Pictures on Lectures

by J. H. Saltzer
address

block word

map

\[ x \]

base address

---

5 720

CP

base

map

\[ \begin{array}{c}
1 \\
2 \\
3 \\
4 \\
5 \\
\end{array} \]

\[ \begin{array}{c}
x \\
172 \\
\end{array} \]

core

172 Block

not in

supervisor

segment mechanism

user-A CP \[ \rightarrow \]

\[ S_n \]

\[ w_n \]

Page

PT

Seg. Descriptor list

absolute address of page-map of the data

user-B CP

SDL
Two-dimensional address space

word \# $2^n$

Seg \# 0 1 2 3 4

Indirect address

LDA alpha, *

alpha → Seg \# word \# ITS flag

indirect to segment

user \#1

CPU

CP scheduler

Supervisor

0~6

& data

user \#2

CPU

err

missing page fault
incremental dump

Multi-level Algorithm

Drum

every 24 hrs.
explicitly

1 week
1 month

Disk

down
up

Tape

rarely used

retrieve it from incremental tape
A. 7094 Salvage

B. Hierarchy Design

Pseud Proc.  
supervisor seg.

Private  
2-dim. memory
Situation:
Block - to stop process

Single:
Performs a computation
Type out results

Two process:
"A"
Performs a computation
Place result in Work Queue for B

"B"
Get next result
Type out results
Block
empty
"A"

Performs
Place
-call Wake up (B)

"B"

Get
Block
Type
-empty

T1 Bis running.

Calls to wake-up are ignored.

"A"

Queue full
Put flag in Queue
Block

Perform
Place
Call
-segment common to A & B

"B"

Get next result
Look for flag from A
Type output result

users program
Compiler

Typewriter

Converts Codes to ASCII
Canonical form
Convention
Erase and Kill editing

Escape Processing
ASCII control characters
Back space
carriage return
New line
Tab.
-space
CPU

MC

32 interrupt cells

MC

32 cells

MC

32 cells

Drum Controller

3 interrupts

GIOC

8 interrupts

3 interrupts by CPU

Interrupt Vector

instruction pair

SCU store control unit

TRA transfer

XED

execute double

System clock (Calendar)

(Calendar)

52-bit

1/μsec

alarm clock register

interrupt on Calendar ≥ Alarm
Address Space Switching

relational register

1000

CPU

25

ILC

interval timer

Memory

0

1000

1100

1200

1300

1400

1500

1600

1700

1800

1900

2000

2100

2200

2300

2400

2500

LRR load relational register

as if in location 0

absolute program

Process A

descriptor base register

CPU

desc. seg.

Process B

desc. seg.

call swap-DBR

LDBR

(x)←

(segment 10)

location 155

share some segment

Swap-

S # 10

loc. 156

return

call swap-DBR
### Process Table & stack

<table>
<thead>
<tr>
<th>Process #</th>
<th>DBR value</th>
<th>Stack pointer</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### System clock

![Diagram of System Control and CPs](image)
Compute a priority label and time limit.

Switching to address space of a process being scheduled.

Wake up C

Delete

ASCII conversion -> Canonical conv. -> Erase & kill

User Program

Pre-emption is ordered by Traffic Controller.

Running calls Block

Pre-emption

Give to Top of Ready list

Quit called by another process called wake up

Blocked

Ready

A-4  電気試験所（研D） 昭41.3.50,000
Quit ordered by some user to halt execution

Basic File System → Block → Traffic Controller → Known Process Table
→ Paged

Traffic Controller

Quit → Block → Wake up → Restart

Process Exchange

System Interrupt Interceptor

Interrupt Handler

1. write

Data Control words for GIOC-channel
Process "X"

T.C. wake up 151

Control is guaranteed to return interrupted process after external interrupt

I/O alarm clock

System Interrupt Interceptor priority classes inhibit word

System Interrupt

Save Processor State

Mask interrupts of equal or lower priority

Call handler for this interrupt

Restore Processor State (including Mask)

Return to user
Limited Core Memory

<table>
<thead>
<tr>
<th>Tables</th>
<th>entries</th>
<th>usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Process table</td>
<td>1 / process</td>
<td>Multi-plexing</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Data</td>
</tr>
<tr>
<td>Active Segment</td>
<td>segment / page</td>
<td>Contain</td>
</tr>
<tr>
<td>table</td>
<td>table in-core</td>
<td>Page location</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Information</td>
</tr>
</tbody>
</table>

Private Segments

<table>
<thead>
<tr>
<th>1 / process</th>
<th>usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Segment name table</td>
<td>Contains names correspond to segment numbers</td>
</tr>
<tr>
<td>Descriptor segment</td>
<td>hardware map</td>
</tr>
<tr>
<td>Process data segment</td>
<td>Store processor machine conditions</td>
</tr>
</tbody>
</table>

Diagram:
- **Descriptor Segment**
- **PTW**
- **Active Segment Table**
machine conditions of non-running process

Block

Contents of Call Stack
must be preserved

Get work

Swap-DBR

Process Segment

Process Data Segment

Current Procedure

When in T.C. or B.F.S.

Process Table

T.C.
B.F.S
P.T.
A.S.T
S.N.T
P.D.S
C.P

File System

Traffic Controller
Rings of Protection

descriptor segment

access bit

Pointer → PageTable

Central Processor

current ring register
Ring simulator

descriptor

segments, 

V-ring

DBR

access bit

read & write

Proc

1

2 1 0

Data

A:S

DMP

WP

Account name

Person — Project — Account

personal directory

project directory

account directory

Account

Data

Segment
Account Data Segment

Secondary storage charge
Total words on secondary
time list charged
2 days ago
alpha link

linkage information
for variable 'C'
in segment 'a'

information for
XYZ in XYZ

symbolic name of segment

entry point
(word #)

---

stepping stone

call create_process (stepping_stone, linker, search, ID #)

create process

Process Control

Process Table

BHS

TC

item 2 & 3
Segment "n"

\[ SB = n \]

SP

\[ × \]

\[ ≠ \]

\[ 32 \]

next SP

Temporary storage area for variables

\[ A \]

\[ XYZ : \text{Save} \]

\[ XYZ \]

Return

Process A

Address space of B

\[ SB \]

\[ SB \rightarrow \]

\[ SB \rightarrow \]

\[ SP \rightarrow \]

\[ SP \rightarrow \]

Alternative File System Approach:

missing \{ page, segment \} faults

Map \rightarrow Seg. Control

read (file name, Record#, loc.)

v Read

Information Storage Hierarchy

write

Buffer control

read(file, loc.)

delete

BFS

hardware
Present File System Approach

Directory

<table>
<thead>
<tr>
<th>name</th>
<th>access list</th>
<th>physical location</th>
<th>device name</th>
<th>location on device</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td></td>
<td>DIM</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td></td>
<td>DIM</td>
<td></td>
</tr>
</tbody>
</table>