Presentation with Beamer

Team I

June 9, 2021

Outline

No pause here

- wow
- no test

Description

- Test

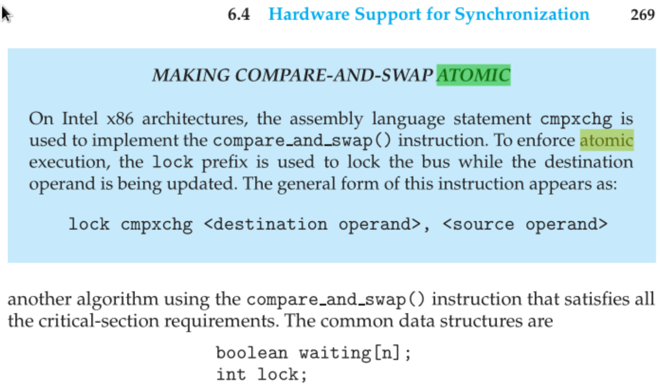
Description

- Test
- Testing

Description

- Test
- Testing
- Tester

Something



MAKING COMPARE-AND-SWAP ATOMIC

On Intel x86 architectures, the assembly language statement `cmpxchg` is used to implement the `compare_and_swap()` instruction. To enforce atomic execution, the `lock` prefix is used to lock the bus while the destination operand is being updated. The general form of this instruction appears as:

```
lock cmpxchg <destination operand>, <source operand>
```

another algorithm using the `compare_and_swap()` instruction that satisfies all the critical-section requirements. The common data structures are

```
boolean waiting[n];
int lock;
```