**Bài thực hành số 11**

**Lớp: 139365 – Học phần: Thực Hành Kiến Trúc Máy Tính   
Đào Minh Nhật – 20215107**

**Assignment 1:**

**Code:**

**Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, số

Mô tả được tạo tự động**

**Giải thích:**

Biến IN\_ADRESS\_HEXA\_KEYBOARD lưu địa chỉ 0xFFFF0012 dùng để lưu địa chỉ của hàng cần duyệt.

Biến OUT\_ADRESS\_HEXA\_KEYBOARD lưu địa chỉ 0xFFFF0014 để phát hiện nút nào đã được bấm.

Cách hoạt động:

Thanh $t3 lưu địa chỉ của hàng cần duyệt

Duyệt lần lượt từng hàng 0x01, 0x02, 0x04, 0x08 bằng cách dùng lệnh dịch bit sll. Nếu $t3 = 0x10 thì sẽ reset lại $t3 = 0x01 để duyệt lại từ hàng đầu.

Nếu phát hiện ra nút bấm thì dùng lệnh lb từ địa chỉ 0xFFFF0014 để lấy ra nút đã được bấm và in ra.

**Kết quả:**

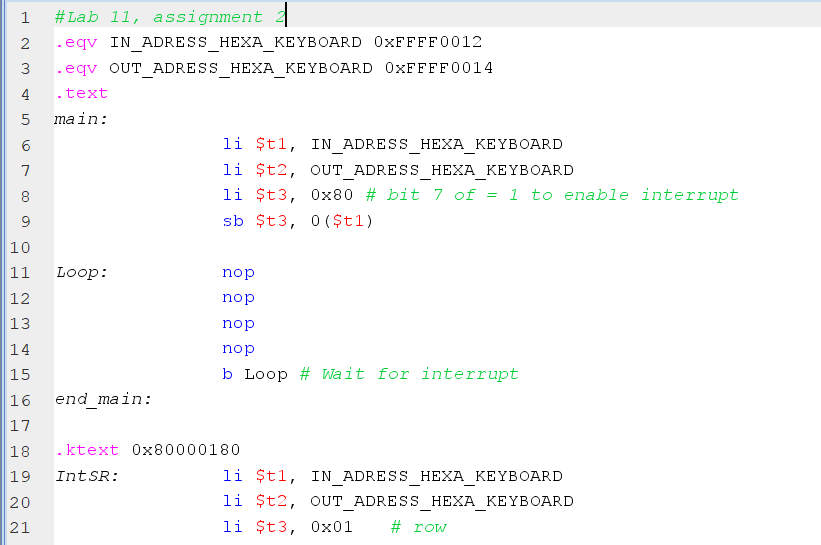
**Ảnh có chứa văn bản, ảnh chụp màn hình, màn hình, phần mềm

Mô tả được tạo tự độngẢnh có chứa văn bản, ảnh chụp màn hình, màn hình, phần mềm

Mô tả được tạo tự động**

**Assignment 2:**

**Code:**

****

Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, số

Mô tả được tạo tự động

**Giải thích:**

Gán thanh ghi $t3 = 0x80 đặt bit thứ 7 là 1 để kích hoạt ngắt trên bàn phím (interrupt).

Vòng lặp loop để chờ ngắt.

Sử dụng chỉ thị .ktext để viết code ở địa chỉ 0x80000180 để viết chương trình con phục vụ ngắt.

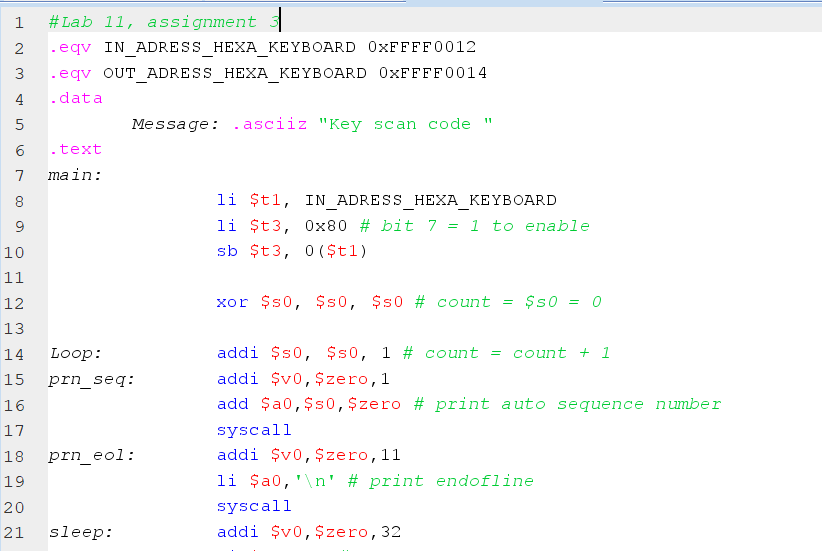
Hàm IntSR thực hiện in ra nút đã bấm.

Sau khi in được nút đã bấm thì gán lại thanh ghi $t3 = 0x80 vào địa chỉ IN\_ADRESS\_HEXA\_KEYBOARD để chuẩn bị cho lệnh ngắt tiếp theo.

Hàm next\_pc : Lấy giá trị của thanh ghi $at (Coproc0.$14) và cộng thêm 4 để chuyển đến lệnh kế tiếp.

**Assignment 3:**

**Code:**



Ảnh có chứa văn bản, ảnh chụp màn hình, số, phần mềm

Mô tả được tạo tự động

Ảnh có chứa văn bản, ảnh chụp màn hình, số, phần mềm

Mô tả được tạo tự động

Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, số

Mô tả được tạo tự động

**Giải thích:**

Chương trình thực hiện đếm số lần lượt từ 0.

Nếu suất hiện ngắt bàn phím thì sẽ in ra nút được bấm.

Cách hoạt động:

Khởi tạo $t3 = 0x80 ( khởi tạo bit 7 = 1) để kích hoạt ngắt từ bàn phím.

$s0 là đếm số.

Vòng lặp loop thực hiện đếm số chờ lệnh ngắt từ bàn phím.

Hàm prn\_seq: in ra chuỗi số đếm.

Hàm prn\_eol: in ra xuống dòng.

Hàm IntSR: thực hiện in ra số được bấm.

Lệnh addi và sw dùng để lưu trữ các giá trị $ra, $at, $v0, $a0, $t1, $t3 vào stack.

Hàm get\_cod: thực hiện lấy giá trị của nút được bấm gán vào thanh ghi $a0. Bằng cách duyệt lần lượt từng hàng.

Hàm prn\_cod: in ra nút bấm vừa tìm được và xuống dòng.

Hàm next\_pc: Cộng thêm 4 vào thanh $at để giá trị trả về là địa chỉ của lệnh tiếp theo.

Hàm restore: lấy lại các giá trị đã được lưu trong stack ban đầu.

Hàm return: trở lại đếm số và chờ lệnh ngắt tiếp theo

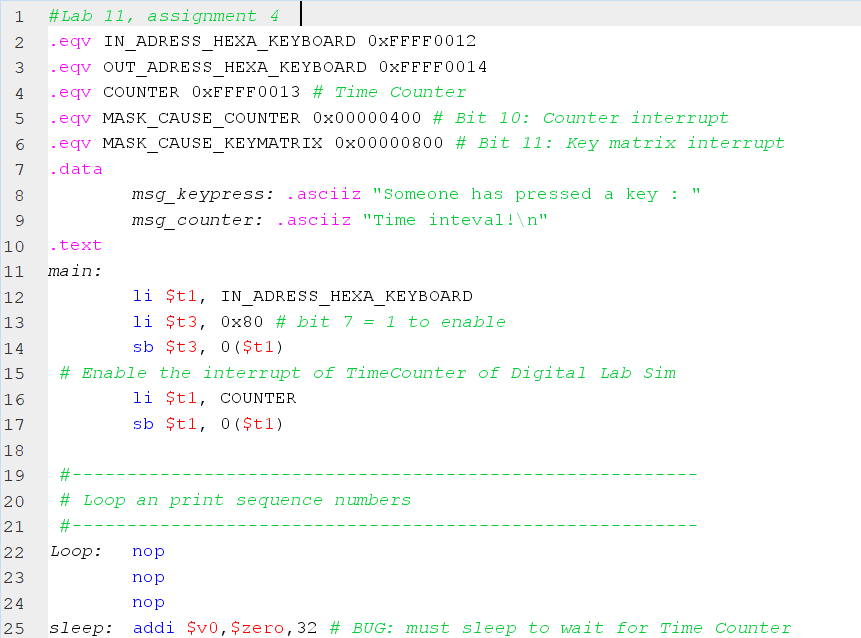
**Kết quả:**

Ảnh có chứa văn bản, phần mềm, biểu đồ, Sơ đồ

Mô tả được tạo tự động

**Assignment 4:**

**Code:**

****

**Ảnh có chứa văn bản, ảnh chụp màn hình, số, phần mềm

Mô tả được tạo tự động**

**Ảnh có chứa văn bản, ảnh chụp màn hình, số, phần mềm

Mô tả được tạo tự động**

**Ảnh có chứa văn bản, ảnh chụp màn hình, số, Phông chữ

Mô tả được tạo tự động**

**Giải thích:**

Hàm ‘main’:

* Khởi tạo bit 7 = 1 và gán vào địa chỉ đầu vào của bàn phím để kích hoạt ngắt.
* Ghi giá trị trong thanh ghi $t1 (COUNTER - địa chỉ thanh ghi đếm thời gian) vào địa chỉ thanh ghi đếm thời gian để kích hoạt ngắt đếm thời gian.
* Vòng lặp loop và sleep để chờ ngắt.

Hàm ‘dis\_int’: vô hiệu hóa ngắt bằng cách gán giá trị 0 vào địa chỉ thanh ghi đếm thời gian.

Hàm ‘get\_caus’: sao chép giá trị của thanh ghi Cause vào thanh ghi $t1 để lấy nguyên nhân ngắt.

Hàm ‘IsCount’: kiểm tra ngắt đếm (Counter interrupt).

* $t2 lưu giá trị ngắt đếm MASK\_CAUSE\_COUNTER
* So sánh $t1(nguyên nhân) với $t2. Nếu bằng nhau tức là ngắt đếm thì nhảy tới hàm ‘Counter\_Intr’ để thực hiện ngắt đếm.
* Ngược lại thì thực hiện hàm ‘IsKeyMa’ để kiểm tra ngắt bàn phím.

Hàm ‘IsKeyMa’ : kiểm tra ngắt bàn phím (Key matrix interrupt)

* $t2 lưu giá trị ngắt đếm MASK\_CAUSE\_KEYMATRIX
* So sánh $t1(nguyên nhân) với $t2. Nếu bằng nhau tức là ngắt bàn phím thì nhảy tới hàm ‘Counter\_Intr’ để thực hiện ngắt đếm.
* Ngược lại thì tới hàm ‘end\_process’ để kết thúc với nguyên nhân ngắt khác với 2 nguyên nhân trên.

Hàm ‘Counter\_Intr’ : với nguyên nhân ngắt là ngắt đếm thì in ra ‘Time inteval!’.

Hàm ‘Keymatrix\_Intr’: in ra nút đã được bấm.

Hàm ‘end\_process’: kết thúc kiểm tra ngắt và gán giá trị 0 vào thanh ghi $13 (Cause) để xóa nguyên nhân ngắt.

Hàm ‘en\_int’: kích hoạt lại ngắt bằng cách gán giá trị thanh ghi đếm thời gian vào địa chỉ của nó.

Hàm next\_pc: Cộng thêm 4 vào thanh $at để giá trị trả về là địa chỉ của lệnh tiếp theo.

**Assignment 5:**

**Code:**

**Ảnh có chứa văn bản, ảnh chụp màn hình, màn hình, phần mềm

Mô tả được tạo tự động**

**Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, số

Mô tả được tạo tự động**

**Giải thích:**

KEY\_CODE 0xFFFF0004: Địa chỉ thanh ghi lưu trữ mã ASCII của phím từ bàn phím, kích thước 1 byte. ($k0)

KEY\_READY 0xFFFF0000: Địa chỉ thanh ghi cho biết liệu có mã ASCII mới từ bàn phím hay không, có giá trị bằng 1 nếu có, tự động xóa sau lệnh lw, kích thước 1 byte. ($k1)

DISPLAY\_CODE 0xFFFF000C: Địa chỉ thanh ghi lưu trữ mã ASCII để hiển thị trên màn hình, kích thước 1 byte. ($s0)

DISPLAY\_READY 0xFFFF0008: Địa chỉ thanh ghi cho biết liệu màn hình đã sẵn sàng để hiển thị hay chưa, có giá trị bằng 1 nếu đã sẵn sàng, tự động xóa sau lệnh sw, kích thước 1 byte. ($s1)

MASK\_CAUSE\_KEYBOARD: Mặt nạ (mask) để xác định nguyên nhân ngắt từ bàn phím.

Hàm ‘WaitForKey’: nếu giá trị trong thanh ghi $t1 bằng 0, tức là không có mã ASCII mới, thì quay lại nhãn WaitForKey để tiếp tục kiểm tra

Hàm ‘MakeIntR’: so sánh giá trị trong thanh ghi $t1 với 1. Nếu bằng nhau, tức là có mã ASCII mới từ bàn phím, thì gây ra một ngắt.

Hàm ‘get\_caus’: đọc giá trị từ thanh ghi Cause vào thanh ghi $t1 để xác định nguyên nhân của ngắt.

Hàm ‘IsCount’: kiểm tra xem nguyên nhân của ngắt có phải là từ bàn phím hay không bằng cách so sánh giá trị trong thanh ghi $t1 với MASK\_CAUSE\_KEYBOARD.

Hàm ‘ReadKey’: đọc giá trị từ địa chỉ thanh ghi $k0 vào thanh ghi $t0 để lưu trữ mã ASCII của phím từ bàn phím .

Hàm ‘WaitForDis’:

* Đọc giá trị từ địa chỉ thanh ghi $s1 vào thanh ghi $t2 để kiểm tra xem màn hình đã sẵn sàng để hiển thị hay chưa.
* Nếu giá trị trong thanh ghi $t2 bằng 0, tức là màn hình chưa sẵn sàng, thì quay lại nhãn ‘WaitForDis’ để tiếp tục kiểm tra.

Hàm ‘Encrypt’: Tăng input key thêm 1 đơn vị để mã hóa trước khi in ra màn hình.

Hàm ‘ShowKey’: Ghi giá trị trong thanh ghi $t0 vào địa chỉ thanh ghi $s0 để hiển thị mã ASCII của phím trên màn hình.

Hàm ‘next\_pc’ và ‘return’ để quay lại lệnh tiếp theo của chương trình chính.

Kết quả:

Ảnh có chứa văn bản, ảnh chụp màn hình, màn hình, phần mềm

Mô tả được tạo tự động