TO: MSPM Distribution
FROM: J. H. Saltzer
SUBJ: BK.1.02
DATE: March 22, 1967

BK.1.02, describing the process data block, is revised to include PL/I declarations, and to bring it up to date.

Published: 03/10/67
(Supersedes: BK.1.02, 06/02/67)

Identification

The Processor Data Block
J. H. Saltzer, R. L. Rappaport

Purpose

Each processor in the system has its own block of storage known as the Processor Data Block. The processor data block of a processor is maintained by the Traffic Controller, so it must be in wired-down core memory. The processor data block of a processor is contained in the Processor Data Segment (see Section BK.1.01) of this processor. Since the segment descriptor word of the processor data segment is passed along from process to process, the processor data block of a processor is always accessible to the process currently running on this processor.

Contents of the Processor Data Block

Each processor data block has an identical structure. The following items are contained in each processor data block:

1. Processor Serial Number. This number is needed for Test and Diagnostic purposes (identifying an unhealthy processor) and for creating unique identifiers. This number is not the 3-bit processor hardware tag but rather an assigned serial number obtained by looking up the three-bit hardware tag in the configuration table when the processor entered the system.

2. Process Currently running. This is the identification of the process which now has control of the processor.

3. Current Process Index. An index into the Active Process Table which points to the Active Process Table entry for the process currently running on this processor.

4. Time last seen. This is the calendar time at which the processor was last in the Traffic Controller.

5. Initial Execution Meter Value. This is the contents of the execution meter register the last time that the processor was in the Traffic Controller.

6. Processor Index Number. The three-bit processor number, used for indexing into the processor communication table.

7. Processor Hardware Tag. The three-bit hardware identification tag built into the 645 processor.

8. Account Currently Paying. An index into the Active Meter Table pointing to the Active Meter Table entry for the account currently being charged for processor usage.

9. Drain Switch. This switch, if on, indicates that process interrupts are being "drained" on this processor (see Section BJ.6). All process interrupt handlers routinely check the drain switch of the interrupted processor, and take appropriate action whenever this switch is on.

10. Segment Number of Processor Data Segment. The processor data segment is known by the same segment number in every process. This item allows various system modules to ascertain this number easily.

11. Execution Meter Runout Fault Count. This item is incremented by one each time an execution meter runout fault occurs on this processor. It is reset to zero each time that processor usage is accounted for on this processor. It effectively gives 36 more bits (36 bits equal one word) of precision to the execution meter register conceptually transforming it from a 24 bit register into a 60 bit register (24 plus 36).

12. Process Loading Idle Cycles Account. An index into the Active Meter Table pointing to the entry for the account that should be charged for processor usage expended looping in swap_dbr and getwork while trying to load processes (see Section BJ.5.01). If swap_dbr is unsuccessful in loading a process, it sets the index of the Account Currently Paying (item 8) to be this number. In this way processor usage expended in unsuccessful loading can be accurately metered.

13. Empty Ready List Idle Process. The process identification of an "idle" process to which control is given whenever this processor, executing in getwork, finds the ready list empty. That is, when getwork (see Section BJ.4.02) discovers an empty ready list, it calls subroutine readyhim (see Section BJ.5.02) in order to reschedule the process indicated by this item. In this way getwork is never faced with the problem of not having anything to do and processor usage, expended in looping while waiting for useful work, can be accurately metered.

PL/I Declarations

All the above items are entries in the processor data block. The item names and their PL/I declarations are given below.

1.	prds$processor_no	bit(18)
2.	prds$processid	bit(70)
3.	prds$apt_index	fixed binary(17)
4.	prds$time	bit(72)
5.	prds$last_meter	bit(24)
6.	prds$processor_index	bit(3)
7.	prds$processor_tag	bit(3)
8.	prds$current_account	fixed binary(17)
9.	prds$drain_switch	bit(1)
10.	prds$segno	bit(18)
11.	prds$fault_count	bit(36)
12.	prds$idle_account	fixed binary(17)
13.	prds$idle_processid	fixed binary(17)