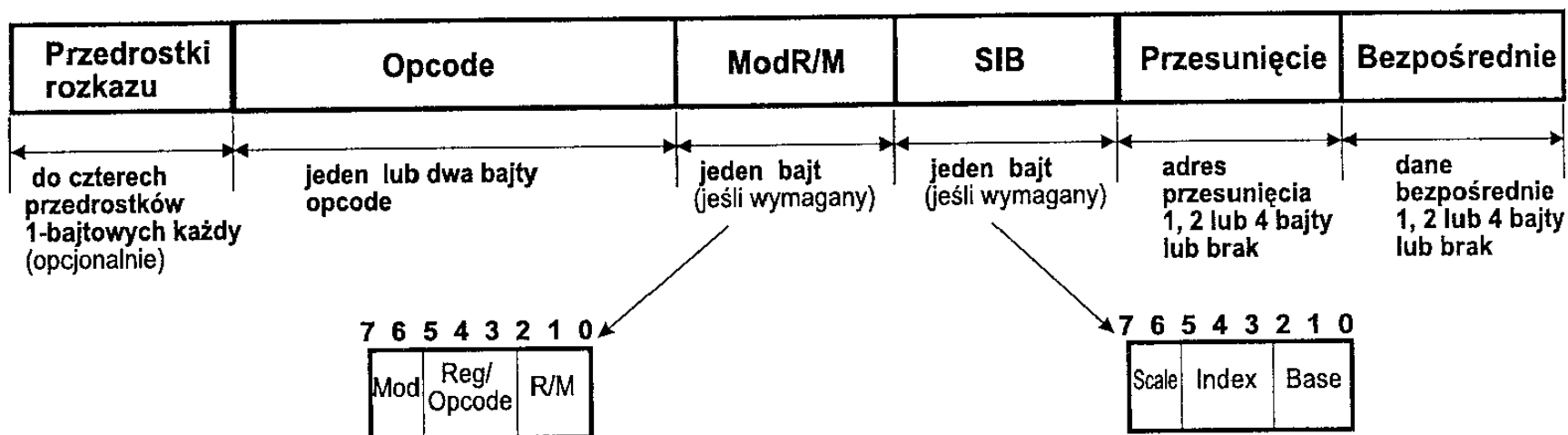


Podstawy przetwarzania informacji w komputerze



Opcode - kod operacyjny definiujący operację wykonywaną przez rozkaz

ModR/M - tryb adresowania rejestru/pamięci

SIB - skalowany indeks bazy

Przesunięcie - adres przesunięcia (przemieszczenia)

Bezpośrednie - dane bezpośrednie

Mod - dwa bity określające czy oba argumenty są rejestrami, czy też jeden z nich jest w pamięci operacyjnej

Reg/Opcode - trzy bity określające rejestr jako argument rozkazu lub będące rozszerzeniem kodu operacji

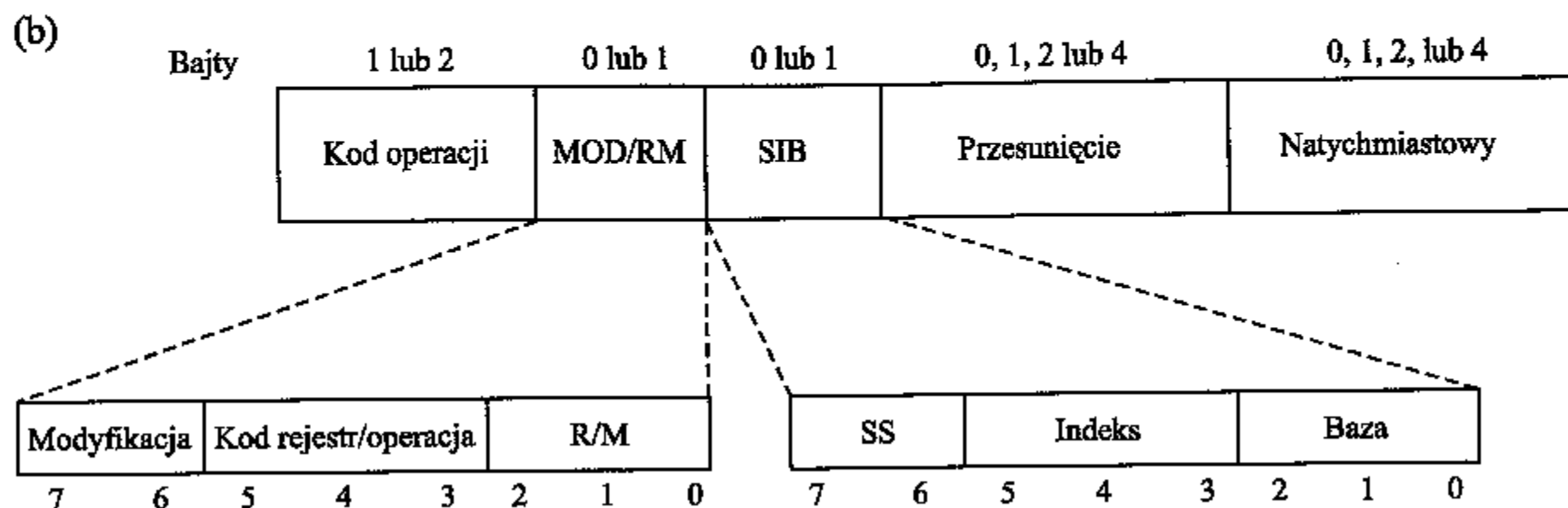
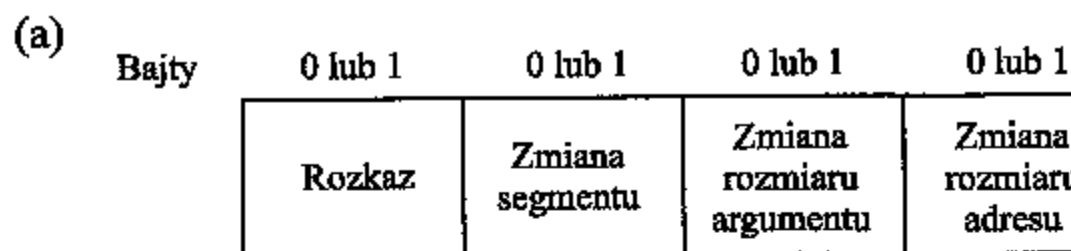
R/M - trzy bity wskazujące rejestr będący argumentem lub rejestr wykorzystywany do obliczenia adresu względem początku segmentu

Scale - dwa bity określające współczynnik skali

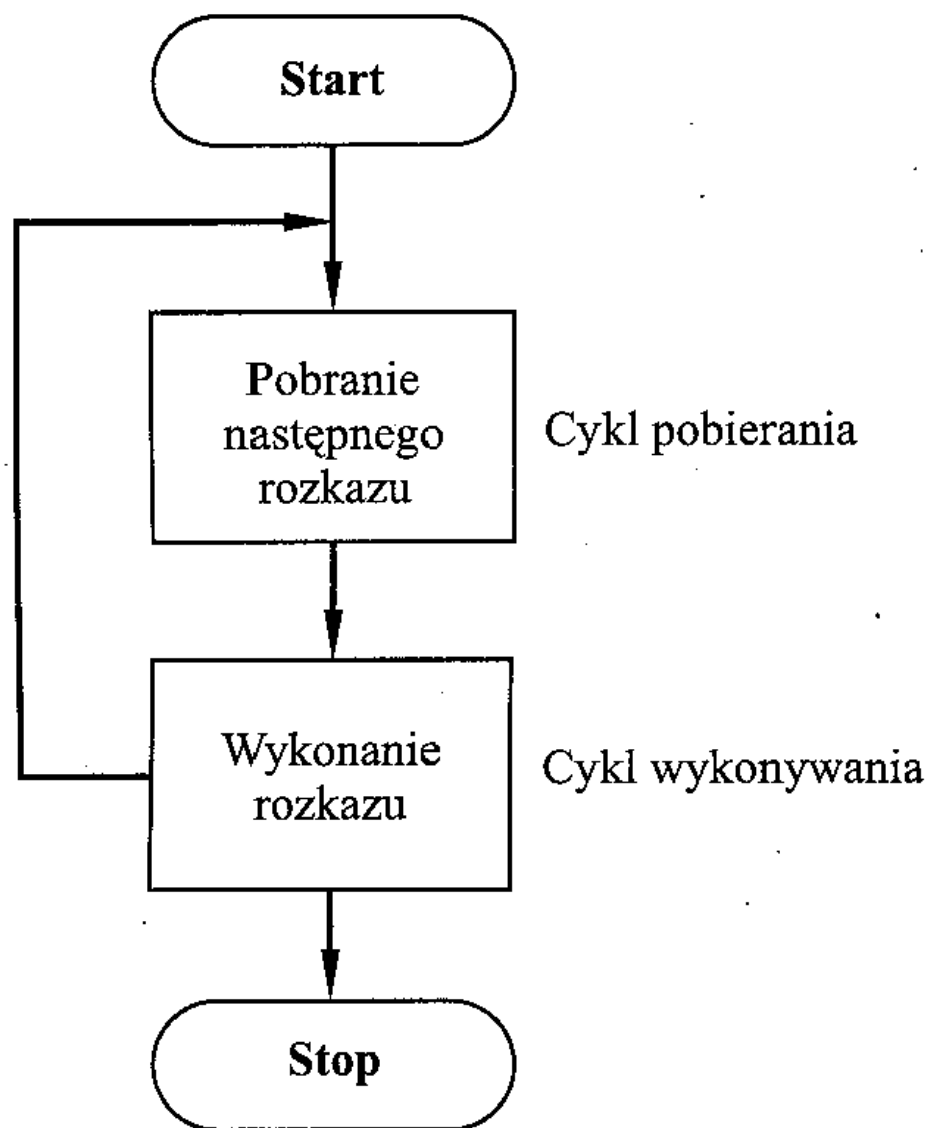
Index - trzy bity określające numer rejestru z rejestru indeksowego

Base - dwa bity określające numer rejestru z rejestru bazowego

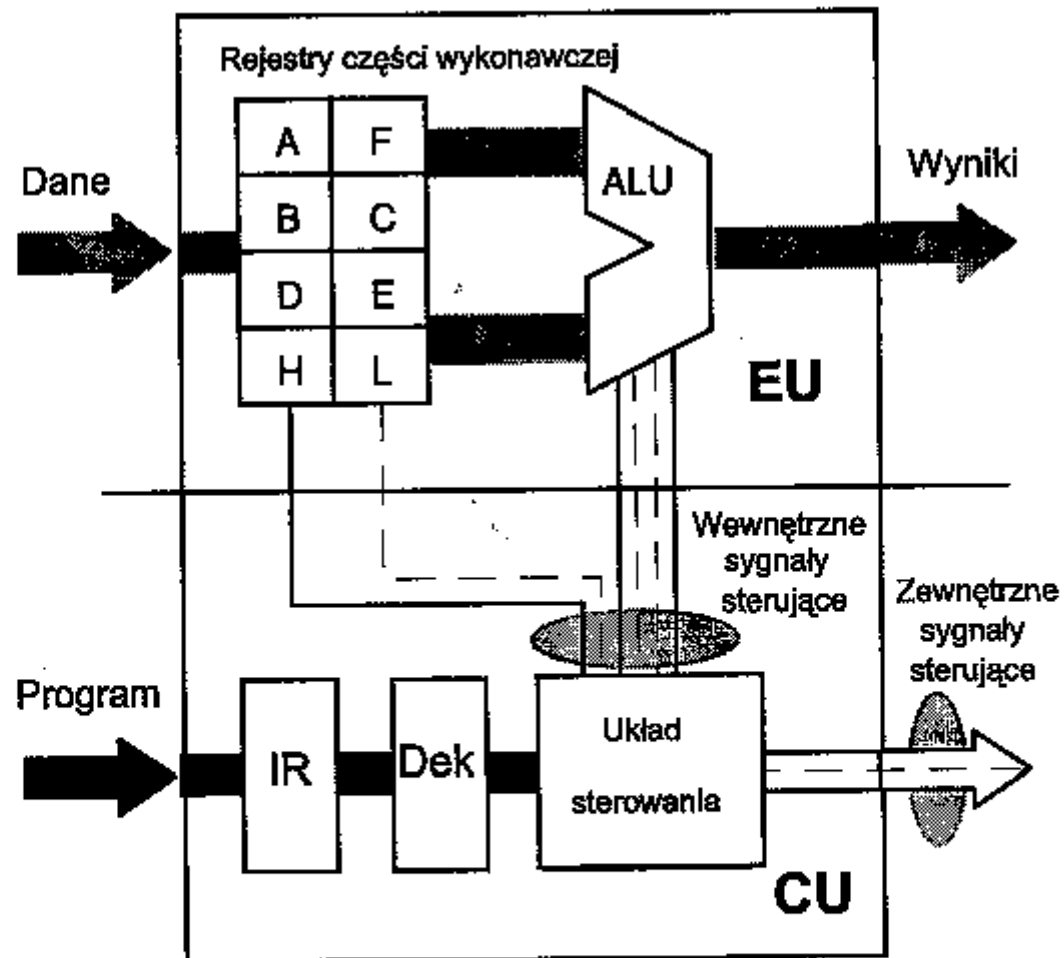
Rysunek 6.1. Format rozkazu



RYSUNEK 10.7. Format rozkazu procesora Pentium: (a) przedrostek; (b) rozkaz



RYSUNEK 3.3. Podstawowy cykl rozkażu



Oznaczenia:

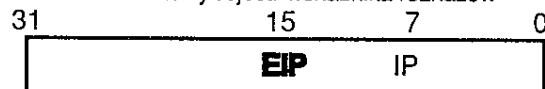
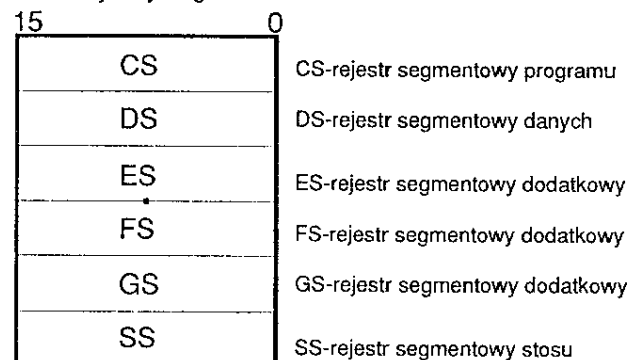
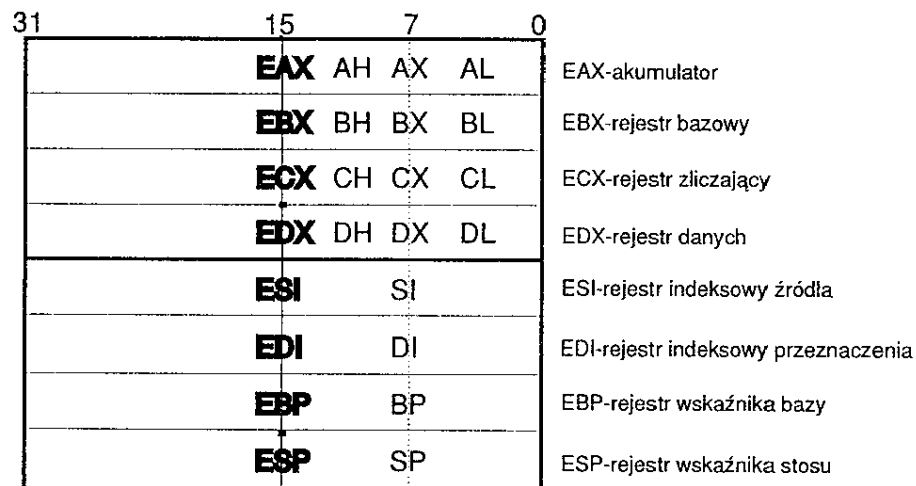
ALU – jednostka arytmetyczno-logiczna

IR – rejestr rozkazów

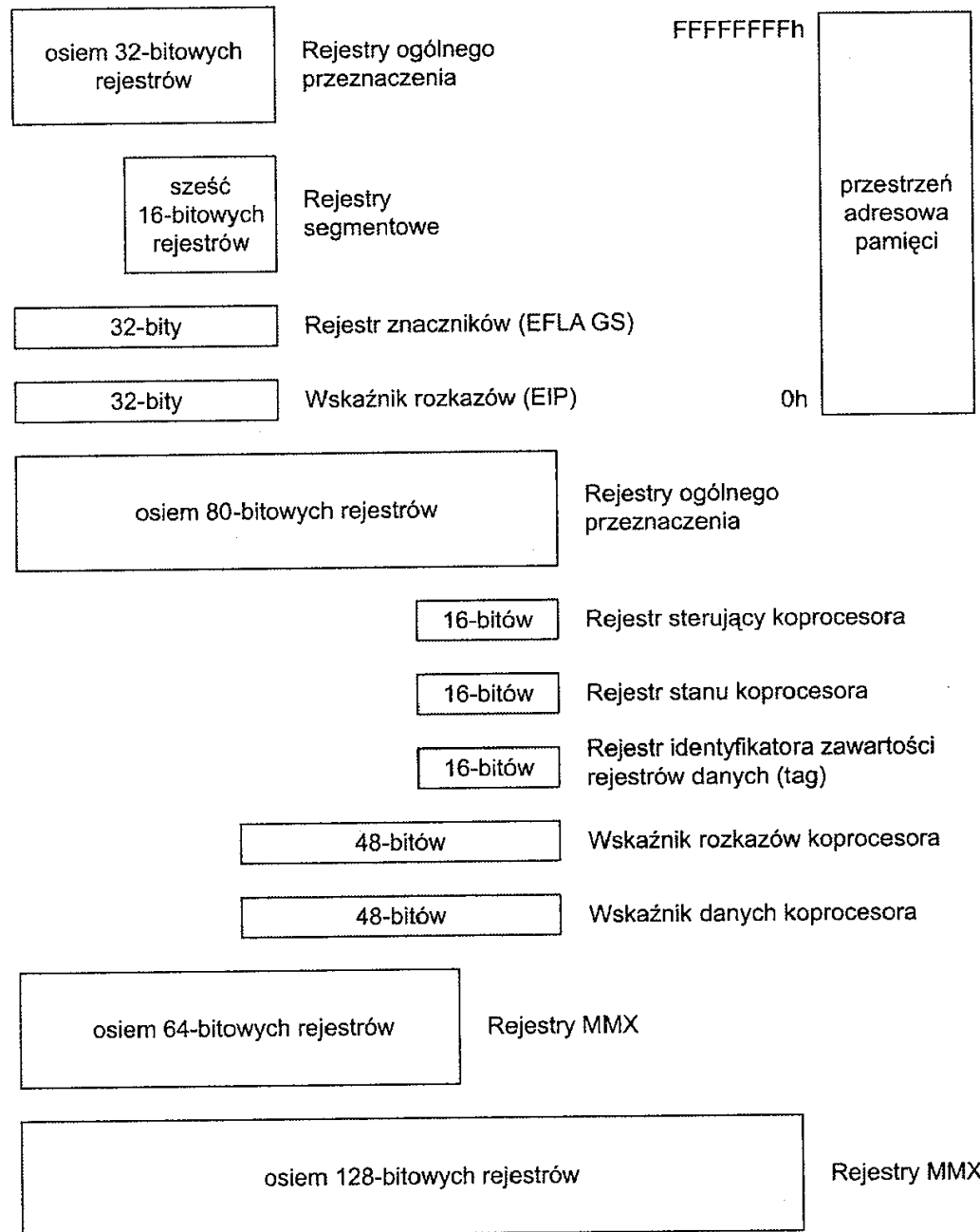
CU – jednostka sterująca

EU – jednostka wykonawcza

Dek – dekodery rozkazów



Rejestry procesora 80386+: powszechnego stosowania, segmentowe i wskaźnika rozkazów



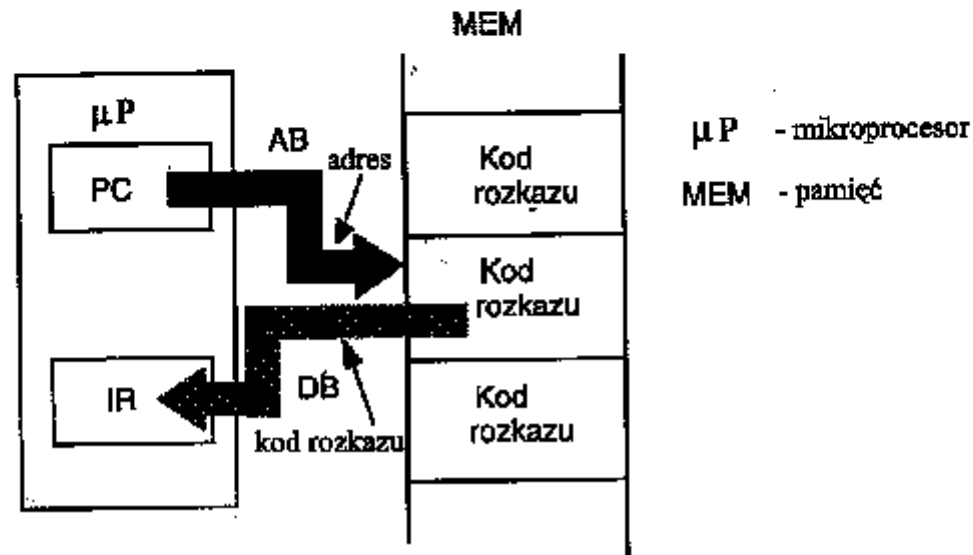
Kolejne etapy realizacji fazy pobrania i fazy wykonania rozkazu można przedstawić następująco:

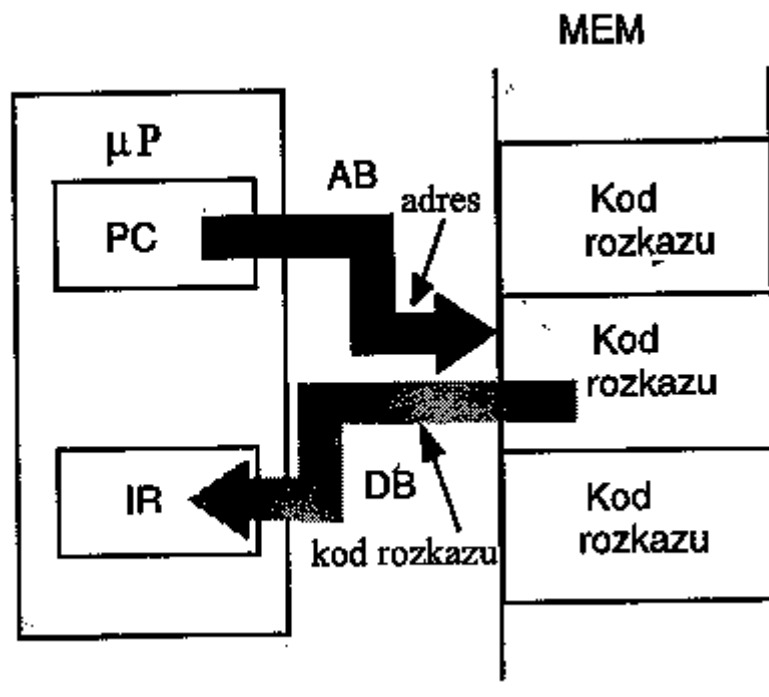
I) Faza pobrania

1. Adresowanie: podanie zawartości licznika rozkazów na magistralę adresową: $AB \leftarrow (PC)$
2. Wczytanie zawartości zaadresowanej komórki pamięci do rejestru rozkazów mikroprocesora: $IR \leftarrow M(PC)$
3. Zwiększenie zawartości licznika rozkazów: $(PC) \leftarrow (PC) + 1$

II) Faza wykonania

4. Zdekodowanie kodu rozkazu i wytworzenie sygnałów sterujących realizujących dany rozkaz.

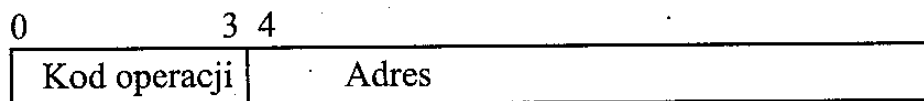




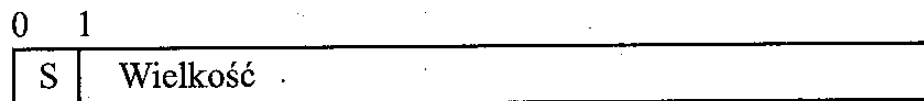
μP - mikroprocesor

MEM - pamięć

(a)



(b)



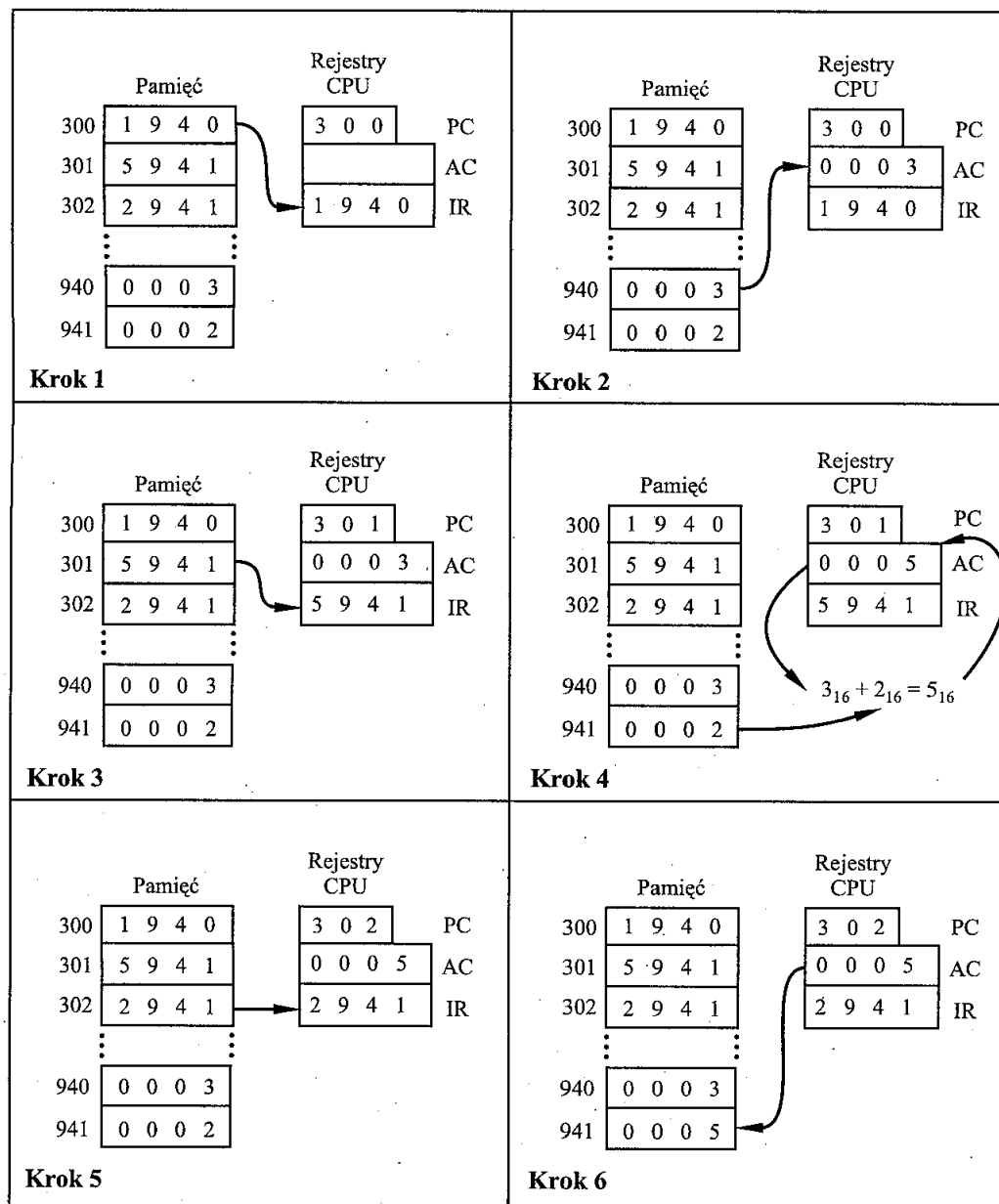
(c)

Licznik rozkazów (PC) = adres rozkazu
Rejestr rozkazów (IR) = wykonywany rozkaz
Akumulator (AC) = tymczasowe przechowywanie

(d)

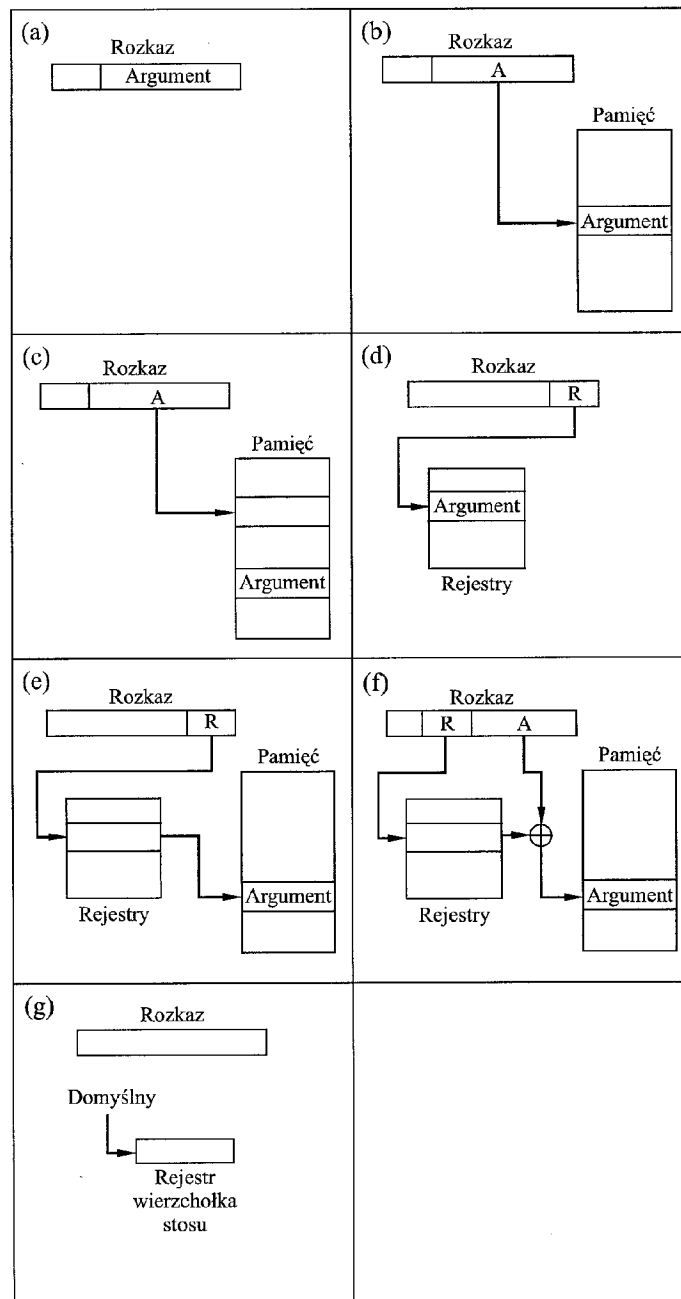
0001 = Ładuj AC z pamięci
0010 = Zapisz AC w pamięci
0101 = Dodaj z pamięci do AC

RYSUNEK 3.4. Własności hipotetycznego komputera: (a) format rozkazu; (b) format liczby całkowitej; (c) wewnętrzne rejestry procesora; (d) częściowa lista kodów operacji



RYSUNEK 3.5. Przykład wykonywania programu

Tryby adresowania pamięci



RYСУNEK 10.1. Tryby adresowania: (a) natychmiastowy; (b) bezpośredni; (c) pośredni; (d) rejestrowy; (e) rejestrowy pośredni; (f) z przesunięciem; (g) stosowy

Programowanie komputera

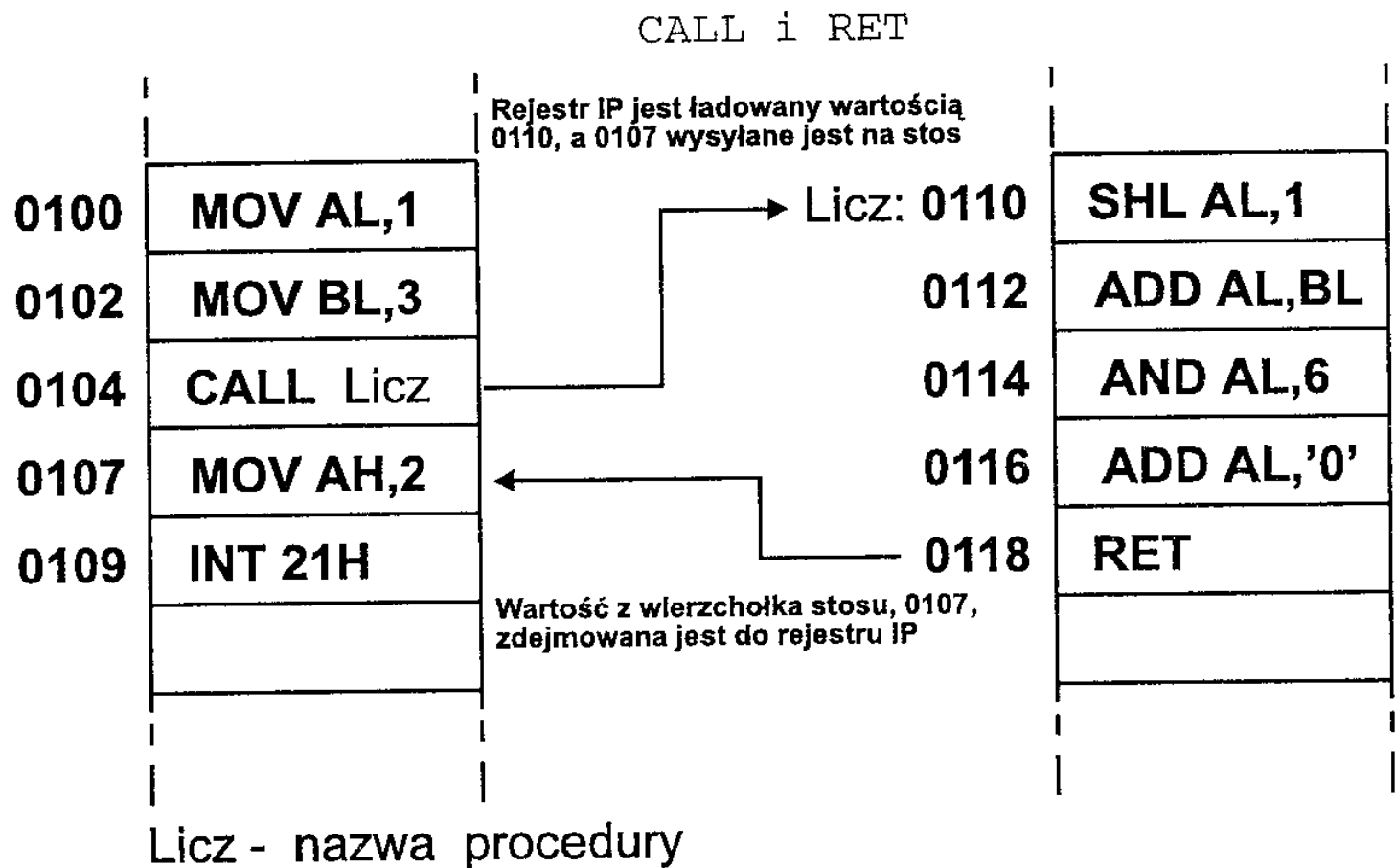
(a)					(b)	
Adres		Zawartość			Adres	Zawartość
101	0010	0010	0000	0001	101	2201
102	0001	0010	0000	0010	102	1202
103	0001	0010	0000	0011	103	1203
104	0011	0010	0000	0100	104	3204
201	0000	0000	0000	0010	201	0002
202	0000	0000	0000	0011	202	0003
203	0000	0000	0000	0100	203	0004
204	0000	0000	0000	0000	204	0000

(c)			(d)		
101	LDA	201	<u>Etykieta</u>	<u>Operacja</u>	<u>Otwarty</u>
102	ADD	202	FORMUL	LDA	I
103	ADD	203		ADD	J
104	STA	204		ADD	K
				STA	N
201	DAT	2			
202	DAT	3	I	DATA	2
203	DAT	4	J	DATA	3
204	DAT	0	K	DATA	4
			N	DATA	0

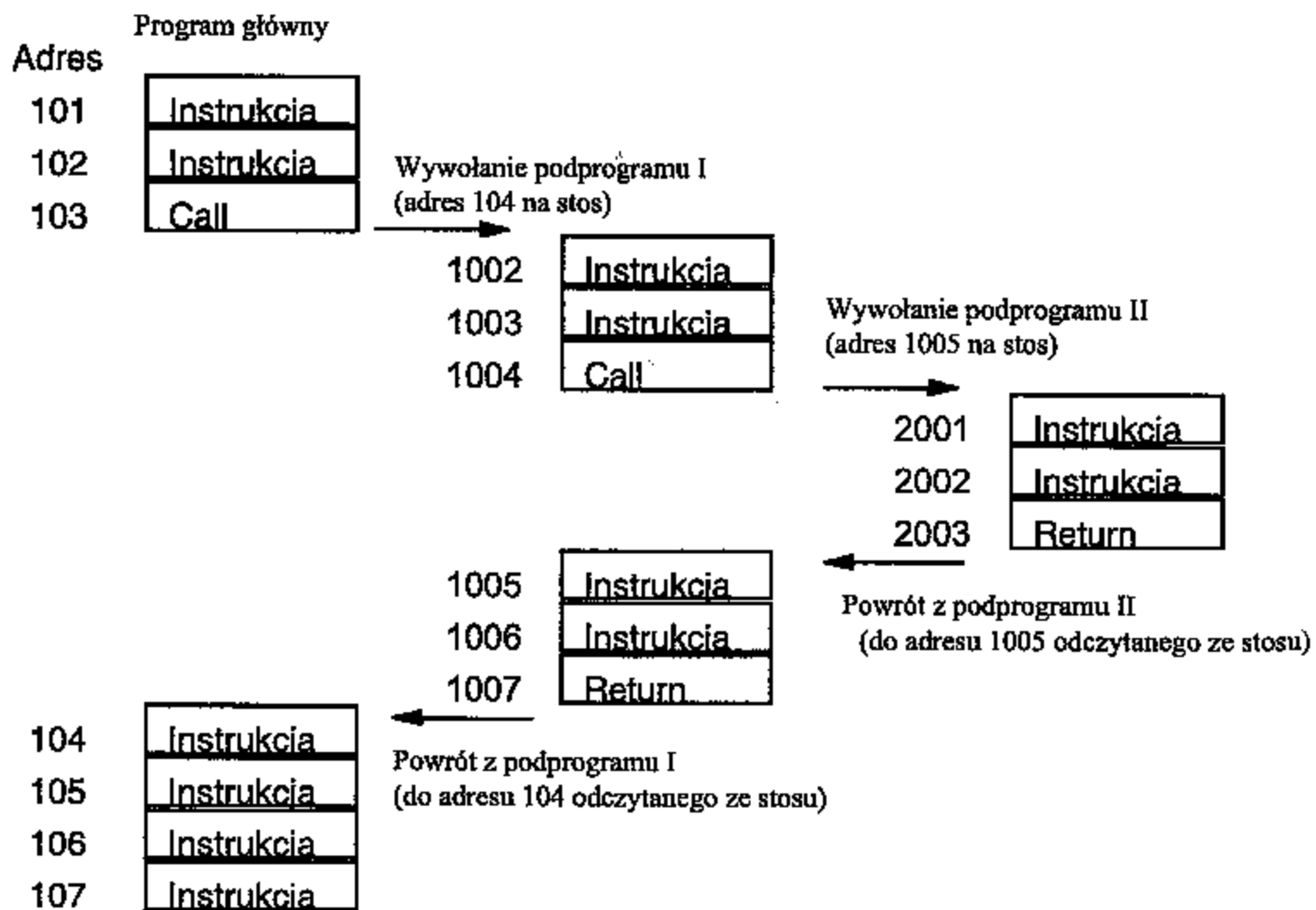
RYSUNEK 9.11. Obliczanie wyrażenia $N = I + J + K$: (a) program binarny; (b) program szesnastkowy; (c) program symboliczny; (d) assembler

Pamięć zarezerwowano dla czterech zmiennych począwszy od lokacji 201. Program składa się z czterech rozkazów:

1. Ładuj zawartość lokacji 201 do AC.
2. Dodaj zawartość lokacji 202 do AC.
3. Dodaj zawartość lokacji 203 do AC.
4. Zapisz zawartość AC w lokacji 204.



Rysunek 15.1.1.1. Rozkazy CALL i RET – wywoływanie procedury w programie



```

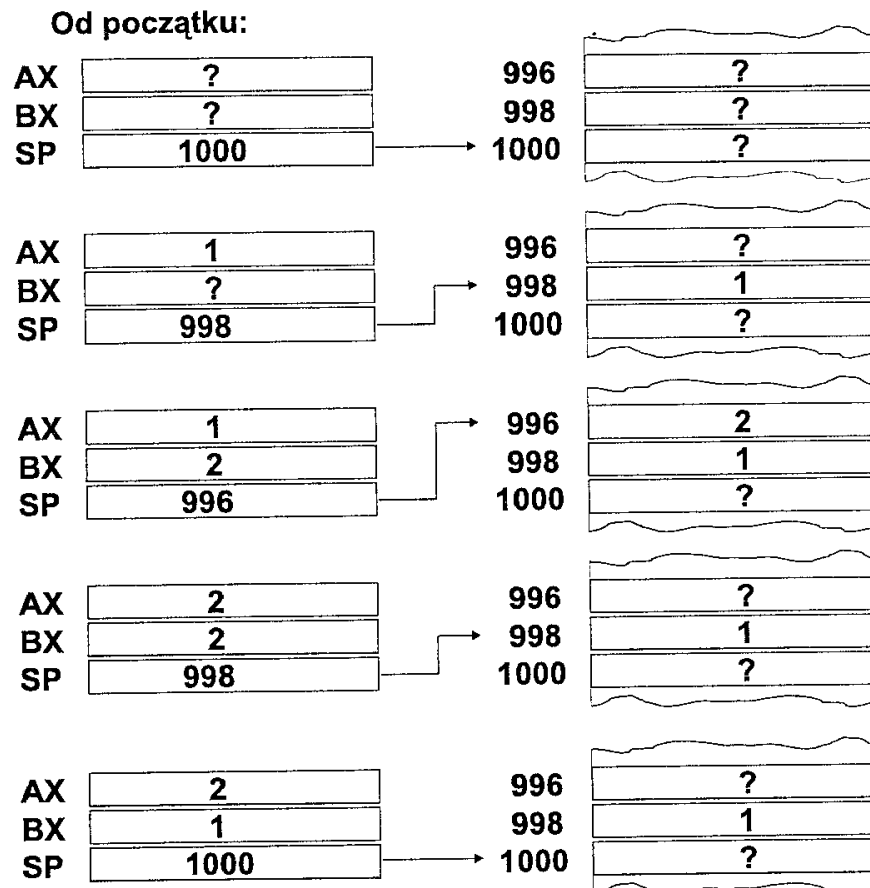
...
MOV AX,1
PUSH AX
MOV BX,2
PUSH BX
POP AX
POP BX
...

```

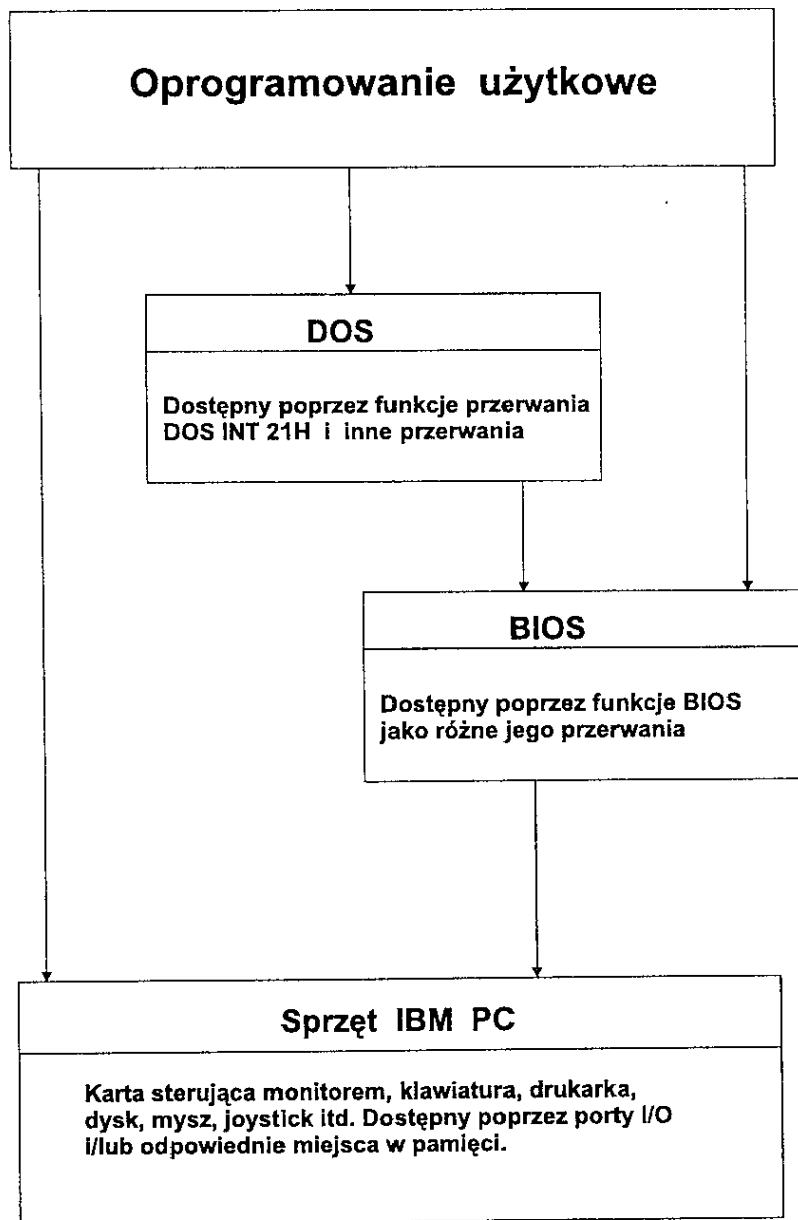
```

;prześlij 1 do rejestru akumulatora AX
;odłóż na stos zawartość rejestru AX
;prześlij do rejestru BX zawartość 2
;odłóż na stos zawartość rejestru BX
;zdejmij ze stosu zawartość rejestru AX
;zdejmij ze stosu zawartość rejestru BX

```

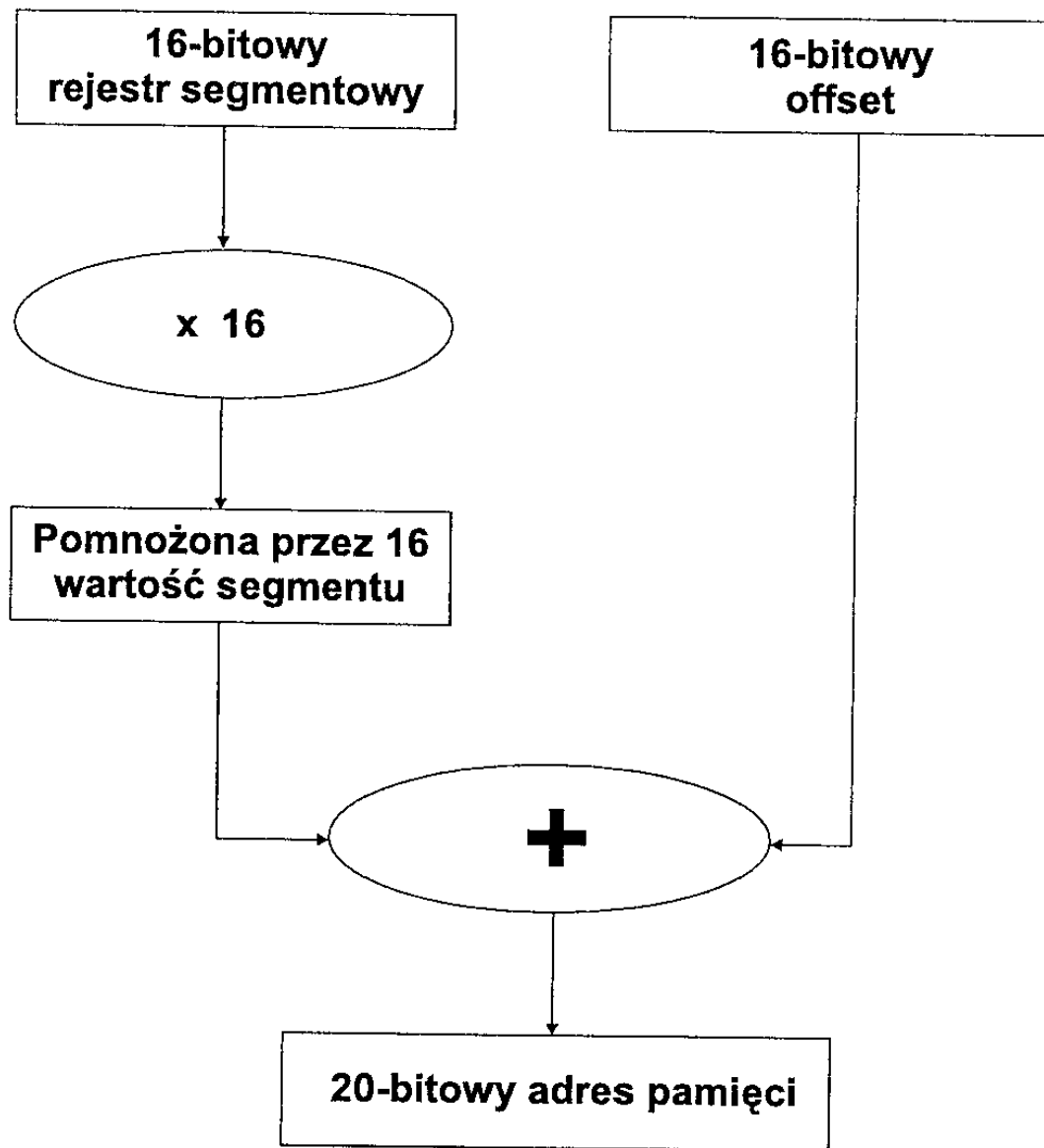


Rysunek 2.2. Rejestry AX, BX, SP i stos

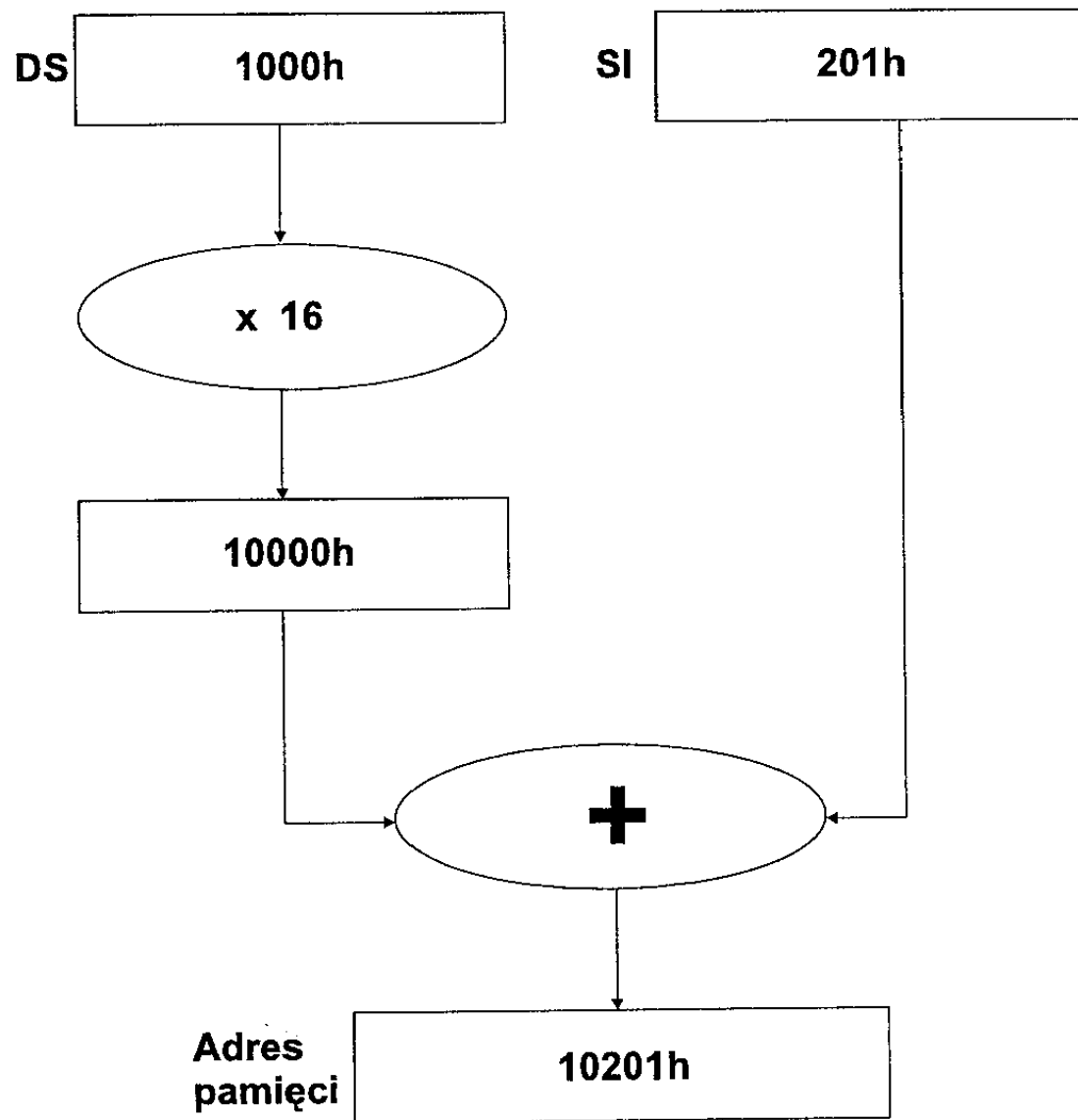


Rysunek 3.1. Oprogramowanie systemowe BIOS i DOS

Obliczanie adresów pamięci



Rysunek 2.3. Obliczanie adresów pamięci – schemat ogólny



Rysunek 2.4. Obliczanie adresów pamięci – przykład