

Wykład 2

Koprocesor numeryczny 8087

Bartłomiej Zieliński, PhD, DSc

8087

Program:

- Ogólna charakterystyka
- Trochę historii – jak to się zaczęło
- Jednostka centralna 8086/8087
- Zasady współpracy 8086 i 8087
- Wewnętrzna struktura 8087
- Rejestry, typy danych
- Obsługa błędów
- Lista rozkazów

8087

- Charakterystyka ogólna
 - 20 bitów adresowych, 8/16 bitów danych
 - Magistrala multipleksowana adres/dane
 - Może współpracować z 8086, 8088, 80186, 80188
 - Automatyczne rozpoznanie procesora
 - Ok. 100× szybsze operacje arytmetyczne
 - Operacje:
 - Arytmetyczne (+, -, *, :, √)
 - Porównania
 - Trygonometryczne
 - Potęgowe
 - Logarytmiczne

8087

- Uwagi ogólne
 - Koprocesor zazwyczaj współdzieli magistralę systemową z μp
 - Podobnie jak sterownik DMA
 - System μ proc. „silnie związany”
 - Niecałkowita zgodność ze standardem IEEE 754
 - Standard pojawił się później, niż 8087

8087

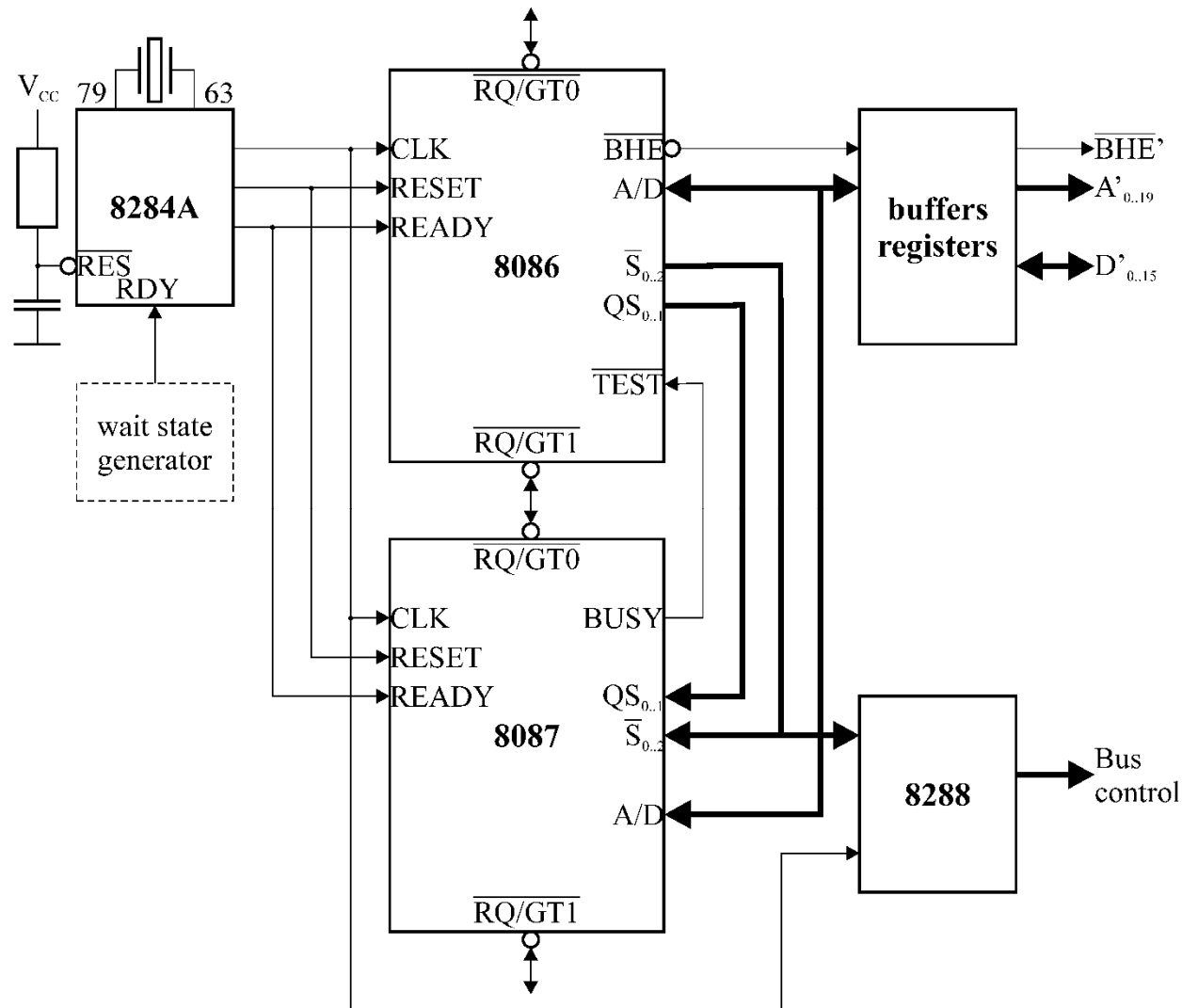
- Trochę historii
 - Intel 8231, 8232 (licencja Am9511, Am9512) (1977-1979)
 - APU (*Arithmetic Processing Unit*) jako urządzenie we-wy
 - 8-b magistrala danych
 - Współpraca z μp z użyciem przerwań lub DMA
 - Może współpracować z dowolnym μp , np. :
 - 8080, 8085, Z-80, Z-8000
 - 6800, 6502 itp.
 - Wykonanie rozkazów równoległe z μp

8087

- Trochę historii (2)
 - Typy danych
 - 16-b i 32-b stałoprzecinkowe
 - 32-b zmiennoprzecinkowe (później także 64-b)
 - Stos danych
 - 8 wartości 16-b, lub
 - 4 wartości 32-b
 - Obsługa jak typowego urządzenia we-wy
 - Odczyt i zapis danych, zapis rozkazu, odczyt stanu
 - Długie czasy wykonania ☹️

8087

- Jednostka centralna 8086/8087



8087

- Zasady współpracy 8086 i 8087
 - Idea:
 - Dla programisty – jak pojedynczy układ scalony, ale:
 - Więcej rozkazów (rozszerzona lista rozkazów)
 - Więcej rejestrów (dodane rejestry koprocatora)
 - Nowe typy danych (włączając typy danych koprocatora)
 - Łatwość programowania
 - Wspólny, jednolity, czytelny kod programu

8087

- Zasady współpracy 8086 i 8087
 - Implementacja
 - 8087 określa stan 8086 na podstawie $S_{0..2}$ and $QS_{0..1}$
 - Śledzi i dekoduje rozkazy synchronicznie z 8086 bez potrzeby interwencji z 8086
 - Pobiera wszystkie rozkazy, ale wykonuje tylko własne
 - Gdy jest rozkaz dla 8087, rozpoczyna pracę niezależną
 - 8086 może kontynuować, gdy nie ma więcej rozkazów 8087
 - W przeciwnym razie 8086 czeka na zakończenie rozkazu
 - Błąd 8087 → przerwanie (zwykle przez 8259A)
 - Przesył danych – przełączanie magistrali za pomocą mechanizmu RQ/GT

8087

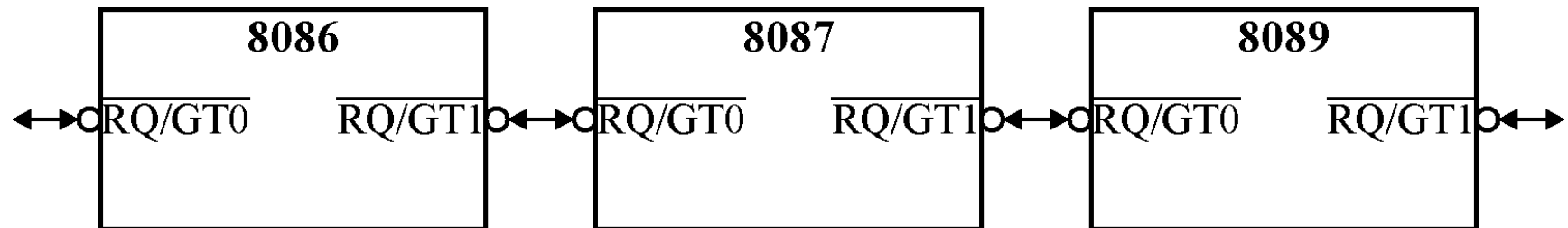
- Zasady współpracy 8086 i 8087
 - Jeśli rozkaz FPU ma argument w pamięci
 - Odczyt
 - 8086 oblicza adres
 - „pusty odczyt” (*dummy read*)
 - » Dane pomijane przez 8086, czytane przez 8087
 - » Adres zapamiętany w 8087
 - Rozmiar danych > 16b → kolejne cykle (RQ/GT)
 - Zapis
 - 8086 oblicza adres
 - „pusty odczyt” (*dummy read*)
 - » Dane pomijane przez 8086 oraz 8087
 - » Adres zapamiętany w 8087
 - Zapis danych podczas kolejnych cykli (RQ/GT)

8087

- Zasady współpracy 8086 i 8087
 - FPU musi znać rozmiar magistrali (8/16 bitów)
 - Jak rozróżnić μ p 8086 i 8088?
 - Pin 34 bezpośrednio po włączeniu zasilania
 - 8086: $\overline{\text{BHE}}=0$
 - 8088: $\overline{\text{SSO}}=1$
 - 8086: dostęp 16-b, jeśli możliwy
 - 8088: zawsze dostęp 8-b
 - Szerokość magistrali nie wpływa na dokładność i prędkość obliczeń

8087

- Zasady współpracy 8086 i 8087
 - Podłączenie wielu koprocessorów
 - Np. 8089 – koprocessor we-wy
 - Przekazanie sterowania magistralą z opóźnieniem $1 T_{clk}$



8087

- Zasady współpracy 8086 i 8087

- Synchronizacja

- Rozkaz WAIT między dwoma rozkazami FPU

- Bezpośrednio po rozkazie FPU

- » Bez pracy równoległej

- » Bez konfliktu danych

- Bezpośrednio przed następnym rozkazem FPU

- » Możliwość pracy równoległej

- » Programista musi zapewnić brak konfliktu danych

- 8086 nie zmienia danych 8087 w pamięci

- W μ p 80286+ WAIT jest wykonywany automatycznie

- » Rozkaz WAIT dekodowany, ale zbędny

FPU command

wait

FPU command

FPU command

wait

CPU commands

FPU command

FPU command

CPU commands

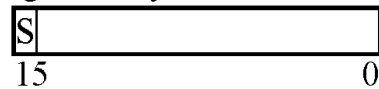
wait

FPU command

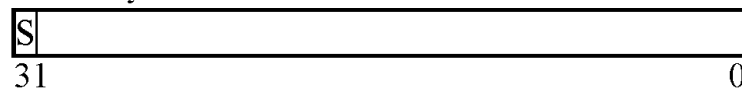
8087

- Typy danych 8087
 - Typy danych

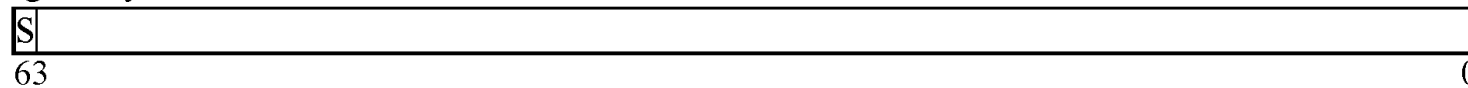
Integer binary word



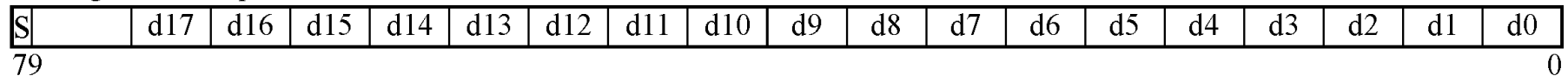
Short binary number



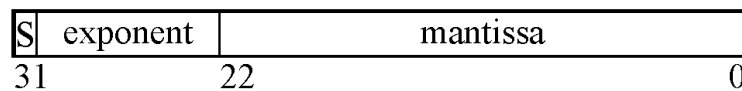
Long binary number



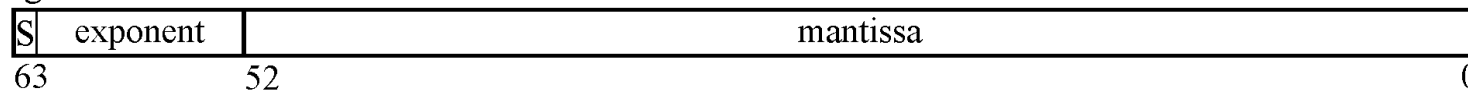
BCD integer number packed



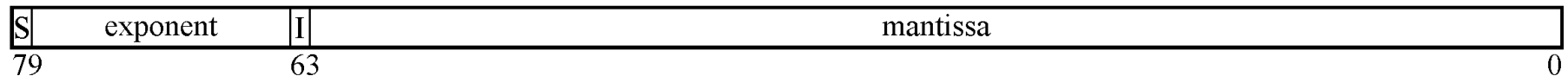
Short real number



Long real number



Extended real number



8087

- Typy danych 8087
 - Zakresy wartości danych

Typ danych	Bity	Cyfry znaczące	Min	Max
Integer word	16	4	-32768	32767
Short integer	32	9	-2×10^9	2×10^9
Long integer	64	18	-9×10^{18}	9×10^{18}
Decimal	80	18	-99...99	99...99
Short real	32	6-7	$\pm 1.2 \times 10^{-38}$	$\pm 3.37 \times 10^{38}$
Long real	64	15-16	$\pm 2.3 \times 10^{-308}$	$\pm 1.67 \times 10^{308}$
Extended real	80	19	$\pm 3.4 \times 10^{-4392}$	$\pm 1.2 \times 10^{4392}$

8087

- Typy danych 8087
 - Typy całkowite
 - Zapis dopełnieniowy (*radix complement*)
 - Liczby całkowite 16-b jak w 8086
 - Typ BCD
 - Spakowany – 2 cyfry dziesiętne na bajt
 - Zapis modułowy (*sign-magnitude*)

8087

- Typy danych 8087
 - Typy rzeczywiste
 - $X = (-1)^S \times M \times 2^E$
 - Liczby znormalizowane $\rightarrow$ cz. całk. $M = 0$
 - Większa dokładność obliczeń
 - Typy krótki i długi tylko w pamięci
 - Automatyczna konwersja do typu rozszerzonego podczas ładowania do 8087

8087

- Typy danych 8087
 - Wartości specjalne
 - Wartości denormalne
 - Wykładnik zbyt mały, by przedstawić w danym formacie
 - » Np. wykładnik=-130 zbyt mały dla formatu krótkiego
 - Niedomiar jest maskowany
 - Można kontynuować obliczenia (z mniejszą dokładnością)
 - Gdy nie można pominąć błędu obliczeń, jest on zgłaszany
 - Denormalizacja może pominąć znaczące 1 na najmniej znaczących pozycjach
 - » Liczba $\rightarrow 0$
 - Uzupełnienie wykładnika wynosi 0
 - Całkowita część mantysy wynosi 0
 - » Widoczne tylko w formacie rozszerzonym

8087

- Typy danych 8087
 - Wartości specjalne
 - Wartości pseudonormalne
 - Tylko w formacie rozszerzonym
 - Natychmiastowy wynik zamaskowanej reakcji na niedomiar
 - » zdenormalizowane → pseudonormalne podczas zapisu do rejestrów
 - Wykładnik znormalizowany, >0
 - Część całkowita mantysy = 0
 - Oznaczone jako wartość prawidłowa w słowie etykiet
 - Nie zatrzymuje obliczeń
 - Można kontynuować obliczenia mimo niedomiaru

8087

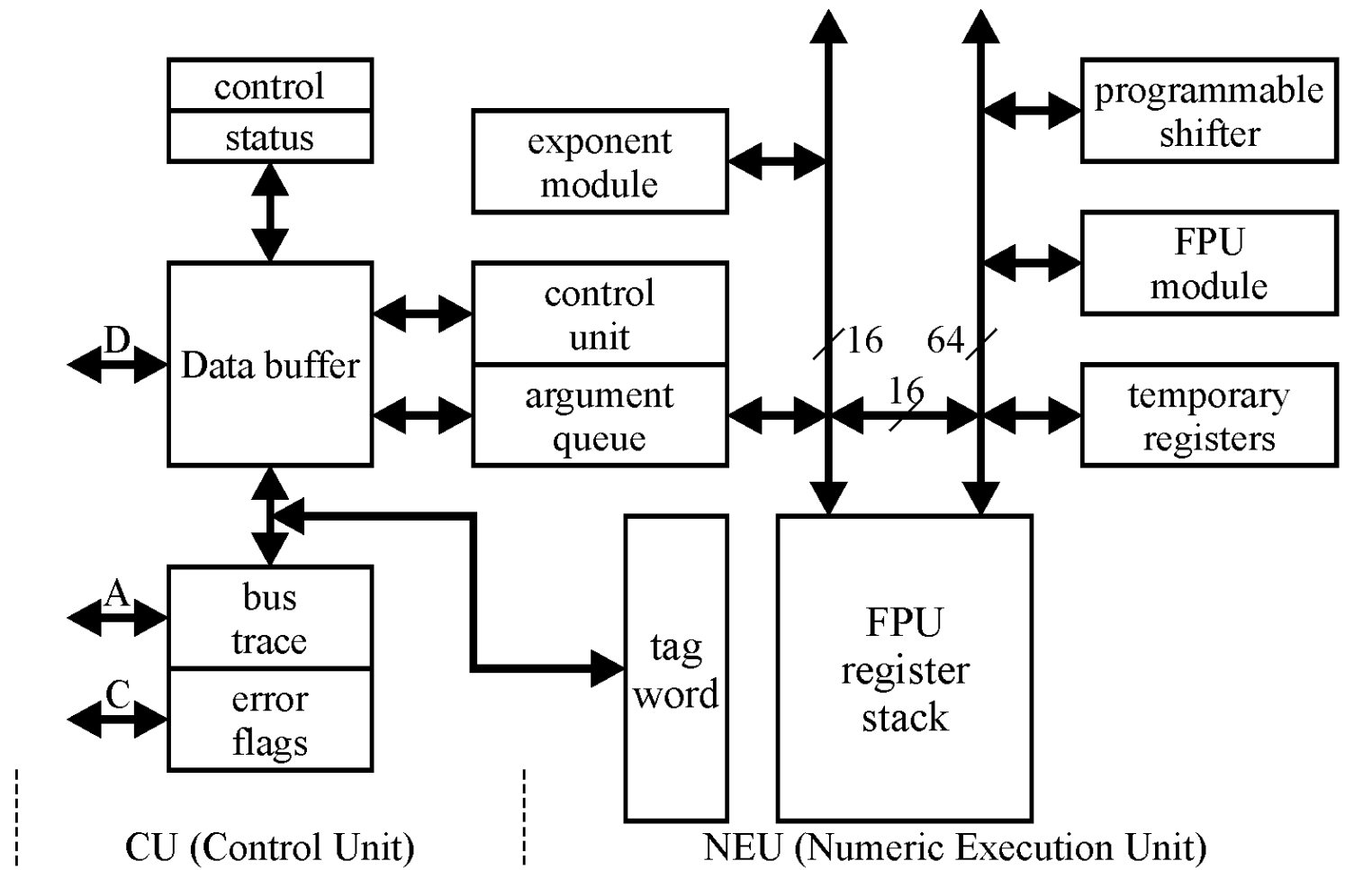
- Typy danych 8087
 - Wartości specjalne
 - Zera i pseudozera
 - W niektórych typach zero może być dodatnie lub ujemne
 - » Ale to ciągle zero i zachowuje się jak zero
 - » Rozkaz FXAM może sprawdzić znak zera
 - Typ rozszerzony → pseudozero
 - » Wartość pseudonormalna, mantysa=0, wykładnik≠0

8087

- Typy danych 8087
 - Wartości specjalne
 - Nieskończoność
 - W słowie etykiet oznaczone jako nieskończoność
 - Wynik zamaskowanej reakcji na nadmiar lub dzielenie przez 0
 - „Nie-liczba” (NAN, *not a numer*)
 - Tylko w typach rzeczywistych
 - W słowie etykiet oznaczone jako nie-liczba
 - Wynik zamaskowanej reakcji na operację niedozwoloną

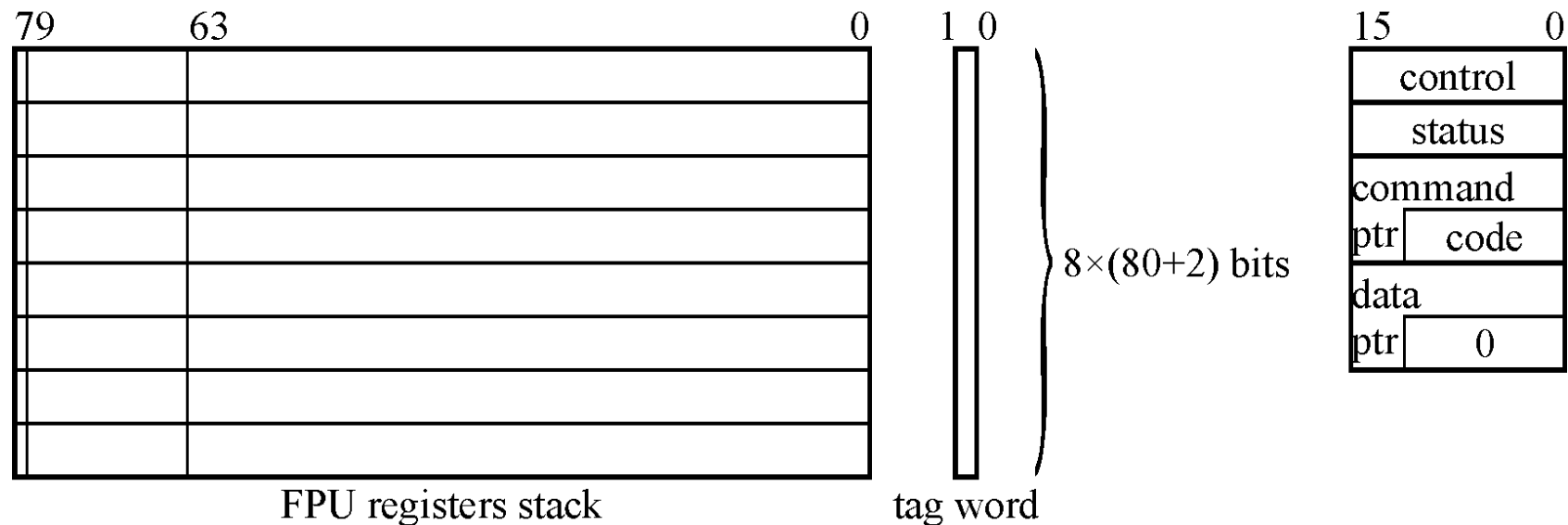
8087

- Struktura 8087



8087

- Rejestry 8087



- Etykiety

- Wartość prawidłowa (znormalizowana lub pseudonormalna)
- Zero (prawdziwe zero)
- Wartość specjalna (NaN, nieskończoność, denormalna)
- Puste

8087

- Rejestry 8087
 - Rejestr sterujący
 - Maski błędów:
 - IM – niedozwolona operacja
 - DM – argument denormalny
 - ZM – dzielenie przez 0
 - OM – nadmiar
 - UM – niedomiar
 - PM – dokładność
 - M – maska przerwania (1 = zamaskowane)
 - PC – dokładność obliczeń (24/53/64 bity mantysy)
 - RC – zaokrąglanie (do najbliższej, w dół, w górę, obcięcie)
 - IC – nieskończoność (ze znakiem/bez znaku)

8087

- Rejestry 8087
 - Rejestr stanu:
 - Znaczniki błędów:
 - IE – niedozwolona operacja
 - DE – argument denormalny
 - ZE – dzielenie przez 0
 - OE – nadmiar
 - UE – niedomiar
 - PE – dokładność
 - IR – zgłoszenie przerwania
 - TOP – szczyt stosu
 - B – zajętość
 - C_{0..3} – kody warunkowe

8087

- Rejestry 8087
 - Kody warunkowe
 - Porównanie (FCOM), test (FTST)

C _{3..0}	FCOM	FTST
00-0	ST > arg. źródłowy	ST > 0
00-1	ST < arg. źródłowy	ST < 0
10-0	ST = arg. źródłowy	ST = ±0
11-1	ST=NAN lub ∞ (nie można porównać)	

8087

- Rejestry 8087
 - Kody warunkowe
 - Sprawdzenie (*examination*, FXAM)
 - ST>0, normalna
 - ST<0, normalna
 - ST>0, pseudonormalna
 - ST<0, pseudonormalna
 - ST>0, zdenormalizowana
 - ST<0, zdenormalizowana
 - ST=+NAN
 - ST=-NAN
 - ST=+∞
 - ST=-∞
 - ST=+0
 - ST=-0
 - ST jest pusty

8087

- Rejestry 8087

- Kody warunkowe – skoki warunkowe

- przykład: porównanie zmiennoprzecinkowych A i B

```
FILD A          ; load A to top of the stack
FCOMP B         ; compare with B
FSTSW st        ; store state
FWAIT           ; wait until ready
MOV AH, st+1    ; copy state MSB to AH
SAHF            ; save state in flag reg.
; bity warunków są teraz w znacznikach µp
```

8087

- Błędy 8087
 - Niedozwolona operacja
 - Zapis do niepustego rejestru
 - Odczyt pustego rejestru
 - Użycie argumentu NAN
 - Operacja na argumencie, która może dać nieokreślony wynik (np. `sqrt (negative)`)
 - Nadmiar, niedomiar
 - Wynik zbyt duży/zbyt mały dla określonego formatu
 - Division by zero
 - Niezerowy argument dzielony przez 0
 - Argument zdenormalizowany

8087

- Rozkazy 8087

- Przykładowe formaty

- MF – format liczby

- Krótki rzeczywisty
 - Krótki całkowity
 - Długi rzeczywisty
 - Słowo całkowite

- MOD – tryb przemieszczenia (jak w 8086)

- R/M – tryb adresowania (jak w 8086)

Data transfer to/from memory

<i>Escape</i>	MF		MOD		R/M
Disp			Disp		

Data transfer between registers

<i>Escape</i>			ST(i)
---------------	--	--	-------

Arithmetical operations

<i>Escape</i>		
---------------	--	--

Communication with 8086/88 AX register

1	0	0	1	1	0	1	1
<i>Escape</i>							

Communication with 8086/88 AX memory

1	0	0	1	1	0	1	1
<i>Escape</i>			MOD		R/M		
Disp				Disp			

8087

- Rozkazy 8087

- Przesył danych

- FLD – ładowanie wartości rzeczywistej
 - FST/FSTP – zapis wartości rzeczywistej (+ z.z.s.)
 - FSTP – FST + zdjęcie ze stosu
 - FILD – ładowanie wartości całkowitej
 - FIST/FISTP – zapis wartości całkowitej (+ z.z.s.)
 - FBLD – ładowanie wartości BCD
 - FBSTP – zapis wartości BCD + z.z.s.
 - (z.z.s. = zdjęcie ze stosu (ang. z.z.s.))

8087

- Rozkazy 8087

- Atrybetyczne (+, -, ×, :)

- FADD/FADDP – suma 2 liczb rzeczywistych (+ z.z.s.)
 - FIADD – suma 2 liczb całkowitych
 - FSUB/FSUBP – różnica liczb rzeczywistych (+z.z.s.)
 - FISUB/FISUBR – różnica liczb całkowitych (odwrotna)
 - FSUBR/FSUBRP – odwrotna różnica liczb rzecz. (+ z.z.s.)
 - FMUL/FMULP – iloczyn liczb rzeczywistych (+ z.z.s.)
 - FIMUL – iloczyn liczb całkowitych
 - FDIV/FDIVP – iloraz liczb rzeczywistych (+ z.z.s.)
 - FIDIV/FIDIVR – iloraz liczb całkowitych (odwrotny)
 - FDIVR/FDIVRP – odwrotny iloraz liczb rzecz. (+ z.z.s.)

8087

- Rozkazy 8087
 - Arytmetyczne (c.d.)
 - FSQRT – pierwiastek kwadratowy
 - FSCALE – skalowanie przez 2 (mul/div przez potęgę 2)
 - FPREM – reszta z dzielenia
 - FRNDINT – zaokrąglenie rzeczywistej do całkowitej
 - FXTRACT – ekstrakcja wykładnika i mantysy
 - FABS – wartość bezwzględna
 - FCHS – zmiana znaku

8087

- Rozkazy 8087

- Porównania

- FCOM/FCOMP/FCOMPP – porównanie (+ z.z.s. (+ z.z.s.))
 - FICOM/FICOMP – porównanie całkowite (+ z.z.s.)
 - FTST – porównanie z zerem
 - FXAM – sprawdzenie liczby na szczycie stosu

8087

- Rozkazy 8087

- Funkcje

- FPTAN – tangens
 - FPATAN – arcus tangens
 - F2XM1 – obliczenie $2^x - 1$
 - FYL2X – obliczenie $Y \times \log_2 X$
 - FYL2XP1 – obliczenie $Y \times \log_2(X + 1)$

8087

- Rozkazy 8087
 - Ładowanie stałych
 - FLDZ – zero (+0.0)
 - FLD1 – 1 (+1.0)
 - FLDPI – π
 - FLDL2T – $\log_2 10$
 - FLDL2E – $\log_2 e$
 - FLDLG2 – $\log_{10} 2$
 - FLDLN2 – $\log_e 2$

8087

- Rozkazy 8087

- Sterowanie (1)

- FINIT/FNINIT – inicjalizacja (zerowanie programowe)
 - FDISI/FNDISI – wyłączenie przerwań
 - FENI/FNENI – włączenie przerwań
 - FLDCW – załadowanie słowa sterującego z pamięci
 - FSTCW/FNSTCW – zapis słowa sterującego do pamięci
 - FSTSW/FNSTSW – zapis słowa stanu do pamięci
 - FCLEX/FNCLEX – zerowanie znaczników błędów

8087

- Rozkazy 8087

- Sterowanie (2)

- FSTENV/FNSTENV – zapis skróconego stanu do pamięci
 - FLDENV – odczyt skróconego stanu z pamięci
 - FSAVE/FNSAVE – zapis pełnego stanu do pamięci
 - FRSTOR – odczyt pełnego stanu z pamięci
 - FINCSTP – zwiększenie wskaźnika stosu
 - FDECSTP – zmniejszenie wskaźnika stosu
 - FFREE – „opróżnienie” rejestru (etykieta = Pusty)
 - FNOP – nic nie rób
 - FWAIT