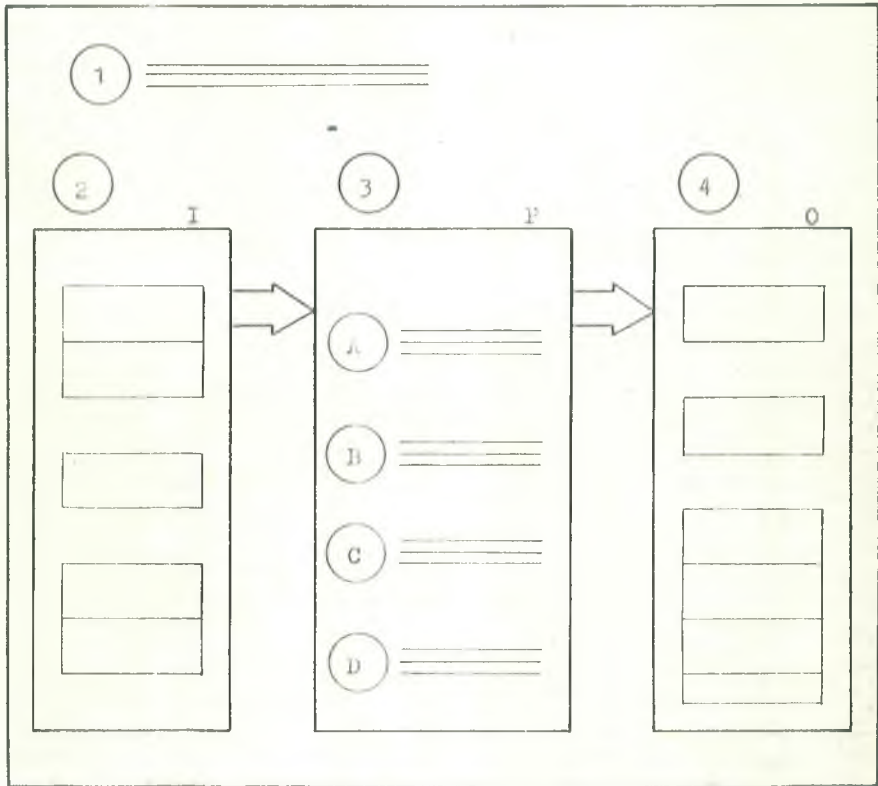
Pojedynczy diagram ogólny IFO składa się z trzech sekcji: wejścia/I/, przetwarzania/P/ i wyjścia/O/.

Sekcja przetwarzania zawiera ciąg ponumerowanych kroków, które są potrzebne do spełnienia opisywanej funkcji. Kroki te to najczęściej podfunkcje, otrzymane na drodze rozwijania funkcji głównej. Zawartość tej sekcji jest inna w diagramach IFO dla funkcji na poziomie systemu, a nieco inna dla funkcji na poziomie programu. Sekcja przetwarzania zawiera również odwołania do diagramów niższego poziomu lub czasami do dokumentacji nie związanej bezpośrednio z MIPO.

Sekcja wejścia zawiera informacje o danych wejściowych, niezbędnych do realizacji wszystkich podfunkcji.

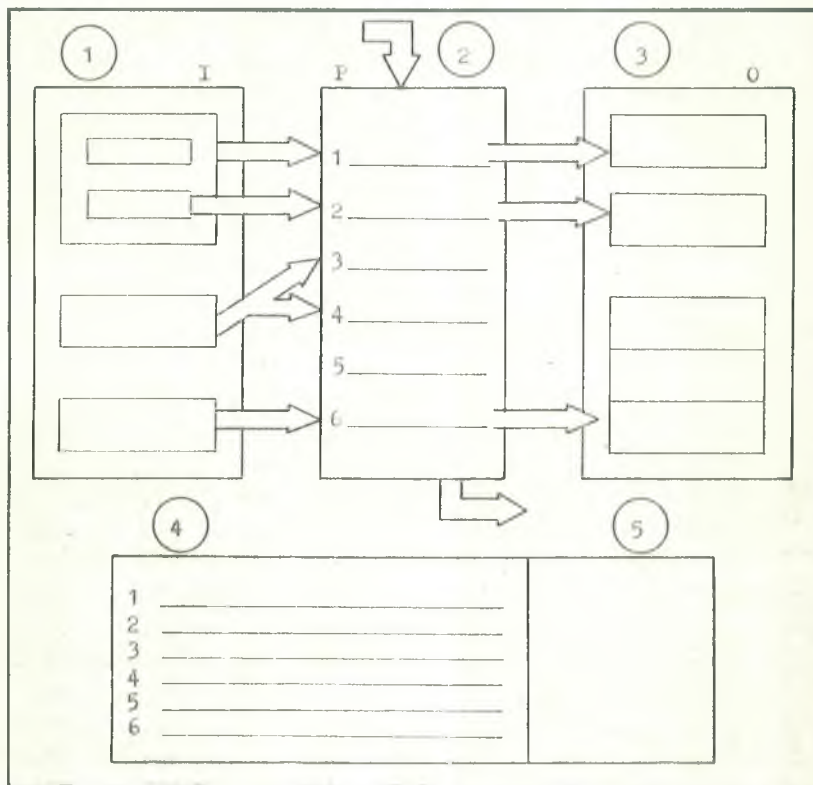
Sekcja wyjścia zawiera informacje o danych wyjściowych tworzonych lub modyfikowanych przez poszczególne podfunkcje.

Strzałki wskazują odpowiednie powiązania między elementami tych trzech sekcji.



- ① - nazwa funkcji opisanej w diagramie
- ② sekcja wejścia
- ③ sekcja przetwarzania - podfunkcje i odwołania do diagramów niższego poziomu
- ④ sekcja wyjścia

Rys. 2 Diagram ogólny IPO



- 1 sekcja wejścia - szczegółowe elementy danych
- 2 sekcja przetwarzania - szczegółowe funkcje
- 3 sekcja wyjścia - szczegółowe elementy danych
- 4 opis rozszerzający
- 5 uwagi

Rys. 3 Diagram szczegółowy IFO



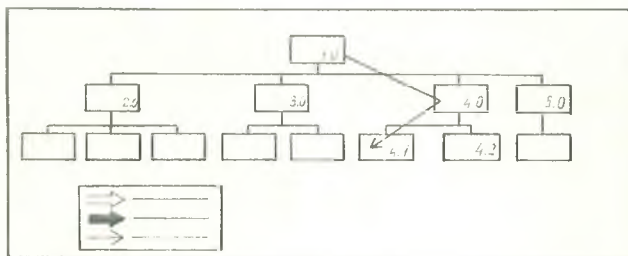
Poziom szczegółowości informacji zawartych w diagramach ogólnych IPO jest zależny od etapu, na którym są wykonywane.

b/ Diagramy szczegółowe IPO

Różnią się od diagramów ogólnych poziomem szczegółowości - opisują szczegółowe podfunkcje, szczegółowe elementy danych wejściowych i wyjściowych oraz obrazują związki między nimi. Ponadto, poza sekcją wejścia, przetwarzania i wyjścia zawierają dodatkowo sekcję opisu rozszerzającego.

Związki między elementami poszczególnych sekcji obrazowane są przy użyciu różnego rodzaju strzałek. Opis rozszerzający daje dodatkowe wyjaśnienia. Dla funkcji na poziomie programu sekcja przetwarzania diagramu IPO zawiera podfunkcje poziomu modułu. Natomiast dla funkcji na poziomie modułu zawiera kolejne kroki przetwarzania, wykonywane w danym module. Opis rozszerzający wyjaśnia użyte skróty i pojęcia, które należy przybliżyć czytelnikowi oraz zawiera odwołania do innych diagramów szczegółowych, a także do dokumentacji z NIPO bezpośrednio nie związanej, takiej jak tablice decyzyjne, projekty zapisów, projekty zbiorów, tablice wejść itp.

Związek między diagramem VTOC, diagramami ogólnymi i diagramami szczegółowymi IPO przedstawia rys.4.



VTCC

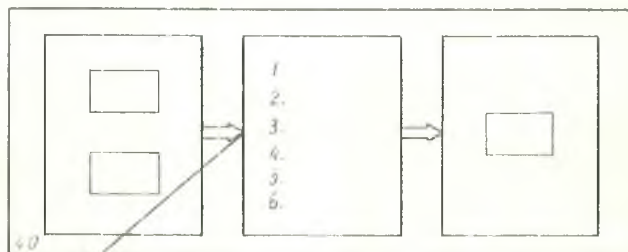


DIAGRAM OGÓLNY

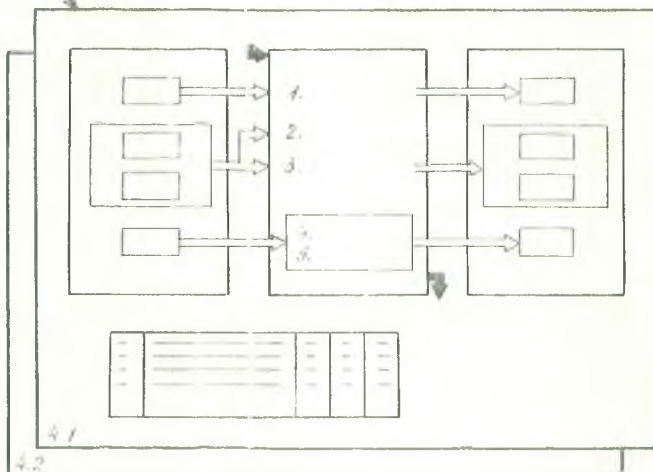


Diagrama - SZCZEGÓLNE

Dys. 4 Związki między diagramami JFC



4. WARUNKI STOSOWANIA TECHNIKI HIPO

Dokumentacja może być wykonana techniką HIPO na wszystkich etapach pracy nad projektem. W zależności od ilości niezbędnych informacji, odbiorców oraz zakresu wykonywanego projektu, dokumentacja ma zróżnicowany skład i stopień szczegółowości.

Potrzeby dokumentacyjne są różne dla różnych etapów projektowania i dlatego w niniejszym opracowaniu omówione zostaną one dla:

- określania żądań /zadania projektowego/,
- założeń systemu,
- projektu technicznego,
- wdrożenia,
- konserwacji systemu.

Dokumentacja wykonana na jednym etapie jest materiałem wejściowym do prac prowadzonych na etapie następnym.

. Określanie żądań i założenia systemu

Na etapie sporządzania założeń do projektowanego systemu nacisk kładziony jest na zabezpieczenie łatwego zrozumienia całości przedsięwzięcia zarówno przez projektantów jak i przez przyszłych użytkowników systemu. Należy również zwrócić uwagę na to, aby projektowany system w pełni realizował wszystkie stawiane mu wymagania. Dokumentacja HIPO, pozwalając na łatwość odczytania często dosyć trudnych zagadnień, świetnie służy jako materiał poglądowy w czasie rozmów roboczych projektantów i użytkowników. Pozwala na ścisłą współpracę między tymi grupami ludzi oraz daje możliwość szybkiego wyszukiwania braków, a także punktów niezrozumiałych. Często poprawki wynikające z uprzednich niedomówień lub niezrozumień mogą być nanoszone bez zachwiania ogólnej konstrukcji przedsięwzięcia.

Etap kończy się z chwilą poprawnego zdefiniowania wszystkich głównych funkcji systemu w powiązaniu z odpowiednimi elementami danych wejściowych i wyjściowych. Jednocześnie prace są udokumentowane w sposób przejrzysty, czytelny i zrozumiały.



. Projekt techniczny i wdrożenie

W tych etapach rozwijane są uprzednio zdefiniowane funkcje. Rozwijanie i uszczegóławianie prowadzi się do poziomu, który pozwala programistom kodować poszczególne moduły bezpośrednio z dokumentacji. Diagramy HIPO projektu technicznego pokazują wszystkie elementarne powiązania danych wejściowych, funkcji i danych wyjściowych oraz powiązania między poszczególnymi funkcjami. Podczas prac nad rozwijaniem funkcji zdefiniowanych w pierwszym etapie dokonywana jest ich kolejna weryfikacja. Na tym etapie odbywa się przybliżanie funkcji systemu do potrzeb przetwarzania. Dokumentacja służy grupom programistów i operatorów. Poprzez swą czytelność daje możliwość szybkiego zapoznania się z ogólną koncepcją systemu jak i z jego rozwiązaniami szczegółowymi. Dodatkowo, diagramy HIPO łatwo można rozszerzać dla wprowadzenia nowych członów do projektu.

Dokumentacja tego etapu pozwala na szybkie zapoznanie z projektowanym systemem osób, które zostały dołączone do grupy opracowującej, a także osób, które będą go w przyszłości eksploatowały.

. Konserwacja

W trakcie eksploatacji i konserwacji systemu dokumentacja HIPO pozwala na szybkie zlokalizowanie wykrytych błędów. Ponieważ diagramy HIPO pokazują, co w danym miejscu system robi, a nie jak to robi, nanoszenie poprawek i zmian sprowadza się najczęściej do modyfikacji opisów rozszerzających, bez zachwiania struktury logicznej systemu.



5. RODZAJE PAKIETÓW HIPO

Ogólnie rozróżnia się dwa zasadnicze rodzaje pakietów HIPO - wstępny pakiet projektowy /INITIAL DESIGN PACKAGE/ i szczegółowy pakiet projektowy /DETAIL DESIGN PACKAGE/ oraz opcjonalnie trzeci pakiet konserwacyjny /MAINTENANCE PACKAGE/, który powstaje w trakcie zmian systemu na etapie wdrożenia i eksploatacji. Każdy z tych pakietów zawiera wszystkie rodzaje diagramów HIPO, lecz każdy ma różne przeznaczenie, inny stopień rozbudowania, odmienną charakterystykę, a także innego odbiorcę.

5.1 Wstępny pakiet projektowy

Sporządzany jest przez niewielką grupę ludzi /2 lub 3 osoby/, analityków i projektantów, w pierwszym etapie pracy nad projektem. Opisuje główne funkcje systemu oraz wiąże je z danymi wejściowymi i wyjściowymi. Dane wejściowe i wyjściowe określane są w sposób ogólny, najczęściej bez podawania nośników, na jakich będą się w przyszłości znajdować. Podobnie funkcje i podfunkcje formułowane są ogólnie, chociaż jasno i wyraźnie.

Wstępny pakiet projektowy służy do zapoznania kierownictwa jednostki informatycznej z prowadzonymi przez zespół pracami oraz jako materiał "roboczy" do konsultacji i uzgodnień z użytkownikiem. Zatem potrzeby i wymagania użytkownika w szczególny sposób determinują zakres informacyjny i stopień szczegółowości tego pakietu.

Wstępny pakiet projektowy jest wynikiem prac prowadzonych nad ustaleniem i sporządzeniem założeń do projektowanego systemu. Powstaje równoległe jako uboczny produkt procesu projektowania, a nie stanowi dodatkowego zadania.

W przybliżeniu można go porównać do dokumentacji znanej pod określeniem "Założenia systemu informatycznego".

5.2 Szczegółowy pakiet projektowy

Powstaje w trakcie prac grupy rozwijającej projekt, na bazie kompletnego i uzgodnionego z użytkownikiem wstępnego pakietu projektowego. Projektanci i programiści



określają szczegółowe funkcje systemu, wiążąc je jednocześnie z potrzebami technicznymi przetwarzania. Rozszerzają architekturę systemu, budując szczegółowy VFCO do poziomów modułu. Rozbudowują istniejące diagramy IFC oraz dołączają nowe dla rozszerzonych poziomów VFCO. Diagramy IIO dla modułów zawierają szczegółowe kroki przetwarzania, a także dokładny opis elementów danych wejściowych i wyjściowych.

Łącząc między danymi wejściowymi, makroinstrukcjami^{1/} i danymi wyjściowymi w diagramach szczegółowych wskazują logiczny przepływ informacji i kolejność wykonywania poszczególnych makroinstrukcji. Opis rozszerzający uzupełniają etykiety programowe oraz inne informacje niezbędne do kodowania modułów i łączenia ich w programy.

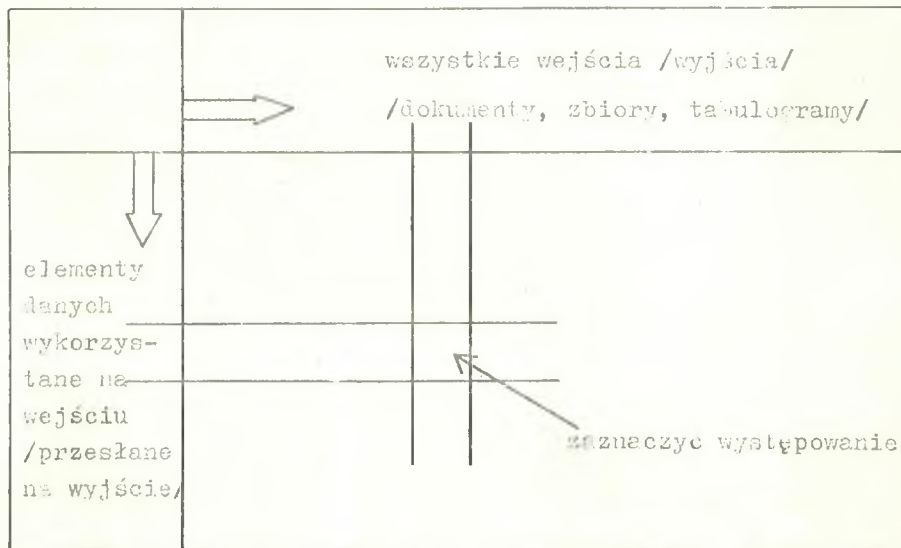
Oprócz diagramów HIPO w skład pakietu szczegółowego wchodzi również często elementy dokumentacji nie związanej bezpośrednio z HIPO, takie jak: projekty zbiorów, projekty dokumentów źródłowych i wynikowych, itp. W przypadku dużego systemu i złożoności problematyki jaką obejmują, w trakcie prac tworzyć można i dołączać do dokumentacji różnego rodzaju tablice, jak np:

- . tablice danych wejściowych i tablice danych wyjściowych /patrz rys.5/ - wskazujące przyporządkowanie elementów danych wejściowych/wyjściowych/ poszczególnym dokumentom,
- . tablice opisu danych /patrz rys.6/ - wykonywane dla każdego programu /lub modułu/, przypisujące poszczególnym elementom danych ich cechy charakterystyczne,
- . tablice decyzyjne - opisujące związki przyczynowo - skutkowe w poszczególnych funkcjach systemu.

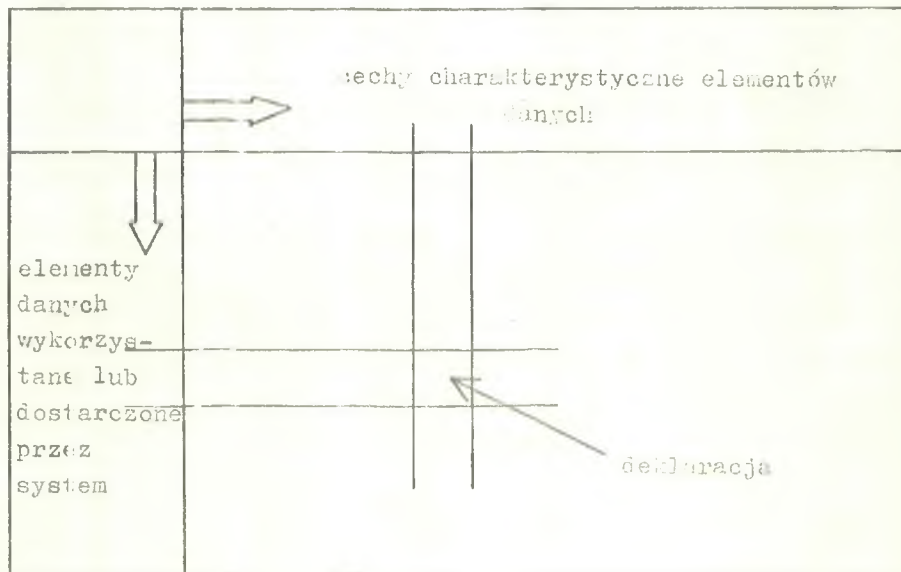
Tak wykonany pakiet szczegółowy służy programistom do kodowania modułów i programów, jak również do pisania instrukcji dla użytkowników. Często jest on końcową /ostateczną/ dokumentacją projektu i służy również grupom konserwującym system.

"Szczegółowy pakiet projektowy swymi cechami i zastosowaniem odpowiada dokumentacji znanej pod nazwą "Projektu technicznego".

1/ makroinstrukcje - sekwencja rozkazów /często zapisanych w pseudo-kodzie/ wyrażająca funkcje najniższych poziomów.



Rys. 5 Tablica danych wejściowych /wyjściowych/



Rys. 6 Tablica opisu danych

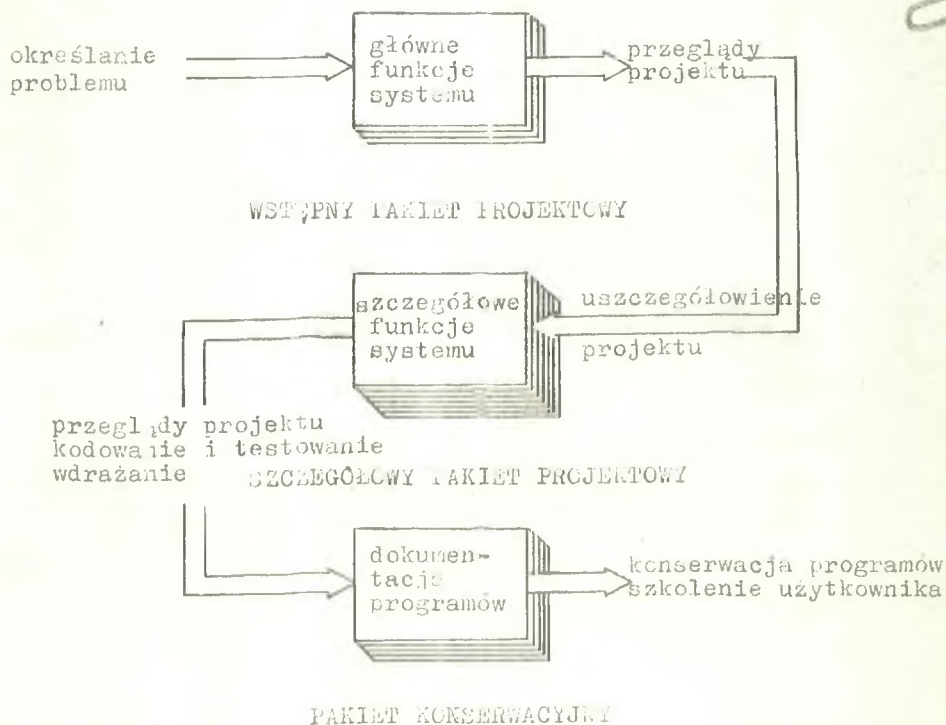


5.3 Pakiet konserwacyjny

Powstaje w trakcie modyfikacji /zarówno zmian jak i rozbudowy/ systemu. Jeśli występuje samodzielnie, jest to szczegółowy pakiet projektowy z wprowadzonymi zmianami na etapie programowania, testowania i wdrożenia bądź konserwacji.

Może dodatkowo zawierać wydruki programów, dane testowe, dane techniczne przebiegów testowych oraz uzupełniające instrukcje użytkowe.

Związki między rodzajami pakietów BIIO obrazuje rys.7.



Rys. 7 Związki między rodzajami pakietów BIIO



... i inne ...
... w postaci ...
...

- a/ ...
- b/ po ...
- c/ przedstawia funkcje projektowanego systemu w sposób systematyczny, używając struktury hierarchicznej, która nie zniekształca relatywnego ich znaczenia,
- d/ pozwala na łatwe i szybkie wykrywanie błędów, zarówno w czasie opracowywania jak i eksploatacji systemu,
- e/ pomaga w szybkim lokalizowaniu błędów programów i łatwą modyfikację programów już działających,
- f/ ułatwia i skraca proces programowania,
- g/ ułatwia testowanie poprzez możliwość budowania przypadków testowych bezpośrednio z diagramów HIPO.

Przykład zastosowania techniki HIPO do sporządzania założeń systemu umieszczony jest w Załączniku.



Bibliografia

1. S.Gałka, Ulepszona Technologia Oprogramowania,
Informatyka nr 10/76r
2. HIPO - A Design Aid and Documentation Technique,
Installation Management IBM GC20-1851-1
3. IBM, Materiały szkoleniowe
4. M.N.Jones, HIPO for Developing Specifications,
Datamation 1976, nr 3
5. Dr.D.S.Koreinann, HIPO - eine Entwurfs- und
Dokumentationstechnik für komplexe Programme,
Angewandte Informatik 4/76
6. J.F.Stay, HIPO and integrated program design,
IBM Systems Journal, Volume Fifteen/Number Two/1976



Z A Ł A C Z N I K

Wstępny pakiet projektowy HIPO
dla systemu "ZAKUP TOWARÓW"
/przykład/



OPIS OGÓLNY ZADANIA

Zadaniem systemu "Zakup towarów" jest kontrola stanów magazynowych, sporządzanie zleceń zakupu, kontrola przyjęcia towarów oraz kontrola poprawności faktury i wystawianie zleceń przelewu.

1. Sporządzanie zleceń zakupu

Każdy towar przypisany dostawcy w zbiorze dostawców podlega badaniu celem zdecydowania, czy konieczne jest jego zamówienie. Jeśli stan magazynu jest niższy od maksymalnego stanu zapasów, należy sprawdzić, czy stan magazynu powiększony o ilość towaru już zamówionego przekracza średni stan zapasów. Nie może wystąpić sytuacja, że zostanie zamówiona taka ilość towaru, iż średni stan zapasów zostanie przekroczony o ponad 10%.

Dla każdego wystawionego zlecenia zakupu tworzony jest zapis zlecenia zakupu.

2. Kontrola przyjęcia

Dostawca jest upoważniony do przysłania towaru, jeśli istnieje zapis zlecenia zakupu. Należy drukować komunikat błędu w przypadku dostawy nierejestrowanej, a dokumentów takiej dostawy nie uwzględniać w dalszym przetwarzaniu. Dostawa jest poprawna, jeśli istnieje zapis zlecenia zakupu danego towaru i ilość przysłana nie przekracza ilości zamówionej.

Dla każdej poprawnej dostawy należy utworzyć zapis przyjęcia, zmniejszyć ilość zamówioną w zapisie zlecenia zakupu, a także w zbiorze asortymentowym zwiększyć stan magazynu i zmniejszyć ilość już zamówioną.

3. Kontrola finansowa

Nie może być przyjęta faktura z kodem dostawcy, dla którego nie istnieje zapis przyjęcia. Jeśli ilość zafakturowana nie zgadza się z ilością przyjętą lub jeśli



wartości poszczególnych pozycji i całkowita wartość faktury są źle obliczone należy wydrukować komunikat błędu i nie przetwarzać danych z takiej faktury.

Wartość faktury należy zmniejszyć o rabat, który wynosi:

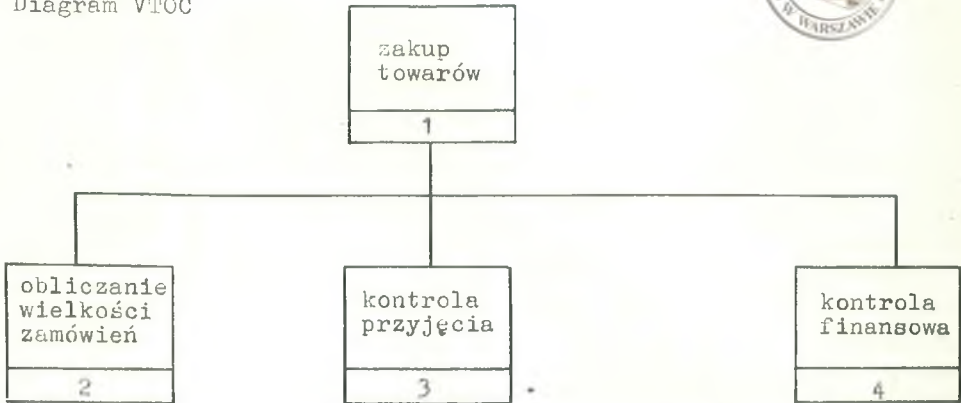
2% jeśli zostanie opłacona w ciągu 14 dni

1% jeśli zostanie opłacona w ciągu 30 dni

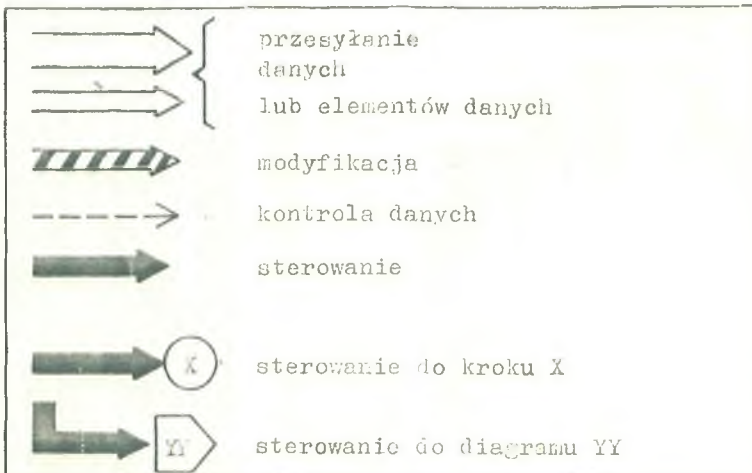
Ilość przyjętą w zapisie przyjęcia redukuje się do zera dla towarów zapłaconych.

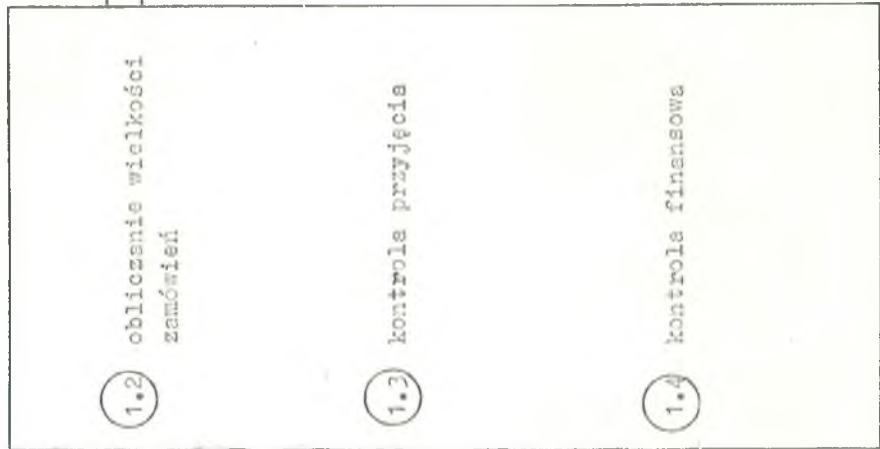
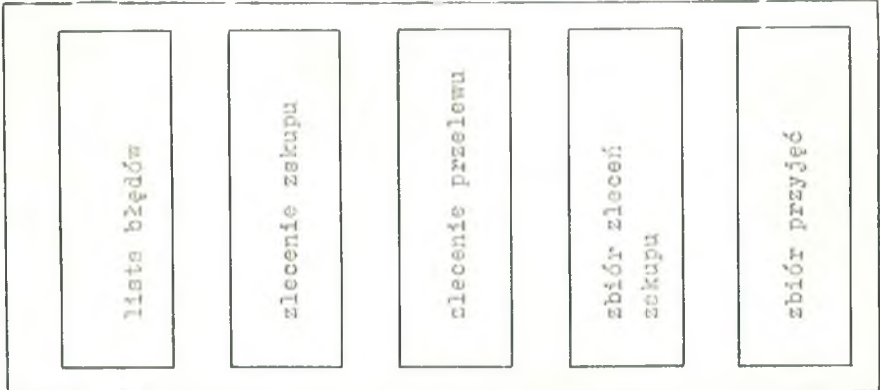


Diagram VTOC



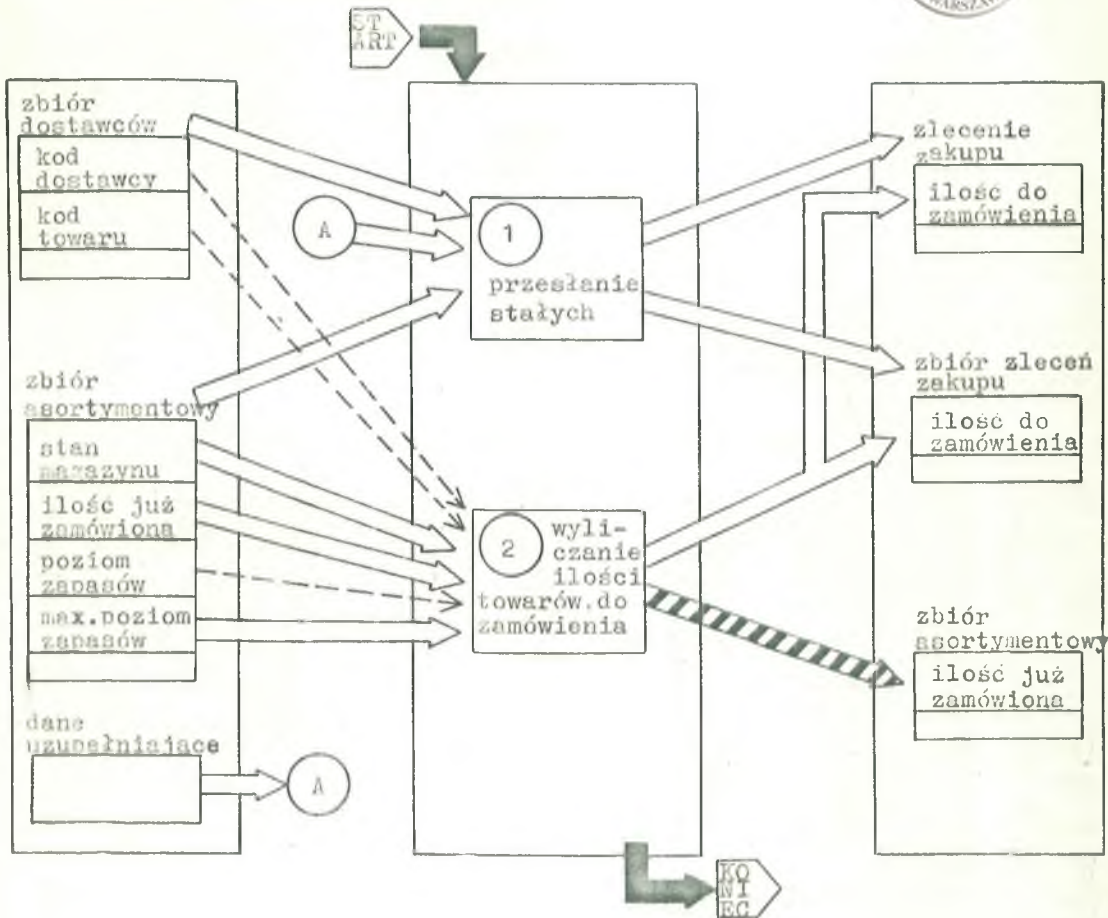
1	ogólnie zakup towarów	str.5
2	przeгляд stanu magazynowego i wystawienie zlecenia zakupu	diagram AA str.6
3	sprawdzanie, czy przysłane towary były zamawiane	diagram AB str.7
4	sprawdzanie, czy zafakturowane towary były przyjęte przez magazyn	diagram AC str.9







Obliczanie wielkości zamówień



Nr	Opis	Uwagi
1	utworzenie zlecenia zakupu	
2	towary do zamówienia, a także ich niezbędne ilości są wyznaczane dla każdego dostawcy dla każdego dostawcy umieszczony jest zapis zlecenia zakupu	TD nr 1 str. 11

Kontrola przyjęcia

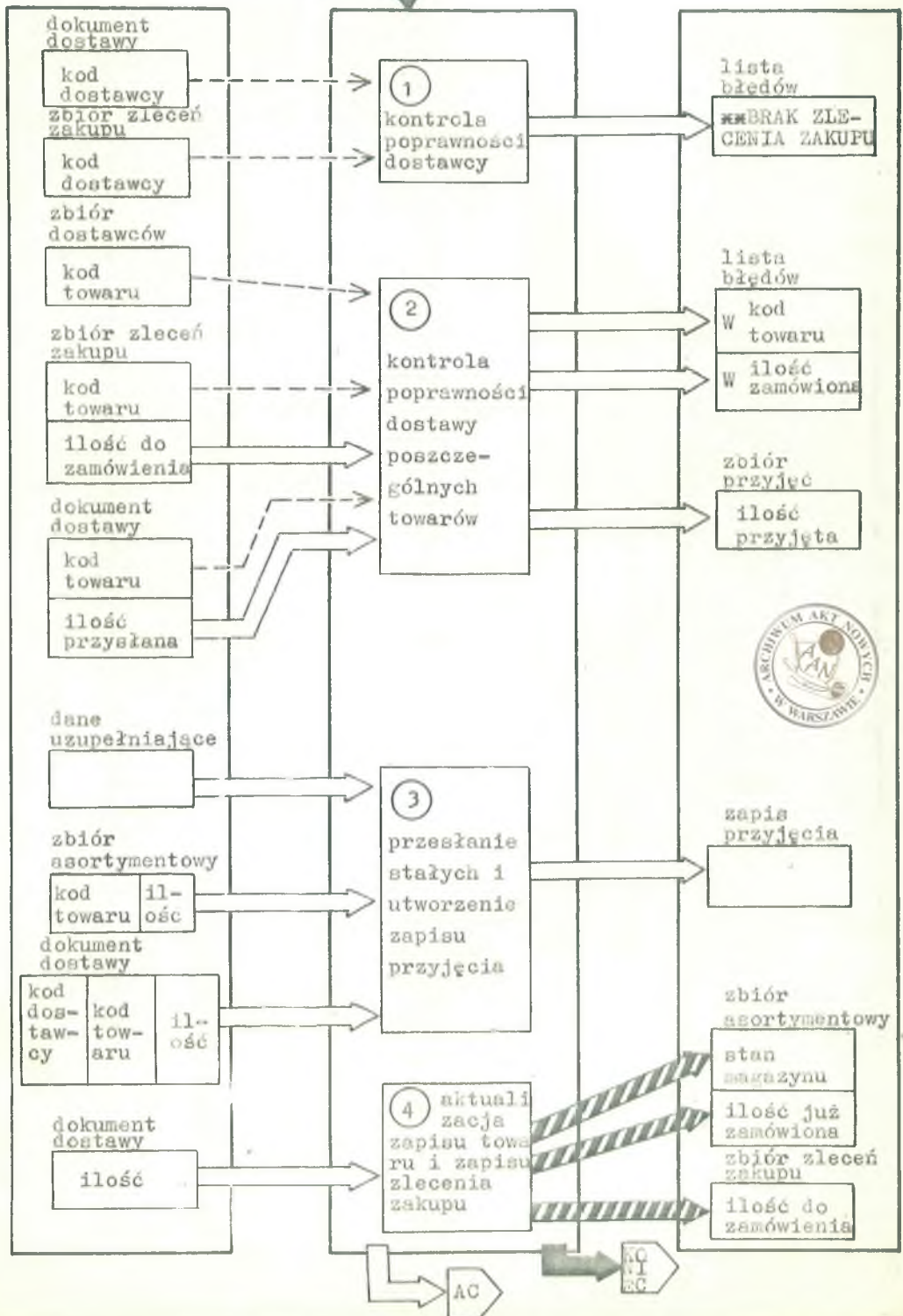




Diagram AB

/cd/

Nr	Opis	Uwagi
1	kontrola poprawności dostawców	TD nr 2 str. 12
2	kontrola poprawności dostawy	TD nr 3 str. 12
3	utworzenie zapisu przyjęcia dla każdego towaru	
4	powiększenie stanu magazynu o ilość przysłaną zmniejszenie ilości już zamówionej zmniejszenie ilości do zamówienia w zapisie zlecenia zakupu	



Diagram AC
Kontrola finansowa

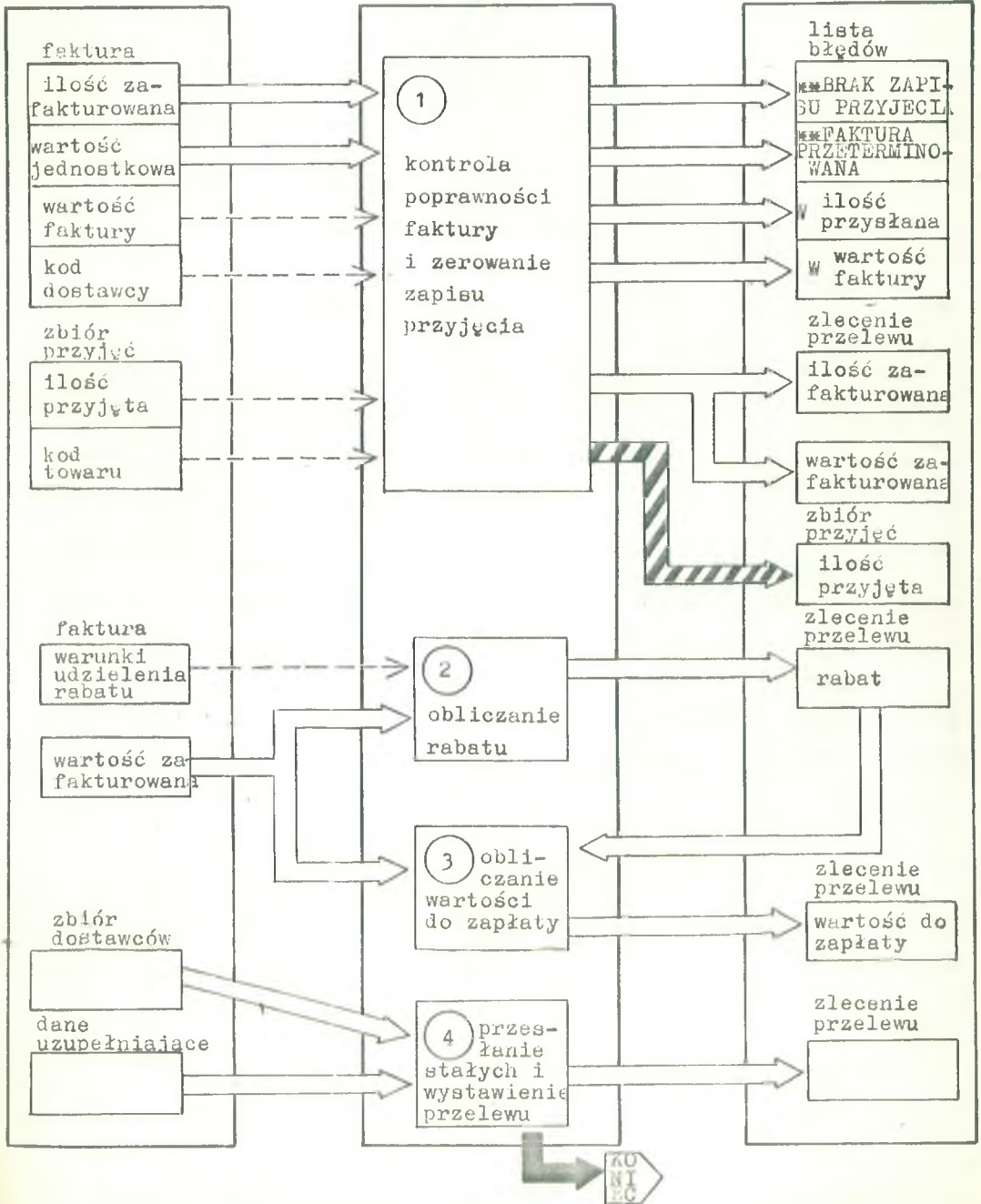




Diagram AC
/cd/

Nr	Opis	Uwagi
1	kontrola poprawności faktury ilości zafakturowanej wartości poszczególnych towarów wartości faktury	TD nr 4 str. 13 TD nr 5 str. 13 TD nr 6 str. 14 TD nr 7 str. 14
2	warunki 0 bez rabatu 1 2% rabatu za opłacenie faktury w ciągu 14 dni 2 1% rabatu za opłacenie faktury w ciągu 30 dni	
3	obniżenie wartości faktury o wartość rabatu	
4	zredukowanie do zera ilości przyjętej w zapisie przyjęcia	



wyliczanie ilości towarów, które należy zamówić	R1	R2
stan magazynu + ilość już zamówiona > maksymalny poziom zapasów	N	T
ilość do zamówienia = = poziom zapasów + 10% + - stan magazynu + - ilość już zamówiona powiększyć ilość już zamówioną o ilość do zamówienia wypisać zlecenie zakupu następna pozycja towaru	X X X X	X

Tablica decyzyjna nr 1

kontrola upoważnień dostawców	R1	R2
czy zapis zlecenia zakupu jest dostępny dla danego kodu dostawcy	T	K
BRAK ZAPISU ZLECENIA ZAKUPU		K
następne zlecenie zakupu		K
kontrola dostarczonego towaru	K	

Tablica decyzyjna nr 2

kontrola poprawności dostawy	R1	R2	R3
czy zapis zlecenia zakupu jest zgodny z kodem dostarczonego towaru	N	T	E L S E
czy ilość dostarczona > ilość zamówiona	-	T	
W kod towaru	K		
W ilość dość ilość przyszła- ną do stanu magazynu zmniejszyć ilość już zamówioną o ilość przysianą		K	K K K K
utworzyć zapis przy- jęcia	K	K	K
następny kod towaru	K	K	K

Tablica decyzyjna nr 3





	R1	R2
kontrola faktury		
czy istnieje zapis przyjęcia	N	T
BRAK ZAPISU PRZYJĘCIA	X	
sprawdzić ilość przysłaną		X
następne faktura	X	X

Tablica decyzyjna nr 4

	R1	R2	R3
kontrola ilości przysłanej			
czy została przetworzone ostatnia pozycja faktury	T	N	X
ilość zafakturowana = ilość przyjęta	-	N	T
W ilość przyjęta		X	
wyliczyć koszt każdego towaru zafakturowanego			X
sprawdzić wartość każdej pozycji na fakturze			X
następna pozycja zafakturowana		X	
sprawdzić wartość faktury	X		

Tablica decyzyjna nr 5

kontrola wartości faktury	I1	R2
wartość całkowita = wartość faktury	H	H
	H	H

Tablica decyzyjna nr 7

kontrola wartości poszczególnych towarów	R1	R2
wartość wyliczona = wartość wg faktury	T	H
/cena jednostkowa * ilość/		
w towar zafakturowany	H	H
sumować wartości pozycji zafakturowanych	H	
wyzerować ilość przyjętą w zapisie przyjęcia	H	

Tablica decyzyjna nr 6



Proj 727
Exp. B