

**Харченко К.В.**

ННК “Інститут прикладного системного аналізу” НТУУ “КПІ”, Київ, Україна

## Розширення віртуальної машини LLVM паралельними інструкціями для реалізації системи передачі повідомлень

Розглянуто архітектуру побудови паралельної системи з віртуальною машиною LLVM з використанням парадигми передачі повідомлень.

Розробка віртуальної машини Low Level Virtual Machine [1,2] почалась у 2000 році в Університеті Іллінойсу. Проста та ефективна реалізація віртуальної машини з Intermediate Representation коду програми дозволяє широко використовувати код на процесорах X86, MIPS, ARM та багатьох інших архітектурах. JIT-компіляція реалізує перехід від проміжного набору команд до оптимізованого машинного коду. Для організації системи паралельних обчислень з парадигмою передачі повідомлень між паралельними процесорами [3, 4] необхідно забезпечити такі базові функції:

- встановлення зв'язку з паралельною віртуальною машиною.
- запуск задачі (функції) на паралельній віртуальній машині.
- передача повідомлення до паралельної та від паралельної машини.
- завершення задачі на паралельній віртуальній машині.

Таким чином, якщо в LLVM на рівні байт-коду реалізувати інструкції для вказаних функцій, тоді організація віртуального паралельного середовища буде мати вигляд (Рис. 1).

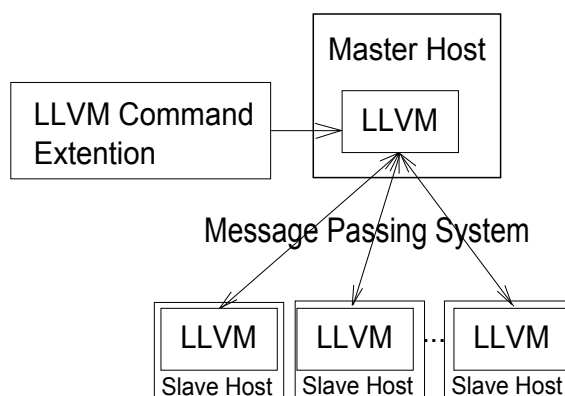


Рис. 1. Архітектура паралельної віртуальної машини на базі LLVM

Порівняно з парадигмою *shared memory*, запропонована реалізація середовища буде мати такі переваги:

- легку реалізацію розширення команд LLVM і реалізацію на рівні ядра віртуальної машини.
- ефективну систему авторизації запуску задач.
- легку сумісність за типами даних гетерогенних систем у зв'язку з невеликою кількістю типів даних у LLVM.
- можливість використання у мобільних та побутових платформах з різними типами процесорів для вирішення задач перекодування аудіо- та відео-потоків, т.ін.
- чітку організацію архітектури програмного коду для паралельних обчислень у LLVM.
- більші можливості для реалізації паралельних обчислень порівняно реалізованою системою SIMD в LLVM [5].

**Література.** 1. LLVM, Wikipedia:[http://ru.wikipedia.org/wiki/Low\\_Level\\_Virtual\\_Machine](http://ru.wikipedia.org/wiki/Low_Level_Virtual_Machine)  
 2. LLVM: <http://llvm.org/> 3. Message Passing Interface, Wikipedia:[http://ru.wikipedia.org/wiki/Message\\_Passing\\_Interface](http://ru.wikipedia.org/wiki/Message_Passing_Interface) 4. PVM: <http://www.csm.ornl.gov/pvm/> 5. LLVM, Generic Vector Support:<http://llvm.org/releases/1.7/docs/ReleaseNotes.html>