

# ***SimPPC***

Symulator mikroprocesora PowerPC™ 601

Podręcznik użytkownika





**Wydanie I, styczeń 2014**

Podręcznik użytkownika aplikacji SimPPC w wersji 1.1

**PowerPC jest zastrzeżonym znakiem towarowym firmy International Business Machines Corp.**

Wszystkie znaki towarowe użyte w niniejszym opracowaniu są zastrzeżonymi znakami towarowymi ich właścicieli i zostały one użyte tylko w celach informacyjnych oraz edukacyjnych.

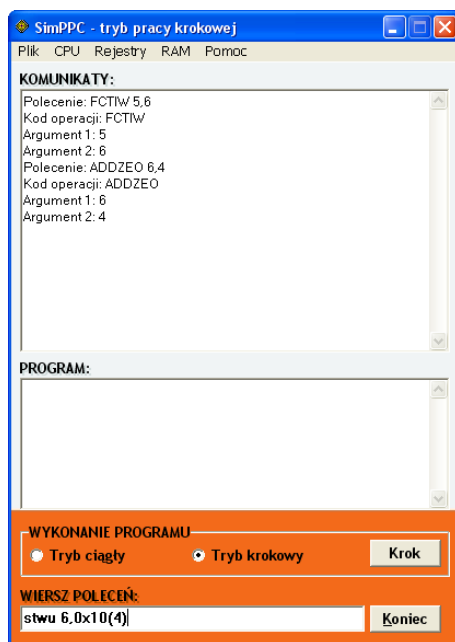
## Spis treści

1. Wprowadzenie.....	2
2. Instrukcja obsługi.....	3
2.1. Główny formularz aplikacji.....	3
2.2. Menu główne.....	5
2.2.1. Menu „Plik”.....	5
2.2.2. Menu „CPU”.....	5
2.2.3. Menu „Rejestry”.....	5
2.2.4. Menu „RAM”.....	6
2.2.5. Menu „Pomoc”.....	6
2.3. Formularz „GPR”.....	6
2.4. Formularz „FPR”.....	8
2.5. Formularz „Rejestry warunków i specjalne”.....	9
2.6. Formularz „Rejestr stanu i sterowania jednostki zmiennopozycyjnej”.....	11
2.7. Formularz „RAM”.....	13
3. Quasi-assembly SimPPC.....	15
3.1. Interpretacja programów.....	16
3.2. Rozkazy mikroprocesora PowerPC™ 601 realizowane przez SimPPC.....	16
3.3. Ograniczenia i uproszczenia.....	16
4. Ekstrakod – mnemoniki.....	19
Dodatek A. Lista rozkazów.....	22
Literatura.....	32

## 1. Wprowadzenie

Niniejszy podręcznik użytkownika przedstawia aplikację **SimPPC w wersji 1.1** oraz sposób jej obsługi. Aplikacja SimPPC, zwana dalej aplikacją, jest symulatorem 32-bitowego mikroprocesora **PowerPC™ 601** o architekturze RISC. Aplikacja ma charakter edukacyjny i przybliża jej użytkownikom budowę, sposób działania oraz listę rozkazów mikroprocesora PowerPC™ 601 jako przedstawiciela rodziny PowerPC™.

Aplikacja została opracowana dla systemu operacyjnego **Microsoft Windows XP** w środowisku programistycznym **Borland C++ Builder Personal 6.0**. Pierwsza wersja aplikacji (v1.0) powstała w 2005 roku. Aktualna wersja (v1.1), która jest przedstawiona w niniejszym podręczniku użytkownika, jest wersją rozwojową i została opracowana w 2013 roku.



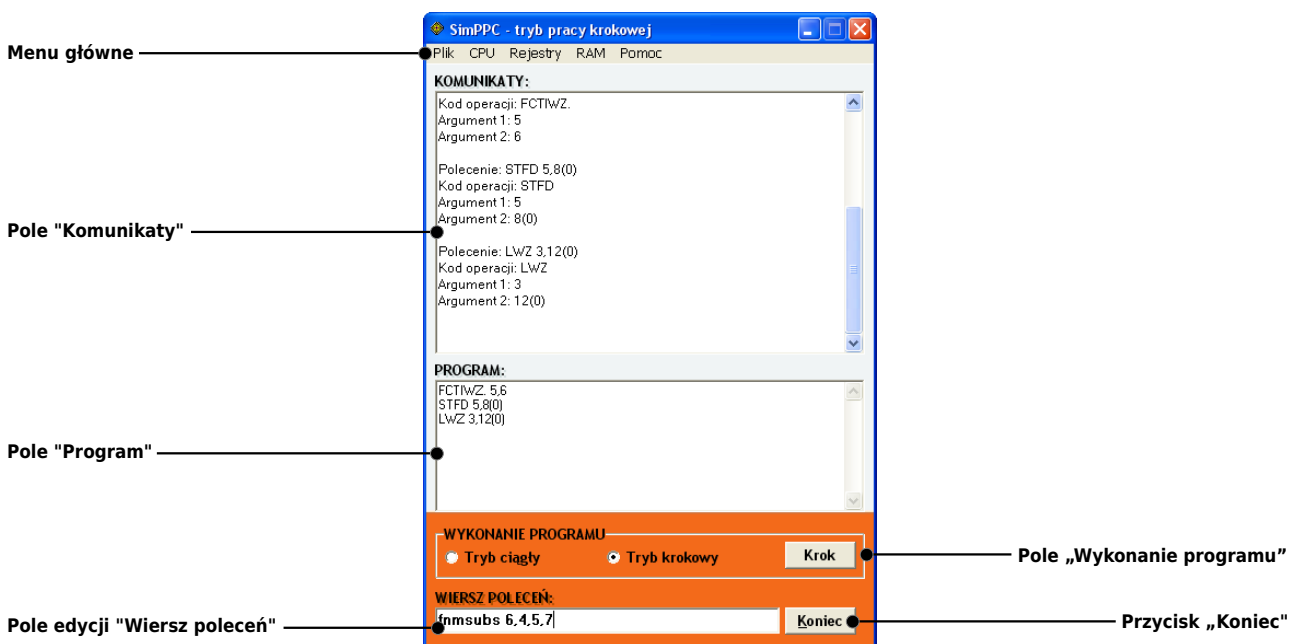
*Rys. 1. Główny formularz SimPPC v1.1  
w trakcie użytkowania aplikacji*

## 2. Instrukcja obsługi

Poniższe podrozdziały zawierają instrukcję obsługi przeznaczoną dla użytkowników aplikacji. Należy mieć na uwadze, iż aplikacja jest nieczuła na wielkość znaków (ang. *case-insensitive*) wprowadzanych bezpośrednio przez użytkowników lub też wczytywanych z plików tekstowych.

### 2.1. Główny formularz aplikacji

Główny formularz aplikacji wyświetlany jest niezwłocznie po jej uruchomieniu.

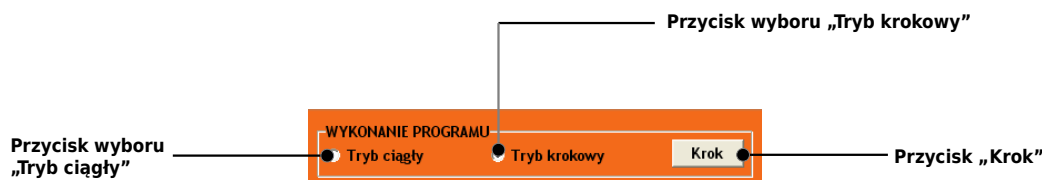


Rys. 2. Główny formularz aplikacji

Główny formularz aplikacji składa się z następujących elementów.

- Pole edycji "Wiersz poleceń" – służy do wprowadzania poleceń, którymi są rozkazy wykonywane przez mikroprocesor oraz mnemoniki służące do bezpośredniego wprowadzania danych do rejestrów lub pamięci RAM. Wprowadzone polecenie należy zatwierdzić klawiszem [Enter]. Więcej informacji na temat rozkazów oraz mnemoników znaleźć można, odpowiednio, w rozdziałach **3. Quasi-asembler SimPPC** i **4. Ekstrakod – mnemoniki**.
- Pole "Komunikaty" – służy do wyświetlania komunikatów informujących użytkownika o składni wprowadzonych rozkazów lub o ewentualnych błędach.

- c) Pole "Program" - wyświetlany jest w nim program wczytany z pliku. Tekst ten jest pozbawiony komentarzy oraz niektórych znaków zawartych w kodzie źródłowym danego programu. Bliższe informacje odnośnie sposobu interpretacji programów przez aplikację znajdują się w podrozdziale **3.1. Interpretacja programów**.
- d) Menu główne - służy do wyświetlania formularzy, wczytywania programów oraz kasowania stanu symulowanego mikroprocesora. Więcej informacji na ten temat znajduje się w podrozdziale **2.2. Menu główne**.
- e) Pole „Wykonanie programu” - w jego skład wchodzi przyciski wyboru „Tryb ciągły” oraz „Tryb krokowy”, a także przycisk „Krok”. Przyciski te spełniają następujące funkcje.
- Przycisk wyboru „Tryb ciągły” - zaznaczenie tego przycisku powoduje wybór trybu pracy ciągłej. W trybie pracy ciągłej aplikacja wykonuje programy wczytanych z plików tekstowych, w sposób ciągły, niezwłocznie po ich otwarciu.
  - Przycisk wyboru „Tryb krokowy” - zaznaczenie tego przycisku powoduje wybór trybu pracy krokowej. W trybie pracy krokowej aplikacja wykonuje programy wczytane z plików tekstowych w sposób krokowy, tj. linijka po linijce, w takt naciskania przycisku „Krok”.
  - Przycisk „Krok” - umożliwia krokowe wykonanie poszczególnych linijek danego programu wczytanego z pliku tekstowego. Przycisk „Krok” jest dostępny dla użytkownika tylko w trybie pracy krokowej, gdy zaznaczony jest przycisk wyboru „Tryb krokowy”.



*Rys. 3. Wchodzące w skład głównego formularza aplikacji pole „Wykonanie programu” wraz z jego przyciskami*

- f) Przycisk „Koniec” - służy do natychmiastowego zakończenia działania aplikacji.

## 2.2. Menu główne

Menu główne składa się z pięciu części:

- a) Plik,
- b) CPU,
- c) Rejestry,
- d) RAM,
- e) Pomoc.

Poszczególne części są szczegółowo omówione w poniższych podrozdziałach.

### 2.2.1. Menu „Plik”

W skład menu „Plik” wchodzi następujące dwa przyciski.

- a) Przycisk "Otwórz" - służy do wczytywania programów zapisanych w formie plików tekstowych. Naciśnięcie przycisku „Otwórz” z menu „Plik” powoduje wywołanie systemowego okna komunikacyjnego służącego do otwierania plików.
- b) Przycisk "Zakończ" - służy do natychmiastowego zamykania aplikacji.

### 2.2.2. Menu „CPU”

Menu „CPU” składa się z przycisku „Reset”. Naciśnięcie przycisku "Reset" powoduje wyzerowanie wszystkich rejestrów symulowanego mikroprocesora.

### 2.2.3. Menu „Rejestry”

Menu „Rejestry” składa się z następujących przycisków.

- a) „GPR” - wyświetlanie formularza bloku rejestrów ogólnego przeznaczenia GPR (ang. *General Purpose Registers*), zwanych dalej rejestrami GPR.
- b) „FPR” - wyświetlanie formularza bloku rejestrów zmiennopozycyjnych FPR (ang. *Floating Point Registers*), zwanych dalej rejestrami FPR.
- c) „CR, SPEC” - wyświetlanie formularza rejestru warunków CR (ang. *Condition Register*), rejestru wyjątków XER (ang. *Integer Exception Register*), rejestru MQ oraz licznika CTR (ang. *Count Register*), zwanych dalej zbiorczo rejestrami warunków i specjalnymi.
- d) „FPSCR” - wyświetlanie formularza rejestru stanu i sterowania jednostki zmiennopozycyjnej FPSCR (ang. *Floating Point Status and Control Register*).

## 2.2.4. Menu „RAM”

Menu "RAM" składa się z następujących przycisków.

- Przycisk „Mapa” - służy do wyświetlania formularza „RAM”, który przedstawia zawartość symulowanej pamięci RAM o wielkości 4KB, która jest do dyspozycji użytkownika.
- Przycisk „Reset” - służy do natychmiastowego kasowania całej zawartości symulowanej pamięci RAM.

## 2.2.5. Menu „Pomoc”

Menu „Pomoc” składa się z następujących 3 przycisków.

- „Zakresy liczb” - służy do wyświetlania formularza prezentującego zakresy liczb całkowitych i zmiennopozycyjnych, wartości graniczne i symbole nieskończoności oraz nie-liczby (NaN).
- „O autorze” - wyświetla formularz przedstawiający informacje o autorze aplikacji.
- „O programie” - wyświetla formularz zawierający informacje o aplikacji oraz skróconą wersję instrukcji obsługi dla użytkowników.

## 2.3. Formularz „GPR”

Formularz „GPR” wyświetlany jest po naciśnięciu przycisku „GPR” z menu „Rejestry”. Jego najważniejszym elementem są 32 pola edycji, które przedstawiają stan rejestrów GPR (GPR0 – GPR31). Znajdujące się w dolnej części formularza „GPR” przyciski „DEC” i „HEX” służą do zmiany systemu liczbowego (odpowiednio dziesiętnego i szesnastkowego), w którym prezentowana jest zawartość rejestrów GPR. Zawartość rejestrów GPR jest aktualizowana na bieżąco, dzięki czemu jest ona zgodna z aktualnym stanem symulowanego mikroprocesora.



Rys. 4. Formularz „GPR” prezentujący wartości rejestrów GPR w systemie dziesiętnym



Należy mieć na uwadze, iż zmiany zawartości rejestrów GPR są sygnalizowane chwilową zmianą koloru stosownych pól edycji z białego na zielony.

Użytkownik ma możliwość zmiany zawartości poszczególnych rejestrów GPR poprzez pola edycji umieszczone na formularzu. W tym celu należy kliknąć lewym przyciskiem myszki na polu edycji prezentującym wybrany rejestr GPR. Formularz „GPR” zmieni wówczas swój kolor na szary.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	
00000012	0000000A	000007BF	00000000	00000000	00000000	00000000	00000000	00000000	00000000	FFFFFF906	00000000	000000FF	ead00785	00000000	00000000	00000000	00000000	00000000	7FFFFFFF	00000000	80000000	00000000	00000000	FFFFFFF	00000000	00000000	00000000	00000000	00000000	0000000E	0000000C	000007DC

Tryb wyświetlania  
☐ DEC ☒ HEX

**Rys. 4. Formularz „GPR” podczas wprowadzania wartości do jednego z rejestrów GPR w systemie szesnastkowym**

Następnie należy wprowadzić żadaną wartość, zgodnie z systemem liczbowym, w którym jest aktualnie prezentowana zawartość rejestrów GPR.

Przykłady wprowadzanych do rejestrów GPR wartości.

a) Tryb wyświetlania DEC

**-2147483647**

**128**

b) Tryb wyświetlania HEX

**7FFFFFFF**

**000007CD**

Wprowadzoną w danym polu edycji wartość należy zatwierdzić klawiszem [Enter]. Formularz „GPR” powróci wówczas do swego standardowego koloru.

## 2.4. Formularz „FPR”

Formularz „FPR” wyświetlany jest po naciśnięciu przycisku „FPR” z menu „Rejestry”. Jego najważniejszym elementem są 32 pola edycji, które przedstawiają stan rejestrów FPR (FPR0 – FPR31). Znajdujące się w dolnej części formularza „FPR” przyciski „DEC” i „HEX” służą do zmiany systemu liczbowego (odpowiednio dziesiętnego i szesnastkowego), w którym prezentowana jest zawartość rejestrów FPR. Zawartość rejestrów FPR jest aktualizowana na bieżąco, dzięki czemu jest ona zgodna z aktualnym stanem symulowanego mikroprocesora.

The screenshot shows a window titled "FPR - rejestry zmiennopozycyjne". It contains a grid of 32 text input fields, each representing a register. The registers are numbered 0 to 31. The values are displayed in hexadecimal (HEX) format. For example, register 0 contains "0000000000000000", register 1 contains "0000000000000000", register 2 contains "0000000000000000", register 3 contains "0000000000000000", register 4 contains "0000000000000000", register 5 contains "BFF0000000000000", register 6 contains "0000000000000000", register 7 contains "404A800000000000", register 8 contains "C031800000000000", register 9 contains "0000000000000000", register 10 contains "BFE851E851E85", register 11 contains "0000000000000000", register 12 contains "0000000000000000", register 13 contains "0000000000000000", register 14 contains "0000000000000000", register 15 contains "0000000000000000", register 16 contains "3EEAD7F29ABCAF48", register 17 contains "0000000000000000", register 18 contains "7FF0000000000000", register 19 contains "0000000000000000", register 20 contains "7FF0000000000001", register 21 contains "0000000000000000", register 22 contains "0000000000000000", register 23 contains "4040AAAA8EB46349", register 24 contains "4025000000000000", register 25 contains "0000000000000000", register 26 contains "FFF0000000000000", register 27 contains "0000000000000000", register 28 contains "0000000000000000", register 29 contains "0000000000000000", register 30 contains "0000000000000000", and register 31 contains "0000000000000000". At the bottom of the window, there is a section labeled "Tryb wyświetlania" with two radio buttons: "DEC" (selected) and "HEX".

*Rys. 5. Formularz „FPR” prezentujący wartości rejestrów FPR w systemie szesnastkowym*

Należy mieć na uwadze, iż zmiany zawartości rejestrów FPR są sygnalizowane chwilową zmianą koloru stosownych pól edycji z białego na pomarańczowy.

Użytkownik ma możliwość zmiany zawartości poszczególnych rejestrów FPR poprzez pola edycji umieszczone na formularzu. W tym celu należy kliknąć lewym przyciskiem myszki na polu edycji prezentującym wybrany rejestr FPR. Formularz „FPR” zmieni wówczas swój kolor na szary.

The screenshot shows the same window "FPR - rejestry zmiennopozycyjne", but the values are now displayed in decimal (DEC) format. The registers are numbered 0 to 31. The values are: register 0: 0, register 1: 0, register 2: 0, register 3: 0, register 4: 0, register 5: -1, register 6: 0, register 7: 53, register 8: -17.5, register 9: 0, register 10: -0.86, register 11: 0, register 12: 0, register 13: 0, register 14: 0, register 15: 0, register 16: 1.28E-5, register 17: 0, register 18: INF, register 19: 0, register 20: NAN, register 21: -1967.0501E-2, register 22: 0, register 23: 33.33333, register 24: 10.5, register 25: 0, register 26: -INF, register 27: 0, register 28: 0, register 29: 0, register 30: 0, and register 31: 0. At the bottom, the "Tryb wyświetlania" section now has the "DEC" radio button selected and the "HEX" radio button unselected.

*Rys. 6. Formularz „FPR” podczas wprowadzania wartości do jednego z rejestrów FPR w systemie dziesiętnym*

Następnie należy wprowadzić żadaną wartość, zgodnie z systemem liczbowym, w którym jest

aktualnie prezentowana zawartość rejestrów FPR. Przykłady wprowadzania wartości do rejestrów FPR są przedstawione poniżej.

- a) Tryb wyświetlania DEC

**-17,5**

**0,3333**

**1,5426E-9**

**INF**

**NAN**

- b) Tryb wyświetlania HEX

**7FF0000000000001**

**3FF0000000000000**

**4005BF0A87427F01**

Wprowadzoną w danym polu edycji wartość należy zatwierdzić klawiszem [Enter]. Formularz „FPR” powróci wówczas do swego standardowego koloru.

Należy mieć na uwadze, iż w przypadku systemu szesnastkowego wprowadzane i prezentowane w formularzu „FPR” wartości są zgodne ze standardem IEEE 754 reprezentacji liczb zmiennopozycyjnych podwójnej precyzji (ang. *64-bit Double Precision*).

## **2.5. Formularz „Rejestry warunków i specjalne”**

Formularz „Rejestry warunków i specjalne” wyświetlany jest po naciśnięciu przycisku „CR, SPEC”, który znajduje się w menu „Rejestry”. Formularz „Rejestry warunków i specjalne” prezentuje aktualny stan następujących rejestrów mikroprocesora PowerPC™ 601.

- a) Rejestr warunków CR (zwany dalej „rejestrem CR”) oraz jego flag z pól CR0 i CR1 (odpowiednio LT, GT, EQ, SO, i FX, FEX, VX, OX).
- b) Rejestr wyjątków XER (zwany dalej „rejestrem XER”) oraz jego flag SO, OV i CA.
- c) Rejestr MQ.
- d) Rejestr licznika CTR.

Widoczne w dolnej części formularza przyciski „DEC” i „HEX” służą do zmiany systemu liczbowego, w którym wyświetlana jest zawartość rejestrów (odpowiednio dziesiętnego i szesnastkowego). Zawartość rejestrów warunków i specjalnych jest aktualizowana na bieżąco, aby odzwierciedlać aktualny stan mikroprocesora.

The dialog box 'Rejestry warunków i specjalne' displays the following values and flags:

- CR:** DD0007A3
- Pola rejestru CR:**
  - CR0: 1, LT: 1, GT: 0, EQ: 1, SO: 1
  - CR1: 1, FX: 1, FEX: 0, VX: 1, OX: 1
- XER:** C0000000
- Flagi rejestru XER:**
  - SO: 1, OV: 1, CA: 0
- MQ:** 00000FFF
- CTR:** 0000003E
- Tryb wyświetlania:** DEC (radio button), **HEX** (radio button, selected)

Rys. 7. Formularz „Rejestry warunków i specjalne” prezentujący wartości rejestrów CR, XER, MQ i licznika CTR w systemie szesnastkowym

Należy mieć na uwadze, iż zmiany zawartości rejestrów warunków i specjalnych lub zmiany stanu prezentowanych flag są sygnalizowane chwilową zmianą koloru odpowiednich pól edycji z białego na fioletowy.

Użytkownik ma możliwość zmiany zawartości rejestrów warunków i specjalnych. W tym celu należy kliknąć lewym przyciskiem myszki na polu edycji prezentującym wybrany rejestr. Formularz „Rejestry warunków i specjalne” zmieni wówczas swój kolor na szary.

The dialog box 'Rejestry warunków i specjalne' displays the following values and flags, with the background color changed to gray:

- CR:** 9D0007A3
- Pola rejestru CR:**
  - CR0: 1, LT: 0, GT: 0, EQ: 1, SO: 1
  - CR1: 1, FX: 1, FEX: 0, VX: 1, OX: 1
- XER:** 00000000
- Flagi rejestru XER:**
  - SO: 1, OV: 1, CA: 0
- MQ:** 00000FFF
- CTR:** 0000003E
- Tryb wyświetlania:** DEC (radio button), **HEX** (radio button, selected)

Rys. 8. Formularz „Rejestry warunków i specjalne” podczas wprowadzania wartości do jednego z rejestrów specjalnych w systemie szesnastkowym

Następnie należy wprowadzić żadaną wartość, zgodnie z systemem liczbowym, w którym jest aktualnie prezentowana zawartość rejestrów warunków i specjalnych. Wprowadzoną w danym polu edycji wartość należy zatwierdzić klawiszem [Enter]. Formularz „Rejestry warunków i specjalne”

powróci wówczas do swego standardowego koloru.

Użytkownik ma ponadto możliwość bezpośredniej zmiany stanu flag rejestrów CR i XER. W celu zmiany stanu danej flagi, należy kliknąć lewym przyciskiem myszki na odpowiednie pole edycji. Spowoduje to jednoczesną zmianę stanu wybranej flagi jak i rejestru, do którego należy.

## 2.6. Formularz „Rejestr stanu i sterowania jednostki zmiennopozycyjnej”

Formularz „Rejestr stanu i sterowania jednostki zmiennopozycyjnej” wyświetlany jest po naciśnięciu przycisku „FPSCR” z menu „Rejestry” i prezentuje on stan flag oraz zawartość rejestru FPSCR.

Znajdujące się w dolnej części formularza „Rejestr stanu i sterowania jednostki zmiennopozycyjnej” przyciski „DEC” i „HEX” służą do zmiany systemu liczbowego (odpowiednio dziesiętnego i szesnastkowego), w którym prezentowana jest zawartość rejestru FPSCR. Zawartość rejestru FPSCR jest aktualizowana na bieżąco, dzięki czemu jest ona zgodna z aktualnym stanem symulowanego mikroprocesora.

Flagi											
0	1	FX	8	1	VXISI	16	1	FL	24	0	VE
1	0	FEX	9	0	VXIDI	17	0	FG	25	0	OE
2	1	VX	10	0	VXZDZ	18	0	FE	26	0	UE
3	0	OX	11	1	VXIMZ	19	0	FU	27	1	ZE
4	0	UX	12	0	VXVC	20	0	Res	28	1	XE
5	0	ZX	13	0	FR	21	0	VXSOFT	29	0	Res
6	0	XX	14	0	FI	22	0	VXSQRT	30	0	RN 0
7	0	VXSNaN	15	0	C	23	1	VXCVI	31	0	RN 1

A0908118    FPSCR

Tryb wyświetlania  
☐ DEC    ☒ HEX

*Rys. 9. Formularz „Rejestr stanu i sterowania jednostki zmiennopozycyjnej” prezentujący wartości rejestru FPSCR w systemie szesnastkowym*

Należy mieć na uwadze, iż zmiany zawartości rejestru FPSCR lub zmiany stanu jego flag są sygnalizowane chwilową zmianą koloru stosownych pól edycji z białego na fioletowy.

Użytkownik ma możliwość zmiany zawartości rejestru FPSCR. W tym celu należy kliknąć lewym przyciskiem myszki na polu edycji „FPSCR”. Formularz „Rejestry warunków i specjalne” zmieni wówczas swój kolor na szary. Następnie należy wprowadzić żadaną wartość, zgodnie z systemem liczbowym, w którym jest aktualnie prezentowana zawartość rejestru FPSCR. Wprowadzoną w polu edycji „FPSCR” wartość należy zatwierdzić klawiszem [Enter]. Formularz „Rejestr stanu i sterowania jednostki zmiennopozycyjnej” powróci wówczas do swego standardowego koloru.

Flagi											
0	<input type="checkbox"/> 0	FX	8	<input type="checkbox"/> 0	VXISI	16	<input type="checkbox"/> 0	FL	24	<input type="checkbox"/> 0	VE
1	<input checked="" type="checkbox"/> 1	FEX	9	<input type="checkbox"/> 0	VXIDI	17	<input checked="" type="checkbox"/> 1	FG	25	<input checked="" type="checkbox"/> 1	OE
2	<input type="checkbox"/> 0	VX	10	<input checked="" type="checkbox"/> 1	VXZDZ	18	<input type="checkbox"/> 0	FE	26	<input checked="" type="checkbox"/> 1	UE
3	<input type="checkbox"/> 0	OX	11	<input type="checkbox"/> 0	VXIMZ	19	<input type="checkbox"/> 0	FU	27	<input type="checkbox"/> 0	ZE
4	<input checked="" type="checkbox"/> 1	UX	12	<input type="checkbox"/> 0	VXVC	20	<input type="checkbox"/> 0	Res	28	<input type="checkbox"/> 0	XE
5	<input type="checkbox"/> 0	ZX	13	<input type="checkbox"/> 0	FR	21	<input type="checkbox"/> 0	VXSOF	29	<input type="checkbox"/> 0	Res
6	<input type="checkbox"/> 0	XX	14	<input type="checkbox"/> 0	FI	22	<input type="checkbox"/> 0	VXSQRT	30	<input type="checkbox"/> 0	RN 0
7	<input type="checkbox"/> 0	VXSNAN	15	<input checked="" type="checkbox"/> 1	C	23	<input checked="" type="checkbox"/> 1	VXCVI	31	<input type="checkbox"/> 0	RN 1

48214160    FPSCR

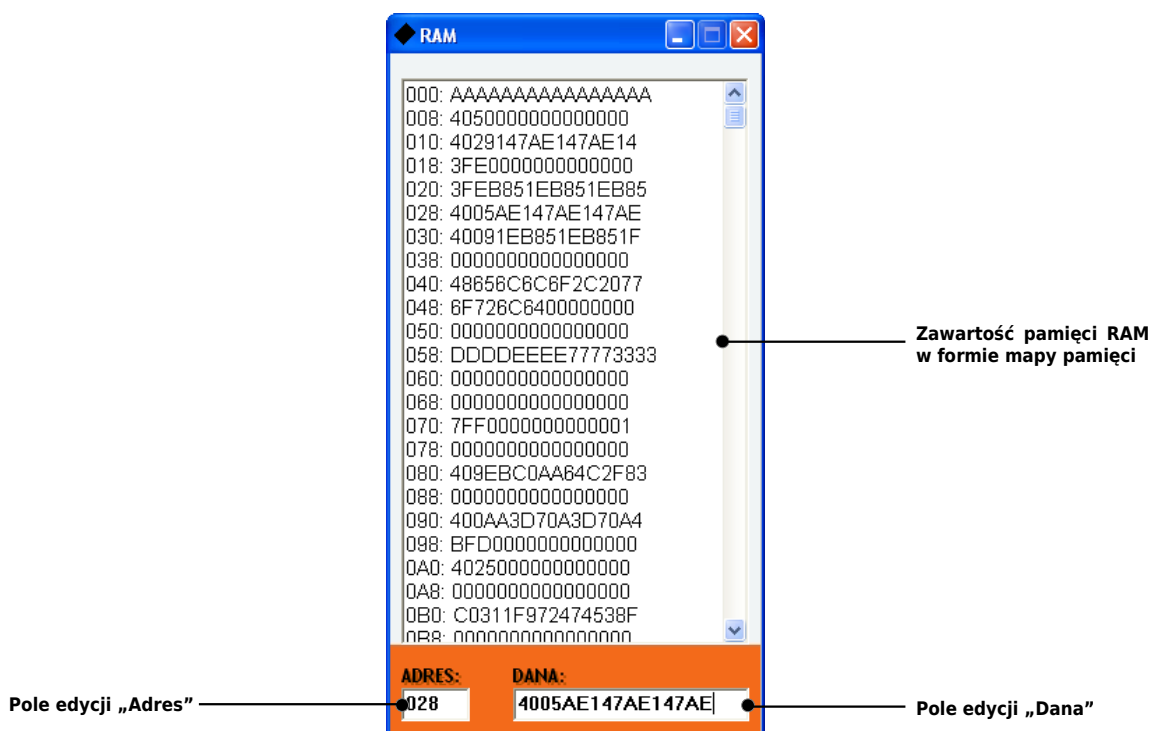
Tryb wyświetlania  
☐ DEC    ☒ HEX

*Rys. 10. Formularz „Rejestr stanu i sterowania jednostki zmiennopozycyjnej” podczas wprowadzania wartości do rejestru FPSCR w systemie szesnastkowym*

Użytkownik ma ponadto możliwość bezpośredniej zmiany stanu flag rejestru FPSCR. W celu zmiany stanu danej flagi, należy kliknąć lewym przyciskiem myszki na odpowiednie pole edycji. Spowoduje to jednoczesną zmianę stanu wybranej flagi jak i zawartości rejestru FPSCR.

## 2.7. Formularz „RAM”

Formularz „RAM” prezentuje zawartość pamięci RAM w formie mapy pamięci. Symulowana pamięć RAM ma wielkość 4KB i jest ona pamięcią danych, zapisywanych w układzie big endian. W celu wyświetlenia formularza „RAM” należy nacisnąć przycisk „Mapa”, który znajduje się w menu „RAM”. Formularz „RAM” daje użytkownikowi możliwość bezpośredniego wprowadzania danych do pamięci RAM. Do tego celu służy pole edycji „Adres” oraz pole edycji „Dana”, które znajdują się w dolnej części formularza „RAM”.



Rys. 11. Formularz „RAM” prezentujący zawartość pamięci RAM

W celu wprowadzenia danej do pamięci RAM poprzez formularz „RAM” należy wykonać następujące czynności.

- W polu edycji „Adres” należy wprowadzić adres w postaci liczby szesnastkowej z zakresu 0 - FFF.
- W polu edycji „Dana” należy wprowadzić wartość w postaci liczby szesnastkowej. Dopuszczalna długość danych wynosi 64 bity.
- Wprowadzone wartości należy zatwierdzić naciśnięciem klawisza [Enter] w polu edycji „Dana” lub w polu edycji „Adres”.

Pomiędzy polami edycji adresu i danych można przemieszczać się przy pomocy klawisza tabulacji

[Tab]. Należy mieć na uwadze, iż dane zostaną zapisane w pamięci RAM tylko wtedy, gdy będą one dopasowane do adresu. Użytkownik może zaadresować bajt (8b), półsłowo (16b), słowo (32b) lub podwójne słowo (64b), które to możliwości obrazują poniższe przykłady.

a) Bajt (8b)

**ADRES: FC7**

**DANA: AB**

b) Półsłowo (16b)

**ADRES: 20A**

**DANA: 1234**

c) Słowo (32b)

**ADRES: 124**

**DANA: AAAABBBB**

d) Podwójne słowo (64b)

**ADRES: 640**

**DANA: 11112222CCCCDDDD**

Wszelkie zmiany zawartości pamięci RAM są sygnalizowane przez aplikację chwilową zmianą koloru tła mapy pamięci widocznej w formularzu „RAM” z białego na błękitny.



### 3. Quasi-assembly SimPPC

Aplikacja umożliwia wykonanie pojedynczych rozkazów mikroprocesora PowerPC™ 601 oraz programów quasi-assemblya SimPPC, zapisanych w plikach tekstowych. Format rozkazów wykonywanych przez aplikację, a także składnia quasi-assemblya SimPPC są zasadniczo zgodne ze standardem przedstawionym przez IBM w „AIX 5L for POWER-based Systems. Assembly Language Reference, Second Edition (April 2001)”.

Najważniejsze zasady dotyczące quasi-assemblya SimPPC oraz rozkazów wykonywanych przez aplikację są następujące.

- a) Rejestry GPR oraz FPR są oznaczane w rozkazach liczbą z przedziału 0 - 31, na przykład:

**ABSO 7,14**

**fadd. 6,4,25**

- b) Pola rejestru CR są oznaczane w rozkazach liczbą z przedziału 0 - 7, na przykład:

**FCMPO 4,5,20**

- c) Bit rejestru CR oznaczany jest w rozkazach liczbą z zakresu 0 - 31, na przykład:

**CRORC 29,30,31**

- d) Wartości argumentów natychmiastowych mogą być wprowadzane w systemie dziesiętnym oraz szesnastkowym. W systemie dziesiętnym można stosować znak minus „-”, poprzedzający wartość ujemną. Wartości zapisane w systemie szesnastkowym muszą być poprzedzone przedrostkiem 0x, x lub \$. Przykładowe rozkazy:

**LSWI 4,8,20**

**ANDI. 7,10,0X0A8**

- e) Komentarze zaczynają się znakiem hash („#”), tzn.

**# To jest komentarz w programie.**

W porównaniu do składni assemblya PowerPC™ IBM składnia quasi-assemblya SimPPC różni się w następujących aspektach.

- a) Znak dolara („\$”) w quasi-assemblyerze SimPPC oznacza wartość szesnastkową lub mnemonik bezpośredniego wprowadzania wartości do pamięci RAM.
- b) Każde polecenie w quasi-assemblyerze SimPPC musi znajdować się w nowej linii.

Programy powinny być zapisywane w plikach tekstowych w kodowaniu (ANSI). Ponadto tworząc programy należy uwzględnić listę rozkazów, które są obsługiwane przez aplikację, a także wziąć pod uwagę ograniczenia dotyczące pamięci RAM.

### **3.1. Interpretacja programów**

Wczytane z plików tekstowych programy są interpretowane przez aplikację w następujący sposób.

- a) Wszelkie białe znaki (spacje, znaki tabulacji) znajdujące się w danej linii są usuwane. Jedynym wyjątkiem jest spacja oddzielająca kod rozkazu od operandów.
- b) Maksymalna liczba przecinków w danej linii kodu wynosi 4. Wynika to z maksymalnej liczby operandów rozkazu, których może być co najwyżej 5.
- c) Znaki dwukropek („:”), lewy („(”) i prawy nawias („)”), minus („-”) oraz znak równości („=”) mogą wystąpić w danej linii tylko raz.
- d) Etykiety nie mogą zaczynać się cyframi. Ponadto w nazwie etykiety nie może wystąpić biały znak lub asterisk (tj. gwiazdka, „\*”). Koniec etykiety oznaczany jest dwukropkiem („:”).
- e) W danym programie można zastosować co najwyżej 64 etykiety.

### **3.2. Rozkazy mikroprocesora PowerPC™ 601 realizowane przez SimPPC**

Aplikacja wykonuje podzbiór listy rozkazów mikroprocesora PowerPC™ 601. Pełną listę rozkazów dostępnych w aplikacji przedstawia **Dodatek A. Lista rozkazów**.

### **3.3. Ograniczenia i uproszczenia**

Użytkownicy powinni być świadomi uproszczeń oraz ograniczeń zastosowanych w aplikacji odnośnie sposobu wykonywania poszczególnych rozkazów, w szczególności tych, które są wykonywane na liczbach zmiennopozycyjnych. Wyniki uzyskiwane w aplikacji mogą w związku z tym odbiegać od tych, które występują w rzeczywistym mikroprocesorze PowerPC™ 601. Różnice dotyczą w szczególności następujących obszarów.

- a) Nie-liczby (NaN)

W aplikacji wszystkie nie-liczby (NaN) są reprezentowane przez wartość 0x7FF0000000000001 w przypadku podwójnej precyzji (64b) oraz przez wartość 0xFF800001 w zakresie pojedynczej precyzji (32b). Tym samym w aplikacji nie ma rozróżnienia pomiędzy SNaN (ang. *signaling NaN*), a QNaN (ang. *quiet NaN*), przez co sposób ustawiania flagi VXSNaN (ang. *Floating-Point Invalid Operation Exception bit*) rejestru FPSCR jest uproszczony. Flaga VXSNaN ustawiana jest w aplikacji wtedy, kiedy wynikiem stosownego rozkazu

zmiennopozycyjnego jest nie-liczba.

b) Niedomiar zmiennopozycyjny (ang. *underflow*)

Aplikacja nie wykrywa niedomiaru zmiennopozycyjnego, a tym samym sygnalizująca stosowny wyjątek flaga UX (ang. *Floating-point underflow exception*) rejestru FPSCR nie jest ustawiana w wyniku wykonywanych rozkazów.

c) Nadmiar zmiennopozycyjny (ang. *overflow*)

W niektórych przypadkach aplikacja może nie wykryć nadmiaru zmiennopozycyjnego, którego wystąpienie sygnalizowane jest flagą OX (ang. *Floating-point overflow exception*) rejestru FPSCR.

d) Pożyczka (ang. *carry*)

W przypadku niektórych rozkazów arytmetycznych (jak np. ADDEO lub SUBFEO) aplikacja może nie wykryć pożyczki, której wystąpienie sygnalizowane jest flagą CA (ang. *Carry bit*) rejestru XER.

e) Zaokrąglanie

Aplikacja nie wykrywa błędów związanych z zaokrąglaniem liczb zmiennopozycyjnych. Sygnalizująca stosowne wyjątki flaga XX (ang. *Floating-point inexact exception*) rejestru FPSCR nie jest ustawiana w wyniku wykonywanych rozkazów. Ze względu na brak obsługi wyników pośrednich (ang. *intermediate result*) w wyniku wykonywanych rozkazów nie są też ustawiane flagi FR (ang. *fraction incremented*) i FI rejestru FPSCR (ang. *rounded result inexact*).

f) Obsługa wyjątków zmiennopozycyjnych

Sposób obsługi poszczególnych wyjątków zmiennopozycyjnych sygnalizowanych przez flagi rejestru FPSCR (VX, OX, UX, ZX, XX) zależy od stanu flag (VE, OE, UE, ZE, XE) rejestru FPSCR, dopuszczających odpowiednie wyjątki. W porównaniu do rzeczywistego mikroprocesora PowerPC™ 601 mogą wystąpić różnice pomiędzy sposobem obsługi wyjątków zmiennopozycyjnych, a stanem dopuszczających je flag (VE, OE, UE, ZE, XE) rejestru FPSCR.

g) Wartości zmiennopozycyjne

W przypadku działań na liczbach zmiennopozycyjnych mogą wystąpić przekłamania w niektórych wartościach. W szczególności dotyczy to bardzo małych wartości zmiennopozycyjnych podwójnej precyzji (64b) oraz ich obustronnej konwersji z systemu szesnastkowego na dziesiętny.

h) Rozkazy skoków

W rozkazach skoków zbiór wartości operandu target\_addr ogranicza się tylko do występujących w danym programie etykiet.

Przykłady:

**B Label1**

**BC 0x0C,0x01,Swap**

**BCA 16,0,LABEL**

**BCL 9,0,Main**

**BCLA 10,0,LABEL**

## 4. Ekstrakod - mnemoniki

W aplikacji dostępny jest ekstrakod w postaci mnemoników, które umożliwiają bezpośrednie wprowadzenie wartości do rejestrów oraz do pamięci RAM. Użytkownik może stosować mnemoniki tak w polu edycji "Wiersz poleceń", jak i w programach quasi-asmblera SimPPC. Należy przy tym mieć na uwadze, iż wprowadzane wartości powinny być dostosowane do formatu oraz długości danych przechowywanych w pamięci RAM lub poszczególnych rejestrach.

### a) Bezpośrednie wprowadzanie wartości do rejestrów GPR i FPR

Składnia mnemoników umożliwiających bezpośrednie wprowadzanie wartości do rejestrów GPR i rejestrów FPR jest następująca:

**REJESTR\_NUMER = WARTOŚĆ**

Rejestry GPR oznaczane są za pomocą litery "R", a rejestry FPR literą "F". Wprowadzone wartości mogą mieć postać liczb dziesiętnych lub szesnastkowych. W tym drugim przypadku wartość musi być poprzedzona przedrostkiem 0x, x lub \$.

Przykłady:

**r5 = 1417**

**f0 = -0,86**

**R5 = -9**

**R7 = x13**

**R17 = 0x33**

**R30 = 0xFF**

**F6 = -76**

**F7 = 12,788**

**f8 = 123,457E-5**

**F5 = xFF00000000000000**

**f15 = \$0771abcdabcdabcd**

### b) Bezpośrednie wprowadzanie wartości do rejestrów CR, MQ, XR, CTR i FPSCR

Mnemoniki umożliwiające bezpośrednie wprowadzanie wartości do rejestrów CR, MQ, XR, CTR i FPSCR mają następującą składnię.

**NAZWA\_REJESTRU = WARTOŚĆ**

Parametr NAZWA\_REJESTRU odpowiada nazwom poszczególnych rejestrów tj. CR, MQ, XR, CTR lub FPSCR.

Wprowadzane wartości mogą być liczbami dziesiętnymi lub szesnastkowymi (z przedrostkami 0x, x lub \$), na przykład:

**CR = xABCDEF56**

**MQ = 0XBBB**

**CTR = 50**

**XER = 0xE0000000**

**FPSCR = 0x90910000**

Ponadto aplikacja obsługuje podzbiór rozszerzonych mnemoników (ang. *Extended Mnemonics*) dla mikroprocesora PowerPC™ 601. Rozszerzone mnemoniki umożliwiają przenoszenie wartości pomiędzy rejestrami warunków i specjalnymi, a rejestrami GPR.

- Przenoszenie wartości z rejestrów warunków i specjalnych do rejestrów GPR.

**MFCR rt**

**MFMQ rt**

**MTXER rt**

**MFCTR rt**

Przykłady:

**MFCTR 8**

**MTXER 6**

- Przenoszenie wartości z rejestrów GPR do rejestrów warunków i specjalnych.

**MTCR rs**

**MTMQ rs**

**MTXER rs**

**MTCTR rs**

Przykłady:

**MTCR 30**

**MTMQ 20**

c) Bezpośrednie wprowadzanie wartości do pamięci RAM

Mnemoniki umożliwiające bezpośrednie wprowadzanie wartości do pamięci RAM mają następującą składnię.

**\$ADRES = WARTOŚĆ**

Wprowadzane wartości mogą być liczbami dziesiętnymi lub szesnastkowymi (z przedrostkami 0x, x lub \$). Sugerowane jest stosowanie wartości w postaci liczb szesnastkowych, gdyż daje to użytkownikowi większą swobodę w dostępie do pamięci RAM (np. możliwość zaadresowania pojedynczych bajtów) oraz większy zakres wprowadzanych wartości. W przypadku wartości dziesiętnych aplikacja obsługuje tylko liczby całkowite, które w pamięci RAM są reprezentowane jako 64-bitowe liczby szesnastkowe.

Przykłady:

**\$7=\$AB**

**\$08=XF5ABCD01**

**\$28=x7f800000**

**\$090=456789**

Należy wziąć pod uwagę, iż białe znaki występujące w mnemonikach są ignorowane, tzn. polecenia

**R3=255**

**R3 = 255**

są sobie równoważne.

d) Rozkaz pusty (NOP)

W ekstrakodzie dostępny jest mnemonik NOP (ang. *no-operation*), zwany rozkazem pustym lub rozkazem beczynności. Rozkaz pusty wprowadza się w następujący sposób.

**NOP**

e) Powrót z podprogramu (BLR)

Powrót z podprogramu określony jest mnemonikiem BLR (ang. *Branch to Link Register*). Wykonanie powrotu z podprogramu zobrazowane jest na poniższym przykładzie.

**Store:**

**STSWI 6,4,12**

**BLR**

**Dodatek A. Lista rozkazów**

ROZKAZ	SKŁADNIA	PRZYKŁAD
ABS	ABS RT, RA	ABS 6,4
ABS.	ABS. RT, RA	ABS. 6,4
ABSO	ABSO RT, RA	ABSO 6,4
ABSO.	ABSO. RT, RA	ABSO. 6,4
ADD	ADD RT, RA, RB	ADD 4,6,3
ADD.	ADD. RT, RA, RB	ADD. 4,6,3
ADDO	ADDO RT, RA, RB	ADDO 4,6,3
ADDO.	ADDO. RT, RA, RB	ADDO. 4,6,3
ADDC	ADDC RT, RA, RB	ADDC 6,4,10
ADDC.	ADDC. RT, RA, RB	ADDC. 6,4,10
ADDCO	ADDCO RT, RA, RB	ADDCO 6,4,10
ADDCO.	ADDCO. RT, RA, RB	ADDCO. 6,4,10
ADDE	ADDE RT, RA, RB	ADDE 6,4,10
ADDE.	ADDE. RT, RA, RB	ADDE. 6,4,10
ADDEO	ADDEO RT, RA, RB	ADDEO 6,4,10
ADDEO.	ADDEO. RT, RA, RB	ADDEO. 6,4,10
ADDI	ADDI RT, RA, SI	ADDI 4,5,0XFFFF8FF0
ADDIC	ADDIC RT, RA, SI	ADDIC 6,4,0XFFFFFFFF
ADDIC.	ADDIC. RT, RA, SI	ADDIC. 6,4,0X1000
ADDIS	ADDIS RT, RA, SI	ADDIS 7,6,0X0011
ADDME	ADDME RT, RA	ADDME 6,4
ADDME.	ADDME. RT, RA	ADDME. 6,4
ADDMEO	ADDMEO RT, RA	ADDMEO 6,4
ADDMEO.	ADDMEO. RT, RA	ADDMEO. 6,4
ADDZE	ADDZE RT, RA	ADDZE 6,4
ADDZE.	ADDZE. RT, RA	ADDZE. 6,4
ADDZEO	ADDZEO RT, RA	ADDZEO 6,4
ADDZEO.	ADDZEO. RT, RA	ADDZEO. 6,4
AND	AND RA, RS, RB	AND 6,4,7
AND.	AND. RA, RS, RB	AND. 6,4,7
ANDC	ANDC RA, RS, RB	ANDC 6,4,5
ANDC.	ANDC. RA, RS, RB	ANDC. 6,4,5
ANDI.	ANDI. RA, RS, UI	ANDI. 6,4,0X5730



ROZKAZ	SKŁADNIA	PRZYKŁAD
ANDIS.	ANDIS. RA, RS, UI	ANDIS. 6,4,0X5730
B	B target_address	B LABEL
BA	BA target_address	BA LABEL
BL	BL target_address	BL LABEL
BLA	BLA target_address	BLA LABEL
BC	BC BO, BI, target_address	BC 19,0,LABEL
BCA	BCA BO, BI, target_address	BCA 16,0,LABEL
BCL	BCL BO, BI, target_address	BCL 9,0,LABEL
BCLA	BCLA BO, BI, target_address	BCLA 10,0,LABEL
CMP	CMP BF, L, RA, RB	CMP 0,4,6
CMPI	CMPI BF, L, RA, SI	CMPI 0,4,0X11
CMPL	CMPL BF, L, RA, RB	CMPL 0,4,5
CMPLI	CMPLI BF, L, RA, UI	CMPLI 0,4,0XFF
CNTLZW	CNTLZW RA, RS	CNTLZW 3,3
CNTLZW.	CNTLZW. RA, RS	CNTLZW. 3,3
CRAND	CRAND BT, BA, BB	CRAND 31,0,5
CRANDC	CRANDC BT, BA, BB	CRANDC 31,0,5
CREQV	CREQV BT, BA, BB	CREQV 4,8,4
CRNAND	CRNAND BT, BA, BB	CRNAND 4,8,4
CRNOR	CRNOR BT, BA, BB	CRNOR 4,8,4
CROR	CROR BT, BA, BB	CROR 4,8,4
CRORC	CRORC BT, BA, BB	CRORC 4,8,4
CRXOR	CRXOR BT, BA, BB	CRXOR 4,8,4
DIV	DIV RT, RA, RB	DIV 4,4,6
DIV.	DIV. RT, RA, RB	DIV. 4,4,6
DIVO	DIVO RT, RA, RB	DIVO 4,4,6
DIVO.	DIVO. RT, RA, RB	DIVO. 4,4,6
DIVS	DIVS RT, RA, RB	DIVS 4,4,6
DIVS.	DIVS. RT, RA, RB	DIVS. 4,4,6
DIVSO	DIVSO RT, RA, RB	DIVSO 4,4,6
DIVSO.	DIVSO. RT, RA, RB	DIVSO. 4,4,6
DIVW	DIVW RT, RA, RB	DIVW 4,4,6
DIVW.	DIVW. RT, RA, RB	DIVW. 4,4,6
DIVWO	DIVWO RT, RA, RB	DIVWO 4,4,6

ROZKAZ	SKŁADNIA	PRZYKŁAD
DIVWO.	DIVWO. RT, RA, RB	DIVWO. 4,4,6
DIVWU	DIVWU RT, RA, RB	DIVWU 4,4,6
DIVWU.	DIVWU. RT, RA, RB	DIVWU. 4,4,6
DIVWUO	DIVWUO RT, RA, RB	DIVWUO 4,4,6
DIVWUO.	DIVWUO. RT, RA, RB	DIVWUO. 4,4,6
DOZ	DOZ RT, RA, RB	DOZ 4,4,6
DOZ.	DOZ. RT, RA, RB	DOZ. 4,4,6
DOZO	DOZO RT, RA, RB	DOZO 4,4,6
DOZO.	DOZO. RT, RA, RB	DOZO. 4,4,6
DOZI	DOZI RT, RA, SI	DOZI 4,4,0X0
EQV	EQV RA, RS, RB	EQV 4,4,6
EQV.	EQV. RA, RS, RB	EQV. 4,4,6
EXTSB	EXTSB RA, RS	EXTSB 6,4
EXTSB.	EXTSB. RA, RS	EXTSB. 4,4
EXTSH	EXTSH RA, RS	EXTSH 4,6
EXTSH.	EXTSH. RA, RS	EXTSH. 6,4
FABS	FABS FRT, FRB	FABS 6,4
FABS.	FABS. FRT, FRB	FABS. 6,25
FADD	FADD FRT, FRA, FRB	FADD 6,4,5
FADD.	FADD. FRT, FRA, FRB	FADD. 6,4,25
FADDS	FADDS FRT, FRA, FRB	FADDS 6,4,5
FADDS.	FADDS. FRT, FRA, FRB	FADDS. 10,8,9
FCMPO	FCMPO BF, FRA, FRB	FCMPO 6,4,5
FCMPU	FCMPU BF, FRA, FRB	FCMPU 6,4,5
FCTIW	FCTIW FRT, FRB	FCTIW 5,6
FCTIW.	FCTIW. FRT, FRB	FCTIW. 5,6
FCTIWZ	FCTIWZ FRT, FRB	FCTIWZ 5,6
FCTIWZ.	FCTIWZ. FRT, FRB	FCTIWZ. 5,6
FDIV	FDIV FRT, FRA, FRB	FDIV 6,4,5
FDIV.	FDIV. FRT, FRA, FRB	FDIV. 6,4,5
FDIVS	FDIVS FRT, FRA, FRB	FDIVS 6,4,5
FDIVS.	FDIVS. FRT, FRA, FRB	FDIVS. 6,4,5
FMADD	FMADD FRT, FRA, FRC, FRB	FMADD 6,4,5,7
FMADD.	FMADD. FRT, FRA, FRC, FRB	FMADD. 6,4,5,7

ROZKAZ	SKŁADNIA	PRZYKŁAD
FMADDS	FMADDS FRT, FRA, FRC, FRB	FMADDS 6,4,5,7
FMADDS.	FMADDS. FRT, FRA, FRC, FRB	FMADDS. 6,4,5,7
FMR	FMR FRT, FRB	FMR 6,4
FMR.	FMR. FRT, FRB	FMR. 6,25
FMSUB	FMSUB FRT, FRA, FRC, FRB	FMSUB 6,4,5,7
FMSUB.	FMSUB. FRT, FRA, FRC, FRB	FMSUB. 6,4,5,7
FMSUBS	FMSUBS FRT, FRA, FRC, FRB	FMSUB. 6,4,5,7
FMSUBS.	FMSUBS. FRT, FRA, FRC, FRB	FMSUBS. 6,4,5,7
FMUL	FMUL FRT, FRA, FRC	FMUL 6,4,5
FMUL.	FMUL. FRT, FRA, FRC	FMUL. 6,4,25
FMULS	FMULS FRT, FRA, FRC	FMULS 14,12,13
FMULS.	FMULS. FRT, FRA, FRC	FMULS. 3,2,1
FNABS	FNABS FRT, FRB	FNABS 6,5
FNABS.	FNABS. FRT, FRB	FNABS. 6,4
FNEG	FNEG FRT, FRB	FNEG 6,5
FNEG.	FNEG. FRT, FRB	FNEG. 6,4
FNMADD	FNMADD FRT, FRA, FRC, FRB	FNMADD 6,4,5,7
FNMADD.	FNMADD. FRT, FRA, FRC, FRB	FNMADD. 6,4,5,7
FNMADDS	FNMADDS FRT, FRA, FRC, FRB	FNMADDS 6,4,5,7
FNMADDS.	FNMADDS. FRT, FRA, FRC, FRB	FNMADDS. 6,4,5,7
FNMSUB	FNMSUB FRT, FRA, FRC, FRB	FNMSUB 6,4,5,7
FNMSUB.	FNMSUB. FRT, FRA, FRC, FRB	FNMSUB. 6,4,5,7
FNMSUBS	FNMSUBS FRT, FRA, FRC, FRB	FNMSUBS 6,4,5,7
FNMSUBS.	FNMSUBS. FRT, FRA, FRC, FRB	FNMSUBS. 6,4,5,7
FRSP	FRSP FRT, FRB	FRSP 6,4
FRSP.	FRSP. FRT, FRB	FRSP. 6,4
FSUB	FSUB FRT, FRA, FRB	FSUB 6,4,5
FSUB.	FSUB. FRT, FRA, FRB	FSUB. 6,5,4
FSUBS	FSUBS FRT, FRA, FRB	FSUB 6,4,5
FSUBS.	FSUBS. FRT, FRA, FRB	FSUB. 6,5,4
ISYNC	ISYNC	ISYNC
LBZ	LBZ RT, D( RA)	LBZ 6,0X6(5)
LBZU	LBZU RT, D( RA)	LBZU 6,0X4(5)
LBZUX	LBZUX RT, RA, RB	LBZUX 6,5,4

ROZKAZ	SKŁADNIA	PRZYKŁAD
LBZX	LBZX RT, RA, RB	LBZX 6,5,4
LFD	LFD FRT, D( RA)	LFD 6,0X10(5)
LFDU	LFDU FRT, D( RA)	LFDU 6,0X24(5)
LFDUX	LFDUX FRT, RA, RB	LFDUX 6,5,4
LFDX	LFDX FRT, RA, RB	LFDX 6,0,4
LFS	LFS FRT, D( RA)	LFS 6,0X8(5)
LFSU	LFSU FRT, D( RA)	LFSU 6,0(5)
LFSUX	LFSUX FRT, RA, RB	LFSUX 6,5,4
LFSX	LFSX FRT, RA, RB	LFSX 6,0,4
LHA	LHA RT, D( RA)	LHA 6,0X13(5)
LHAU	LHAU RT, D( RA)	LHAU 6,0X20(5)
LHAUX	LHAUX RT, RA, RB	LHAUX 6,5,4
LHAX	LHAX RT, RA, RB	LHAX 6,5,4
LHBRX	LHBRX RT, RA, RB	LHBRX 6,5,4
LHZ	LHZ RT, D( RA)	LHZ 6,0X10(4)
LHZU	LHZU RT, D( RA)	LHZU 6,0(4)
LHZUX	LHZUX RT, RA, RB	LHZUX 6,5,4
LHZX	LHZX RT, RA, RB	LHZX 6,5,4
LMW	LMW RT, D( RA)	LMW 29,0(30)
LSCBX	LSCBX RT, RA, RB	LSCBX 6,5,4
LSCBX.	LSCBX. RT, RA, RB	LSCBX. 6,5,4
LSWI	LSWI RT, RA, NB	LSWI 6,7,0X6
LSWX	LSWX RT, RA, RB	LSWX 6,5,4
LWARX	LWARX RT, RA, RB	LWARX 5,0,3
LWBRX	LWBRX RT, RA, RB	LWBRX 6,4,5
LWZ	LWZ RT, D( RA)	LWZ 6,0X28(5)
LWZU	LWZU RT, D( RA)	LWZU 6,0X30(4)
LWZUX	LWZUX RT, RA, RB	LWZUX 6,5,4
LWZX	LWZX RT, RA, RB	LWZX 6,5,4
MASKG	MASKG RA, RS, RB	MASKG 6,5,4
MASKG.	MASKG. RA, RS, RB	MASKG. 6,5,4
MASKIR	MASKIR RA, RS, RB	MASKIR 6,5,4
MASKIR.	MASKIR. RA, RS, RB	MASKIR. 6,5,4
MCRF	MCRF BF, BFA	MCRF 2,3

ROZKAZ	SKŁADNIA	PRZYKŁAD
MCRFS	MCRFS BF, BFA	MCRFS 3,4
MCRXR	MCRXR BF	MCRXR 4
MFCR	MFCR RT	MFCR 6
MFFS	MFFS FRT	MFFS 14
MFFS.	MFFS. FRT	MFFS. 14
MFSPR	MFSPR RT, SPR	MFSPR 6,1
MTCRF	MTCRF FXM, RS	MTCRF 0X80,5
MTFSB0	MTFSB0 BT	MTFSB0 3
MTFSB0.	MTFSB0. BT	MTFSB0. 3
MTFSB1	MTFSB1 BT	MTFSB1 4
MTFSB1.	MTFSB1. BT	MTFSB1 3
MTFSF	MTFSF FLM, FRB	MTFSF 0X80,5
MTFSF.	MTFSF. FLM, FRB	MTFSF. 0XE0,5
MTFSFI	MTFSFI BF, I	MTFSFI 6,4
MTFSFI.	MTFSFI. BF, I	MTFSFI. 0,1
MUL	MUL RT, RA, RB	MUL 6,4,10
MUL.	MUL. RT, RA, RB	MUL. 6,4,10
MULO	MULO RT, RA, RB	MULO 6,4,10
MULO.	MULO. RT, RA, RB	MULO. 6,4,10
MULHW	MULHW RT, RA, RB	MULHW 6,4,10
MULHW.	MULHW. RT, RA, RB	MULHW. 6,4,10
MULHWU	MULHWU RT, RA, RB	MULHWU 6,4,10
MULHWU.	MULHWU. RT, RA, RB	MULHWU. 6,4,10
MULLI	MULLI RT, RA, SI	MULLI 6,4,10
MULLW	MULLW RT, RA, RB	MULLW 6,4,10
MULLW.	MULLW. RT, RA, RB	MULLW. 6,4,10
MULLWO	MULLWO RT, RA, RB	MULLWO 6,4,10
MULLWO.	MULLWO. RT, RA, RB	MULLWO. 6,4,10
NABS	NABS RT, RA	NABS 6,4
NABS.	NABS. RT, RA	NABS. 6,4
NABSO	NABSO RT, RA	NABSO 6,4
NABSO.	NABSO. RT, RA	NABSO. 6,4
NAND	NAND RA, RS, RB	NAND 6,4,7
NAND.	NAND. RA, RS, RB	NAND. 6,4,7

ROZKAZ	SKŁADNIA	PRZYKŁAD
NEG	NEG RT, RA	NEG 6,4
NEG.	NEG. RT, RA	NEG. 6,4
NEGO	NEGO RT, RA	NEGO 6,4
NEGO.	NEGO. RT, RA	NEGO. 6,4
NOR	NOR RA, RS, RB	NOR 6,4,7
NOR.	NOR. RA, RS, RB	NOR. 6,4,7
OR	OR RA, RS, RB	OR 6,4,7
OR.	OR. RA, RS, RB	OR. 6,4,7
ORC	ORC RA, RS, RB	ORC 6,4,7
ORC.	ORC. RA, RS, RB	ORC. 6,4,7
ORI	ORI RA, RS, UI	ORI 6,4,0X0079
ORIS	ORIS RA, RS, UI	ORIS 6,4,0X0079
RLWIMI	RLWIMI RA, RS, SH, MB, ME	RLWIMI 6,4,2,0,0X1D
RLWIMI.	RLWIMI. RA, RS, SH, MB, ME	RLWIMI. 6,4,2,0,0X1A
RLWINM	RLWINM RA,RS,SH,MB,ME	RLWINM 6,4,2,0,0X1D
RLWINM.	RLWINM. RA,RS,SH,MB,ME	RLWINM. 6,4,2,0,0X1D
RLWNM	RLWNM RA,RS,RB,MB,ME	RLWNM 6,4,5,0,0X1D
RLWNM.	RLWNM. RA,RS,RB,MB,ME	RLWNM. 6,4,5,0,0X1D
RRIB	RRIB RA, RS, RB	RRIB 4,5,6
RRIB.	RRIB. RA, RS, RB	RRIB. 4,5,6
SLE	SLE RA, RS, RB	SLE 6,4,5
SLE.	SLE. RA, RS, RB	SLE. 6,4,5
SLEQ	SLEQ RA, RS, RB	SLEQ 6,4,5
SLEQ.	SLEQ. RA, RS, RB	SLEQ. 6,4,5
SLIQ	SLIQ RA, RS, SH	SLIQ 6,4,0X14
SLIQ.	SLIQ. RA, RS, SH	SLIQ. 6,4,0X10
SLLIQ	SLLIQ RA, RS, SH	SLLIQ 6,4,0X3
SLLIQ.	SLLIQ. RA, RS, SH	SLLIQ. 6,4,0X4
SLLQ	SLLQ RA, RS, RB	SLLQ 6,4,5
SLLQ.	SLLQ. RA, RS, RB	SLLQ. 6,4,5
SLQ	SLQ RA, RS, RB	SLQ 6,4,5
SLQ.	SLQ. RA, RS, RB	SLQ. 6,4,5
SLW	SLW RA, RS, RB	SLW 6,4,5
SLW.	SLW. RA, RS, RB	SLW. 6,4,5

ROZKAZ	SKŁADNIA	PRZYKŁAD
SRAIQ	SRAIQ RA, RS, SH	SRAIQ 6,4,0X4
SRAIQ.	SRAIQ. RA, RS, SH	SRAIQ. 6,4,0X4
SRAQ	SRAQ RA, RS, RB	SRAQ 6,4,7
SRAQ.	SRAQ. RA, RS, RB	SRAQ. 6,4,7
SRAW	SRAW RA, RS, RB	SRAW 6,4,5
SRAW.	SRAW. RA, RS, RB	SRAW. 6,4,5
SRAWI	SRAWI RA, RS, SH	SRAWI 6,4,0X4
SRAWI.	SRAWI. RA, RS, SH	SRAWI. 6,4,0X4
SRE	SRE RA, RS, RB	SRE 6,4,5
SRE.	SRE. RA, RS, RB	SRE. 6,4,5
SREA	SREA RA, RS, RB	SREA 6,4,7
SREA.	SREA. RA, RS, RB	SREA. 6,4,7
SREQ	SREQ RA, RS, RB	SREQ 6,4,7
SREQ.	SREQ. RA, RS, RB	SREQ. 6,4,18
SRIQ	SRIQ RA, RS, SH	SRIQ 6,4,0XC
SRIQ.	SRIQ. RA, RS, SH	SRIQ. 6,4,0X14
SRLIQ	SRLIQ RA, RS, SH	SRLIQ 6,4,0X4
SRLIQ.	SRLIQ. RA, RS, SH	SRLIQ. 6,4,0X4
SRLQ	SRLQ RA, RS, RB	SRLQ 6,4,8
SRLQ.	SRLQ. RA, RS, RB	SRLQ. 6,4,8
SRQ	SRQ RA, RS, RB	SRQ 6,4,25
SRQ.	SRQ. RA, RS, RB	SRQ. 6,4,25
SRW	SRW RA, RS, RB	SRW 6,4,5
SRW.	SRW. RA, RS, RB	SRW. 6,4,5
STB	STB RS, D( RA)	STB 6,0X28(4)
STBU	STBU RS, D( RA)	STBU 6,0X40(16)
STBUX	STBUX RS, RA, RB	STBUX 6,4,19
STBX	STBX RS, RA, RB	STBX 6,0,4
STFD	STFD FRS, D( RA)	STFD 6,0X20(4)
STFDU	STFDU FRS, D( RA)	STFDU 6,0X28(4)
STFDUX	STFDUX FRS, RA, RB	STFDUX 6,4,5
STFDX	STFDX FRS, RA, RB	STFDX 6,4,5
STFS	STFS FRS, D( RA)	STFS 6,0X8(4)
STFSU	STFSU FRS, D( RA)	STFSU 6,0X50(4)

ROZKAZ	SKŁADNIA	PRZYKŁAD
STFSUX	STFSUX FRS, RA, RB	STFSUX 6,5,4
STFSX	STFSX FRS, RA, RB	STFSX 6,0,4
STH	STH RS, D( RA)	STH 6,0X8(4)
STHBRX	STHBRX RS, RA, RB	STHBRX 6,0,4
STHU	STHU RS, D( RA)	STHU 6,0X8(4)
STHUX	STHUX RS, RA, RB	STHUX 6,4,5
STHX	STHX RS, RA, RB	STHX 6,0,5
STMW	STMW RS, D( RA)	STMW 29,0X20(4)
STSWI	STSWI RS, RA, NB	STSWI 6,4,12
STSWX	STSWX RS, RA, RB	STSWX 6,4,5
STW	STW RS, D( RA)	STW 6,4(5)
STWBRX	STWBRX RS, RA, RB	STWBRX 6,4,9
STWCX.	STWCX. RS, RA, RB	STWCX. 4,0,3
STWU	STWU RS, D( RA)	STWU 6,0X10(4)
STWUX	STWUX RS, RA, RB	STWUX 6,4,23
STWX	STWX RS, RA, RB	STWX 6,0,4
SUBF	SUBF RT, RA, RB	SUBF 6,4,10
SUBF.	SUBF. RT, RA, RB	SUBF. 6,4,10
SUBFO	SUBFO RT, RA, RB	SUBFO 6,4,10
SUBFO.	SUBFO. RT, RA, RB	SUBFO. 6,4,10
SUBFC	SUBFC RT, RA, RB	SUBFC 6,4,10
SUBFC.	SUBFC. RT, RA, RB	SUBFC. 6,4,10
SUBFCO	SUBFCO RT, RA, RB	SUBFCO 6,4,10
SUBFCO.	SUBFCO. RT, RA, RB	SUBFCO. 6,4,10
SUBFE	SUBFE RT, RA, RB	SUBFE 6,4,10
SUBFE.	SUBFE. RT, RA, RB	SUBFE. 6,4,10
SUBFEO	SUBFEO RT, RA, RB	SUBFEO 6,4,10
SUBFEO.	SUBFEO. RT, RA, RB	SUBFEO. 6,4,10
SUBFIC	SUBFIC RT, RA, SI	SUBFIC 6,4,0X00007000
SUBFME	SUBFME RT, RA	SUBFME 6,4
SUBFME.	SUBFME. RT, RA	SUBFME. 6,4
SUBFMEO	SUBFMEO RT, RA	SUBFMEO 6,4
SUBFMEO.	SUBFMEO. RT, RA	SUBFMEO. 6,4
SUBFZE	SUBFZE RT, RA	SUBFZE 6,4



ROZKAZ	SKŁADNIA	PRZYKŁAD
SUBFZE.	SUBFZE. RT, RA	SUBFZE. 6,4
SUBFZEO	SUBFZEO RT, RA	SUBFZEO 6,4
SUBFZEO.	SUBFZEO. RT, RA	SUBFZEO. 6,4
SYNC	SYNC	SYNC
XOR	XOR RA, RS, RB	XOR 6,4,7
XOR.	XOR. RA, RS, RB	XOR. 6,4,7
XORIS	XORIS RA, RS, UI	XORIS 6,4,0X0079

## Literatura

1. *AIX 5L for POWER-based Systems. Assembler Language Reference*, Second Edition (April 2001), IBM Corp.
2. *PowerPC™ 601 RISC Microprocessor User's Manual*, MPC601UM/AD, Motorola Inc. 1995
3. *PowerPC® Microprocessor Family: The Programming Environments Manual for 64-bit Microprocessors*, Version 3.0, July 15, 2005, IBM Systems and Technology Group
4. Zdzisław Pogoda, *Mikroprocesory RISC rodziny PowerPC: mikroprocesor PowerPC 601*, Gliwice 1995, ISBN 83-86644-02-8