Cache Memory

WHY?

CPU

\[0.25\text{ nsec} - 0.5\text{ nsec}\]

Memory

100 nsec
The Abstraction

Physical Address

Tag Store

To see if it is there

(Also called Cache A)

"Yes" (A Hit!)

Data

To get It

(Also called Cache D)

Data

Note: Hit Ratio =

\[
\frac{\text{Hits}}{\text{Hits} + \text{Misses}}
\]
Write Through/Write Back

```
PROC -> CACHE -> MEM  

vs

PROC -> CACHE -> MEM
```

**Issues**

* Simplicity of Design
* Bus Traffic
* Application Environment (Stack Frame)
* Allocate on Write Miss
  - Sector Cache
Characteristics

* Set Associative (Set Size)
  - Fully Associative
  - Direct Mapped

* Write Back, Write Through

* Replacement Algorithm
  - LRU
  - FIFO
  - Random

* Instructions/Data

* Supervisor/User

* Virtual/Physical
Tag Store Entry: LRU

Tag | V | D | Replace