Virtual Memory
Virtual Memory

* ISA has large VA space.
  - Allows user to uniquely identify lots
* Physical Memory is smaller
  - Cost issue

* Virtual Memory Management
  - Access Control
  - Translation

* The VAX Model

Virtual Memory

```
A
B
C
K
L
M
N

D
E
F
G
K
L
M
N

H
I
J
K
L
M
N
```

Physical Memory

```
A
K
G
E
F
P
I
D
L
M
C
```

Process 1  Process 2  Process n

Balance Set
The PTE

31  V  Prot  M  PFN  0

Can I Believe The PFN

Has the Page Been Written

Do I have the Right to do this access

\{NO, R, R/W\} \{K, E, S, U\}

81 Possibilities in 4 bits?

Note:

No Ref. Bit!
Page Tables

* One for each region

* For example, the P0 Page Table

<table>
<thead>
<tr>
<th>0</th>
<th>PTE for Page 0</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>PTE for Page 1</td>
</tr>
<tr>
<td>2</td>
<td>PTE for Page 2</td>
</tr>
<tr>
<td></td>
<td>...</td>
</tr>
<tr>
<td>L-1</td>
<td></td>
</tr>
</tbody>
</table>

  P0 Base Register

  L

  P0 Length Register

* Sequentially stored in System Virtual Space

* P0LR used for ACV checks

* PTE used for ACV, TNV checks
The Abstraction

Page No.
Which Page Table
Example: 00 PØPT

1. Is Page No. < PØLR?
   NO: ACV Fault!

2. Get Correct PTE

   Index by Page No.

   PTE for Page I

   PØBR

3. V Prot PFN The PTE

   Check Protection
   NO: ACV Fault!

4. Is Page Resident (i.e. V = 1)
   NO: TNV Fault!

5. All Cool: PA
VA of x

PXBR

SBR

PTE of Page Containing X

PTE of Page of System Space

PTE of Page Containing X

PA of x

And now we can access x
One Final Example

Let's modify the VAX ISA to make it easier to see what is going on. We will retain the essential elements but we will reduce all the numbers.

For example: VA will go from 32 bits to 9 4 bits.
Page size will go from 512 B to 16 B.
PA will be 7 4 bits.
PTE will still be 4 bytes.

Page 2 shows a snapshot of virtual & physical memory. Several things are worth noting:

1. **Virtual Memory** = 512 bytes. :: 32 Pages Possible
   P# has a max of 8. In our example: 6 pages were needed.
P1 " " " " " In our example: 2 pages " "
SS " " " " " In our example: 5 pages " "

2. **P# Page Table Starts at VA = 120** (Note: 2, 0 are hex digits)
   Since there must be 6 entries, the P# Page Table consumes all of page 2, half of page 3 of SS.

3. **System Page Table Starts at PA = 50** (Note: 5 is offset, 0 is hex)
   Since there must 5 entries, system page table consumes all of frame 5 and 1/4 of frame 6.

4. Note that sys. page table indicates 3 pages resident (PAGES 032)
   Note that part of P# PTE table indicates page 5 in page 4 of P#.
Finally, a page table computation:

We wish to execute: LD R1, X
WHERE X has VA: 001011000

VA of X: 001011000

PB

Page 5

Because PTE contains 4 bytes

Page 0

To index into P0 page table

10100

100100000

100100000

VA of PTE of Page 5, P0 space

(Which is on page 3 of virtual system space)

Page 3

[4]

01100

SS

SBR

10100000

+1

To index into system page table

10111010

PA of PTE of Page 3, system space

SBR

PTE of Page 3 of System space

It occupies PFN 1

VA of PTE of Page 5 of P0 space 10011000000
is mapped to PA 00101000
So, if we look at PFN 1

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>2</td>
</tr>
</tbody>
</table>

We can read the PTE of Page 5, P0 space. We see Page 5, P0 space occupies PFN 2.

The VA of \( \text{X00'101'1000} \) therefore maps to

PA \( \text{D10'1000} \)

We look in that location and find 17, which we load into R1.
**Flat Model**

```
31  LOGICAL ADDR  0  31
```

**Real Address**

```
19  SEGMENT SELECT 0000  3 of  LOGICAL ADDRESS
```

**Segmented Model**

```
13  Segment Selector
```

---

**Linear Address Space**

```
20
```

---

**Linear Address Space**

```
32
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
32
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```

---

**Linear Address Space**

```
2
```
**Segment Registers**

- 16 bits

**Segment Descriptor**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| BASE [15:8] | LIMIT [15:8] |

**G**: Granularity 1 byte/4K bytes

**P**: Segment Present

**PL**: Privilege Level

**TYPE**: Segment type

2¹³ x 8 bytes each
Segmentation AND Paging
Task State Segment (TSS)

<table>
<thead>
<tr>
<th>I/O</th>
<th>EAX</th>
<th>DSBR</th>
<th>O</th>
<th>T</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td></td>
<td></td>
<td>0</td>
<td>LDT</td>
</tr>
<tr>
<td>0</td>
<td></td>
<td></td>
<td>GS</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EDI</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EAX</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CPLA</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ESI</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CR3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SS2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ESPL</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SS1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ESP1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SSF</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ESP</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ESP</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ESP</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ESP</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ESP</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ESP</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ESP</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ESP</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ESP</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ESP</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ESP</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

}