Bài thực hành số 11

Lớp: 139365 – Học phần: Thực Hành Kiến Trúc Máy Tính

Họ và Tên: Nguyễn Anh Thứ MSSV: 20215144

Bài 1:

A screenshot of a computer program

Description automatically generated with medium confidence

Giải thích:

* Đầu tiên, định nghĩa các địa chỉ của bàn phím và các biến cho các địa chỉ này:
* IN\_ADRESS\_HEXA\_KEYBOARD: Địa chỉ đầu vào để ghi giá trị hàng (row) của bàn phím số.
* OUT\_ADRESS\_HEXA\_KEYBOARD: Địa chỉ đầu ra để đọc giá trị hàng (row) và cột (column) của phím được nhấn.
* Tiếp theo, chương trình chuyển đến phần .text để bắt đầu thực thi mã chương trình chính.
* Tại nhãn main:, chương trình khởi tạo các thanh ghi $t1, $t2, và $t4 để lưu trữ các địa chỉ của bàn phím và giá trị 0x10.
* Nhãn set: được sử dụng để khởi tạo thanh ghi $t3 với giá trị 0x01 để quét hàng (row) thứ nhất trên bàn phím số hệ thập lục phân.
* Tiếp theo, chương trình bắt đầu quét bàn phím và kiểm tra xem có phím nào được nhấn không. Nhãn polling: được sử dụng để bắt đầu vòng lặp quét.
* Chương trình kiểm tra xem giá trị của $t3 (giá trị hàng hiện tại) có bằng giá trị $t4 (giá trị tối đa của hàng) hay không. Nếu có, nó chuyển đến nhãn print để in kết quả. Nếu không, nó ghi giá trị hàng hiện tại vào địa chỉ đầu vào của bàn phím và đọc giá trị quét của phím từ địa chỉ đầu ra của bàn phím.
* Nếu giá trị quét không bằng 0, nghĩa là có phím được nhấn, chương trình sẽ chuyển đến nhãn print để in giá trị phím được nhấn.
* Nếu giá trị quét bằng 0, chương trình dịch trái (sll) giá trị của $t3 lên một bit và sau đó quay lại nhãn polling để tiếp tục quét hàng tiếp theo trên bàn phím.
* Nhãn print: được sử dụng để in giá trị của phím được nhấn. Chương trình sử dụng hệ thống syscall để in giá trị hexa của phím.
* Sau khi in giá trị phím, chương trình sử dụng syscall để tạm dừng chương trình trong 100ms.
* Chương trình sau đó quay lại nhãn set để bắt đầu quét bàn phím từ đầu.
* Nhãn back\_to\_polling: được sử dụng để quay lại nhãn polling để tiếp tục quét hàng trên bàn phím.

-Kết quả chạy

Khi chưa bấm nút:

A screenshot of a computer

Description automatically generated

Bấm nút cột 1 hàng 2:

A screenshot of a computer

Description automatically generated

Bấm nút cột 2 hàng 3:

A screenshot of a computer

Description automatically generated

Bấm nsut cột 3 hàng 4:

A screenshot of a computer

Description automatically generated

Bài 2:

A screenshot of a computer program

Description automatically generated with medium confidence

Kết quả:

A screenshot of a computer

Description automatically generated

Giải thích:

* Đầu tiên, đoạn mã .eqv IN\_ADRESS\_HEXA\_KEYBOARD 0xFFFF0012 định nghĩa địa chỉ đầu vào của bàn phím số hệ thập lục phân.
* Sau đó, chương trình khai báo một chuỗi ký tự "Oh my god. Someone's pressed a button.\n" và lưu trữ trong vùng nhớ dữ liệu (.data) với nhãn Message.
* Chương trình chuyển sang phần mã chương trình (.text) và bắt đầu từ nhãn main:.
* Trong phần này, chương trình kích hoạt ngắt (interrupt) từ bàn phím bằng cách gán giá trị 0x80 cho bit 7 của địa chỉ đầu vào của bàn phím (IN\_ADRESS\_HEXA\_KEYBOARD). Điều này cho phép chương trình chờ và phản ứng khi có phím được nhấn.
* Tiếp theo, chương trình chạy vòng lặp vô hạn (Loop) để chờ ngắt từ bàn phím. Với các lệnh nop, chương trình chỉ đang chờ và không thực hiện bất kỳ công việc nào.
* Sau khi có ngắt từ bàn phím, chương trình chuyển đến nhãn IntSR (Interrupt Service Routine), nơi xử lý ngắt từ bàn phím.
* Trong IntSR, chương trình sử dụng syscall để hiển thị thông điệp "Oh my god. Someone's pressed a button.\n" lên màn hình.
* Tiếp theo, chương trình lấy địa chỉ trả về của quy trình chính (main) và tăng giá trị của địa chỉ trả về lên 4 để nhảy tới lệnh tiếp theo sau lệnh ngắt.
* Cuối cùng, chương trình sử dụng lệnh eret để trở về từ ngoại lệ và tiếp tục thực hiện lệnh tiếp theo trong chương trình chính.

Bài 3:

A screenshot of a computer

Description automatically generated

A screenshot of a computer program

Description automatically generated with low confidence

A screenshot of a computer program

Description automatically generated with medium confidence

Kết quả:

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

* Đầu tiên, đoạn mã .eqv IN\_ADRESS\_HEXA\_KEYBOARD 0xFFFF0012 và .eqv OUT\_ADRESS\_HEXA\_KEYBOARD 0xFFFF0014 định nghĩa các địa chỉ đầu vào và đầu ra của bàn phím số hệ thập lục phân.
* Tiếp theo, chương trình khai báo một chuỗi ký tự "Key scan code " và lưu trữ trong vùng nhớ dữ liệu (.data) với nhãn Message.
* Chương trình chuyển sang phần mã chương trình (.text) và bắt đầu từ nhãn main:.
* Trong phần này, chương trình kích hoạt ngắt (interrupt) từ bàn phím bằng cách gán giá trị 0x80 cho bit 7 của địa chỉ đầu vào của bàn phím (IN\_ADRESS\_HEXA\_KEYBOARD). Điều này cho phép chương trình chờ và phản ứng khi có phím được nhấn.
* Tiếp theo, chương trình sử dụng thanh ghi $s0 để lưu trữ giá trị đếm (count) và khởi tạo nó bằng 0.
* Chương trình chạy trong một vòng lặp (Loop) để tăng giá trị đếm lên mỗi lần lặp và sau đó in giá trị đếm lên màn hình sử dụng syscall.
* Sau khi in giá trị, chương trình sử dụng syscall để in ký tự xuống dòng ('\n').
* Sau đó, chương trình sử dụng syscall để tạm dừng chương trình trong 300ms.
* Chương trình quay lại nhãn Loop để tiếp tục vòng lặp.
* Sau khi có ngắt từ bàn phím, chương trình chuyển đến nhãn IntSR (Interrupt Service Routine), nơi xử lý ngắt từ bàn phím.
* Trong IntSR, chương trình sử dụng syscall để hiển thị thông điệp "Key scan code " lên màn hình.
* Tiếp theo, chương trình lấy giá trị hàng (row) hiện tại và gán lại vào địa chỉ đầu vào của bàn phím để quét hàng tiếp theo.
* Chương trình đọc giá trị quét (scan code) của phím từ địa chỉ đầu ra của bàn phím và in giá trị quét lên màn hình sử dụng syscall.
* Sau khi in giá trị quét, chương trình sử dụng syscall để in ký tự xuống dòng ('\n').
* Tiếp theo, chương trình đánh giá địa chỉ trả về của chương trình chính (main) và tăng giá trị địa chỉ trả về lên 4 để nhảy tới lệnh tiếp theo sau lệnh ngắt.
* Sau đó, chương trình khôi phục các thanh ghi từ ngăn xếp (stack) và trả về từ ngoại lệ (eret) để tiếp tục thực hiện lệnh tiếp theo trong chương trình chính

Bài 4:   
A screenshot of a computer program

Description automatically generated with medium confidence

A screenshot of a computer program

Description automatically generated with medium confidence

A screenshot of a computer program

Description automatically generated with low confidence

Không nhấn phím:

A screenshot of a computer

Description automatically generated

Nhấn phím:

A screenshot of a computer

Description automatically generated

* Đầu tiên, các địa chỉ đầu vào và đầu ra của bàn phím và bộ đếm thời gian được định nghĩa bằng các hằng số .eqv IN\_ADRESS\_HEXA\_KEYBOARD 0xFFFF0012 và .eqv COUNTER 0xFFFF0013.
* Chương trình khai báo các thông điệp dùng để in ra màn hình khi có ngắt từ bàn phím hoặc bộ đếm thời gian, bao gồm msg\_keypress và msg\_counter.
* Trong chương trình chính (main), ngắt từ bàn phím và bộ đếm thời gian được kích hoạt bằng cách gán giá trị 0x80 cho bit 7 của địa chỉ đầu vào của bàn phím và gán giá trị COUNTER vào địa chỉ của bộ đếm thời gian.
* Tiếp theo, chương trình chạy trong một vòng lặp vô hạn (Loop) và sử dụng syscall để tạm dừng chương trình trong 200ms (sleep). Điều này cho phép bộ đếm thời gian tăng và có thể tạo ngắt sau mỗi khoảng thời gian.
* Sau đó, chương trình chuyển đến ngắt chung (IntSR - Interrupt Service Routine) để xử lý các ngắt từ bàn phím và bộ đếm thời gian.
* Trong quá trình xử lý ngắt (IntSR), chương trình kiểm tra giá trị của thanh ghi nguyên nhân (Coproc0.cause) để xác định ngắt từ bàn phím hay bộ đếm thời gian.
* Nếu ngắt từ bàn phím được xác nhận, chương trình sử dụng syscall để in thông điệp "Someone has pressed a key!" lên màn hình.
* Nếu ngắt từ bộ đếm thời gian được xác nhận, chương trình sử dụng syscall để in thông điệp "Time interval!" lên màn hình.
* Sau khi xử lý ngắt, chương trình xóa thanh ghi nguyên nhân (Coproc0.cause) bằng cách gán giá trị 0 cho nó (mtc0 $zero, $13).
* Tiếp theo, chương trình kích hoạt lại ngắt bằng cách gán giá trị COUNTER vào địa chỉ của bộ đếm thời gian.
* Cuối cùng, chương trình đánh giá địa chỉ trả về của chương trình chính (main) và tăng giá trị địa chỉ trả về lên 4 để nhảy tới lệnh tiếp theo sau lệnh ngắt. Sau đó, chương trình trả về từ ngoại lệ (eret) để tiếp tục thực hiện lệnh tiếp theo trong chương trình chính.

Bài 5:

A screenshot of a computer program

Description automatically generated with medium confidence

A picture containing text, font, screenshot

Description automatically generated

Kết quả:

A screenshot of a computer

Description automatically generated with medium confidence

* Các địa chỉ liên quan đến bàn phím và hiển thị được định nghĩa bằng các hằng số .eqv KEY\_CODE, .eqv KEY\_READY, .eqv DISPLAY\_CODE, và .eqv DISPLAY\_READY.
* Trong chương trình chính (loop), ta sử dụng lệnh lw để đọc giá trị từ địa chỉ KEY\_READY. Nếu giá trị bằng 0, chương trình sẽ tiếp tục lặp và chờ đến khi có ký tự từ bàn phím (WaitForKey).
* Sau khi có ký tự từ bàn phím, chương trình kiểm tra giá trị KEY\_READY. Nếu bằng 1, chương trình sẽ gửi một ngắt (MakeIntR) bằng cách sử dụng lệnh teqi và nhảy lại vòng lặp chính (loop).
* Trong phần ngắt (get\_caus), chương trình kiểm tra thanh ghi nguyên nhân (Coproc0.cause) để xác định ngắt từ bàn phím.
* Nếu ngắt từ bàn phím được xác nhận, chương trình sử dụng lệnh lw để đọc giá trị từ địa chỉ KEY\_CODE và lưu vào thanh ghi $t0.
* Tiếp theo, chương trình kiểm tra giá trị DISPLAY\_READY. Nếu bằng 0, chương trình sẽ tiếp tục lặp và chờ đến khi màn hình sẵn sàng để hiển thị ký tự (WaitForDis).
* Sau khi màn hình sẵn sàng, chương trình tăng giá trị ký tự trong thanh ghi $t0 lên 1 (Encrypt) và sử dụng lệnh sw để lưu giá trị ký tự mới vào địa chỉ DISPLAY\_CODE để hiển thị lên màn hình (ShowKey).
* Cuối cùng, chương trình tiếp tục thực hiện lệnh tiếp theo sau lệnh ngắt (next\_pc) và trả về từ ngoại lệ (eret) để tiếp tục thực hiện chương trình chính.