Outline of the Personal Sequential Inference Machine: PSI

Shunichi UCHIDA, Minoru YOKOTA, Akira YAMAMOTO, Kazuo TAKI, and Hiroshi NISHIKAWA
ICOT Research Center, Institute for New Generation Computer Technology, Mita Kokusai Bldg. 21 F, 4-28 Mita 1-chome, Minato-ku, Tokyo 108

Abstract
PSI is a personal computer system being developed as a tool for providing researchers with an efficient programming environment. It directly supports logic programming language, KL0 (Fifth Generation Kernel Language, Version 0), with firmware and hardware.

Its interpreter is implemented in the firmware and several hardware mechanisms are provided to attain almost the same level of performance as the DEC-10 Prolog on DEC2060. It also provides the user with a large memory space, 40 bits × 2 to 16 MW, which is essential for developing actual application programs like an expert system.

To make efficient man-machine interaction possible, such input and output devices as bit-map display, pointing device and key-board are provided. A local area network is also being developed to build a distributed system.

§1 Introduction
In the fifth generation computer project, a variety of software and hardware tools is planned to be developed. One of the most important tools is the sequential inference machine (SIM). SIM is considered a personal computer supporting logic programming language KL0 (Fifth Generation Kernel Language, Version 0).1)

Several models will be planned for SIM in the future. The first experimental model is named PSI (Personal Sequential Inference machine). It is under development and is planned to complete firmware, hardware and the subset of its operating system in the initial stage of the project.2,3)

PSI is designed to be a medium performance personal machine which can attain about 20 K-LIPS (Logical Inference Per Second). It provides large memory space (40 bits × 2 to 16 MW) which is essential for developing experimental application programs like an expert system.

To make man-machine interaction efficient, input and output devices such as bit-map display, pointing device and key-board are provided. A local area network is also being developed to build a distributed system.4)
The outline of the PSI architecture and hardware system is described in this paper.

§2 Design of the Machine Architecture

Machine language of PSI is called KL0. It is a Prolog-like language, however, it is intended to be used in writing the operating system. Thus, a variety of built-in predicates is added in KL0 to describe the kernel of the operating system and device control.

The internal representation of KL0 is obtained by compiling user-described KL0 programs. Optimization techniques, such as tail recursion optimization are made by the compiler. The KL0 interpreter, which interprets internal representation is implemented in firmware. The architecture of PSI is primarily designed to support the interpretation efficiently.

After considering the experience of using DEC-10 Prolog on DEC2060, for performance and memory space we estimate the first experimental model of PSI should attain more than 10 KLIPS in computing speed and should have more than 1MW for main memory.

The machine is designed to attain about 20 KLIPS and is implemented by TTL ICs. It employs tag architecture and microprogram control. Each data cell in the main memory consists of 8 bit tag and 32 bit data fields. Logical address is specified by 32 bits. Logical address space is divided into 256 areas by 8 bits. Each area can be used as an independent stack or a heap area. Maximum size of the area is 16 MW (24 bits).

The operating system of PSI is planned to implement a single-user multi-process support mainly for experiments of intelligent man-machine interface and interactive use of the machine.

Furthermore, it is required that devices such as bit-map display and pointing device can be smoothly controlled by KL0 programs. KL0 language includes several primitives to handle interrupts, exceptions, process creation, deletion and synchronization.

Thus, architectural support is essential. This includes support for process switching, separation of the address space assigned for each process, input/output control and so forth.

The main design characteristics of PSI are summarized as follows:

- Firmware implementation of KL0 interpreter and kernel functions of operating system.
- Partial architectural support of process switching to implement a single-user multi-process system.
- Partial hardware support of important functions for KL0 interpretation, such as unification and resolution by special hardware registers and stack manipulation mechanisms.
- Employment of a cache to improve memory access speed.
- A logical to physical address translation mechanism to implement multiple virtual stacks in main memory.
- Hardware and firmware support of interrupt handling and mode switching for efficient control of various input and output devices.
- Employment of a standard bus (IEEE-796 bus) for peripheral devices. Also, local area network support is planned to be implemented.
- Ease of extension for functional units such as an arithmetic unit including