

Styresystemer og Multiprogrammering

G-opgave #4

Jenny-Margrethe Vej
(rwj935@alumni.ku.dk)
Klaes Bo Rasmussen
(twb822@alumni.ku.dk)

Datalogisk Institut, Københavns Universitet
Blok 3 - 2013

TLB exception handling in Buenos

(1)

Da opgaveteksten siger, load og store håndteres på samme måde, har vi valgt at lave en hjælpefunktion, der håndterer begge tilfælde. Funktionen hedder `tlb_exception_handler` og starter med at sammenligne den virtuelle adresse som fejlen er meldt på, med adresserne i pagetables i den angivne tråd. Hvis vi finder adressen, sættes den ind i TLB, ellers behandler vi den som en access violation, henholdsvis i bruger-tråds tilstand eller kerne tilstand. Da vi behandler både load og store i handleren, findes der fejlbeskeder for begge her.

`tlb_load_exception` og `tlb_store_exception` kalder vores handler, og bruger derfor den samme kode. `tlb_modified_exception` håndteres på næsten samme måde, som i vores handler, hvor vi også sørger for at håndtere access violation i bruger-tråds tilstande samt kernel panic i kerne tilstande. Derfor er det samme kode, der blot er kopieret op, dog uden genbrug af de if-sætninger, der håndterede de 2 forskellige fejlbeskeder for henholdsvis store og load.

(2)

Vi har ændret 2 filer, for at opfylde opgavens krav: `kernel/exception.c` og `proc/exception.c`. Det eneste, vi har ændret, er at tilføje `tlb_load_exception`, `tlb_store_exception` og `tlb_modified_exception` til de switch-cases, der var.

(3)

Vi har via terminalkommandoen `grep -r tlb_fill .` fundet ud af, at `tlb_fill` eksisterer i følgende filer, hvor vi efterfølgende har fjernet dem. `tlb.c`, `tlb.h`, `interrupt.c` samt `process.c`.

Dynamic allocation for user processes

(1)

Ikke implementeret grundet sygdom og travlhed

(2)

Ikke implementeret grundet sygdom og travlhed

Extended tests for TLB exceptions(..)

Ikke testet ordentligt grundet sygdom og travlhed