

Development of General-Purpose Internetworking Unit*



Synopsis:

A general purpose internetworking platform which permits the interconnection of any set of networks has been developed by Kawasaki Steel. The platform consists of several network interfaces with an OS. The OS is designed to realize concurrent processing of protocols by a dynamic

equipments and networks which is referred to as inter-

simultaneously with communication

networking, and management of this internetworking system, the authors began development of a general-purpose internetworking unit.

From the viewpoint of business development, the following two features were required in this unit:

(1) Must contribute to timely commercialization in line with business development.

(3) Loading of an internetworking OS as a means of solving interprocess communication/synchronization, exclusion control, and other problems peculiar to concurrent processing, which are techniques forming the basis of (1) and (2).

(4) High-level protocol generating language for improving the efficiency of protocol processing program

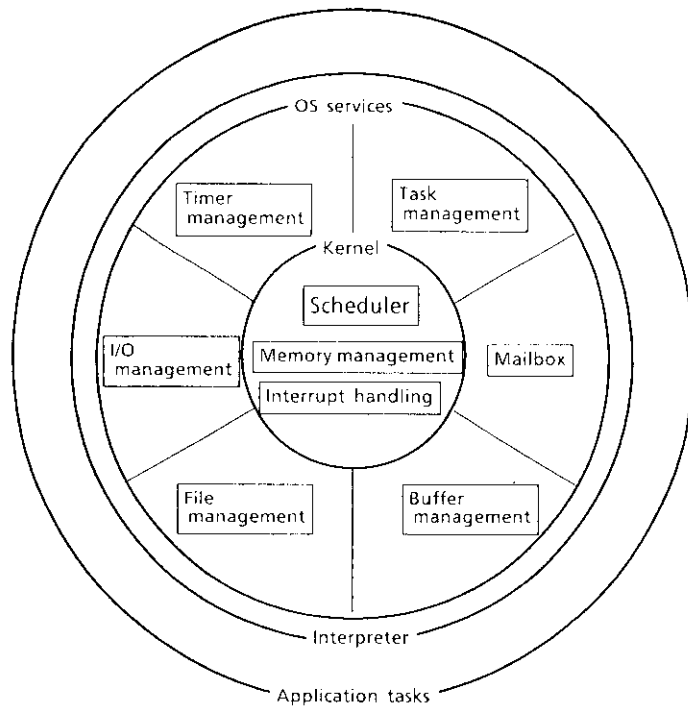


Fig. 2 Internetworking-OS functional structure

A scheduler is provided in the kernel portion to process multiple tasks. OS services for managing resources such as the buffer and timer run around the

Data is physically received by a communication LSI (external processing) and received data is accepted and forwarded to the forward-processing in case of

scheduler. Application tasks run on this OS, but OS services are provided for protocol tasks through a

receiving completion interrupt from the communi-

cation LSI (internal processing).
(2) Forward Processing

Because protocol processing normally has external relations with the transmission media and destination

The received data is encrypted and its integrity is

tion right. The task which has abandoned the execution right transfers to the ready state and waits to acquire

execution pointer; thus, only simple processing is needed in preparation for task switching.

right is not transferred to another task unless the task being executed abandons its right

6. Concurrent Processing of Protocol Process Tasks

Section 6 describes how efficient concurrent processing of protocol tasks is realized on the task

6.1 Event-Waiting Operation

CPU

LSI for communication

transition in waiting for events in the dynamic task

the most case, the operation of the system can be