Published: 10/13/66
(Supersedes: BE.8.02, 6/28/66)

## Identification

Page Management for the Pseudo-supervisor
D.H. Johnson

## Purpose

Basic procedures are described which implement page management without secondary storage and page turning.

## Introduction

A 6.36 or 64.5 process occupies the entire 645 simulated memory. The simulation system loader (MSPM section BE.7.07) loads all user and system segments and then allocates the remaining memory to the stack segment. The stack may be paged or unpaged with the option initially set by the user explicitly or by default. The loader may change the paging setting for the stack if it has difficulty fitting the stack into memory.

Since no secondary storage and page turning procedure exists yet in the pseudo system, a process is limited in size to the current 645 memory length. The entire 645 memory is allocated to a process so some segment must shrink in order that pages of memory may be allocated to other segments. The stack segment has been selected to be reduced as pages of memory are requested. A list of free pages will be maintained and updated whenever pages are requested and released. If the list cannot satisfy a request, unused pages will be truncated from the stack segment. Pertinent information for the stack will be revised whenever this happens. The following procedures describe how to obtain a new page and release a page.

## Obtain a New Page

The procedure below may be used to get a new page.

```
pseudo_supervisor$newpag (loc, size, error, code)
```

where the arguments are:

| identifier | attribute | meaning |
|---|---|---|
| 1. loc | fixed | absolute memory location of page |
| 2. size | fixed | page size requested |

| identifier | attribute | meaning |
|---|---|---|
| 3. error | label | error return |
| 4. code | fixed | error identifying code |

This procedure first attempts to satisfy the request from the free page pool. If it fails, unused page(s) from the top of the stack will be freed until either the request is satisfied or the unused stack area is exhausted. The arg1 is set equal to the absolute memory location of the new page.

Error codes and their meanings:

1. Both free page list and unused stack not able to satisfy request
2. Page size not 1024 or 64.
3. Stack paged, but page table not found.

Release a Page

A page may be given to the free page pool by the following call:

pseudo_supervisor$relpag (loc, size, error, code)

Where the arguments are:

| identifier | attribute | meaning |
|---|---|---|
| 1. loc | fixed | absolute location of page |
| 2. size | fixed | page size |
| 3. error | label | error return |
| 4. code | fixed | code identifying error |

The page defined by arg1 and arg2 is added to the appropriate list in the free page pool.

Error codes and their meanings:

1. Free page list full for given page size.
2. Page size not 1024 or 64.