

**Akademia Górniczo-Hutnicza  
im. Stanisława Staszica w Krakowie**

---

Wydział Elektrotechniki, Automatyki, Informatyki i Inżynierii Biomedycznej

KATEDRA INFORMATYKI STOSOWANEJ



**PRACA MAGISTERSKA**

**RAFAŁ STUDNICKI**

**PODSTAWOWA FUNKCJONALNOŚĆ ERLANGA DLA SYSTEMU  
FREERTOS**

PROMOTOR:  
dr inż. Piotr Matyasik

Kraków 2014

## **OŚWIADCZENIE AUTORA PRACY**

OŚWIADCZAM, ŚWIADOMY ODPOWIEDZIALNOŚCI KARNEJ ZA POŚWIADCZENIE NIEPRAWDY, ŻE NINIEJSZĄ PRACĘ DYPLOMOWĄ WYKONAŁEM OSOBIŚCIE I SAMODZIELNIE, I NIE KORZYSTAŁEM ZE ŹRÓDEŁ INNYCH NIŻ WYMIENIONE W PRACY.

.....

PODPIS

**AGH**  
**University of Science and Technology in Krakow**

---

Faculty of Electrical Engineering, Automatics, Computer Science and Engineering in  
Biomedicine

DEPARTMENT OF APPLIED COMPUTER SCIENCE



**MASTER OF SCIENCE THESIS**

**RAFAŁ STUDNICKI**

**IMPLEMENTATION OF BASIC FEATURES OF ERLANG FOR  
FREERTOS**

SUPERVISOR:  
Piotr Matyasik Ph.D

Krakow 2014



## Spis treści

<b>A. Lista instrukcji maszyny wirtualnej BEAM</b> .....	6
A.1. Typy argumentów .....	6
A.2. Lista instrukcji .....	8
<b>Bibliografia</b> .....	10

## A. Lista instrukcji maszyny wirtualnej BEAM

Dodatek zawiera listę instrukcji maszyny wirtualnej BEAM, jakie może zawierać skompilowany kod pośredni przez nią wykonywany. Lista zawiera nazwę operacji, jej argumenty oraz opis jej działania.

Kod operacji zajmuje zawsze 1 bajt w pliku ze skompilowanym kodem pośrednim modułu.

Argumenty mogą zajmować więcej, zgodnie z opisem w sekcji A.1.

Kolejność bajtów w zapisie kodu pośredniego to *big endian*.

### A.1. Typy argumentów

Każdy z tagów jest możliwy do zapisania przy użyciu 3 bitów. Jednak w kodowaniu binarnym do zapisu typu używane są dodatkowo 1 lub 2 bity. Dzięki nim możliwe jest rozróżnienie pomiędzy argumentami zapisanymi przy użyciu różnej liczby bajtów.

Tag		Typ
binarnie	dziesiętnie	
000	0	uniwersalny indeks, np. do tablicy stałych
001	1	liczba całkowita
010	2	indeks do tablicy atomów
011	3	numer rejestru X maszyny wirtualnej
100	4	numer rejestru Y maszyny wirtualnej
101	5	etykieta, używana w funkcjach skoku
111	7	złożone wyrażenie (np. lista, liczba zmiennoprzecinkowa)

Tablica A.1: Tagi typów danych w pliku ze skompilowanym modułem

Jeżeli tagowana liczba jest nieujemna, mniejsza od 16 (możliwe jest zapisanie jej przy użyciu 4 bitów) to argument jest zapisany przy użyciu jednego bajtu a jego postać binarna to:

$$X_1X_2X_3X_4\mathbf{0TTT}_{(2)},$$

gdzie  $X_1X_2X_3X_4_{(2)}$  to tagowana liczba,  $X_1$  jest jej najbardziej znaczącym bitem, a  $\mathbf{TTT}_{(2)}$  to tag danego typu argumentu.

Na przykład, atom, który w tablicy atomów modułu ma indeks  $2_{10} = 10_2$ , po zakodowaniu będzie miał postać:

$$00100010_2 = 22_{16} = 34_{10}.$$

W przypadku, gdy liczba jest nieujemna, mniejsza lub równa 16, a mniejsza od 2048 (możliwe jest jej zapisanie przy użyciu 11 bitów), argument jest zapisany przy użyciu dwóch bajtów, których postać binarna to:

$$X_1X_2X_301TTT X_4X_5X_6X_7X_8X_9X_{10}X_{11(2)},$$

gdzie  $X_1...X_{11(2)}$  to tagowana liczba,  $X_1$  jest jej najbardziej znaczącym bitem, a  $TTT_{(2)}$  to tag danego typu argumentu.

Na przykład, liczba całkowita  $565_{10} = 010\ 00110101_2$  po zakodowaniu będzie miała postać:

$$01001001\ 00110101_2 = 4935_{16} = 18741_{10}.$$

Jeżeli argument jest liczbą ujemną lub dodatnią wymagającą w zapisie dwójkowym więcej niż 11 bitów to liczba taka zapisywana jest binarnie w kodzie uzupełnień do dwóch poprzedzona odpowiednim nagłówkiem.

Jeżeli zakodowaną liczbę można zapisać na nie więcej niż 8 bajtach, to nagłówek ma następującą postać:

$$N_1N_2N_311TTT_{(2)},$$

gdzie  $N_1N_2N_3_{(2)}$  to rozmiar argumentu w bajtach pomniejszony o 2 (jeżeli argument jest liczbą ujemną zajmującą 1 bajt to powinien on zostać dopełniony do 2 bajtów),  $N_1$  jest jego najbardziej znaczącym bitem, a  $TTT_{(2)}$  to tag danego typu argumentu.

Na przykład, aby zapisać na dwóch bajtach liczbę  $-21_{10} = 11111111\ 11101011_{U2}$ , jej postać binarną należy poprzedzić nagłówkiem:

$$00011001_2 = 19_{16} = 25_{10}.$$

Jeżeli do zapisania liczby w kodzie uzupełnień do dwóch potrzeba przynajmniej 9 bajtów, wtedy nagłówek ma postać:

$$11111TTT N_1N_2N_3N_40000_{(2)},$$

gdzie  $N_1N_2N_3N_4_{(2)}$  to rozmiar argumentu w bajtach pomniejszony o 9,  $N_1$  jest jego najbardziej znaczącym bitem, a  $TTT_{(2)}$  to tag danego typu argumentu.

Na przykład, w celu zapisania liczby  $2^{(15 \times 8) - 1} - 1$  na 15 bajtach, należy zapis tej liczby w kodzie U2 poprzedzić następującym nagłówkiem:

$$11111001\ 01100000_2 = F960_{16} = 63840_{10}.$$

## **A.2. Lista instrukcji**



Kod operacji		Nazwa operacji i jej argumenty	Opis operacji i uwagi
szesnastkowo	dziesiętnie		
01	1	label Lbl	Wprowadza lokalną dla danego modułu etykietę identyfikującą aktualne miejsce w kodzie.
02	2	func_info M F A	Definiuje funkcję F, w module M o arności A.
03	3	int_code_end	???

Tablica A.2: Lista operacji maszyny wirtualnej BEAM

## Bibliografia

- [1] Ericsson AB. Erlang External Term Format. [http://erlang.org/doc/apps/erts/erl\\_ext\\_dist.html](http://erlang.org/doc/apps/erts/erl_ext_dist.html), 2014. [data dostępu: 21.03.2014].
- [2] J. Morrison. EA IFF 85: Standard for interchange format files. *Amiga ROM Kernel Reference Manual: Devices (3rd edition)*, Addison-Wesley, 1(99):1, 1985.