

**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**  
**Khoa Công Nghệ Thông Tin 1**

---

**BÁO CÁO XÂY DỰNG ỨNG DỤNG**

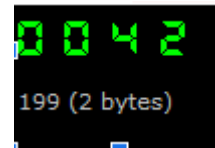
**Bộ Môn: Kiến Trúc Máy Tính**  
**Nhóm Môn Học: Nhóm 3**  
**Đề Tài: KIỂM SOÁT GIAO THÔNG**  
**Giảng viên: VŨ HOÀI NAM**

**I. Thành viên và Công việc.**

Thành viên	MSV	Công việc			
Vương Huy Long	B19DCCN400	Lên ý tưởng cho hàm tổng và hàm con,viết báo cáo, viết hàm WAIT. Tổng hợp code.			
Lê Trường Giang	B19DCCN205	Lên ý tưởng và viết hàm SINH SUÇO . Giải thích code cho dương.			
Vũ Minh Trí	B19DCCN689	Lên ý tưởng và viết hàm PORT4, hàm THÔNGBAO. Giải thích code cho dương.			
Nguyễn Hoàng Dương	B19DCCN153	Làm slide giải thích ứng dụng , thuyết trình giải thích ứng dụng.			

## II. Ý TƯỞNG VÀ KHÁI NIỆM.

1. **Ý tưởng:** Trong quá trình giao thông có thể sẽ *phát sinh sự cố ngoài ý muốn* khiến giao thông bị cản trở. Ta cần xử lý các sự cố sao cho không xảy ra mâu thuẫn giữa các tuyến đường.
2. **Các thiết bị I/O sử dụng:**
  - **Traffic light** : Hiển thị quá trình lưu thông.
  - **Led\_Display**: Hiển thị thời gian của đèn giao thông.



3. **Các khái niệm và cách hoạt động.**

Giao thông được điều khiển bằng chuỗi 16 bit **FEDC\_BA98\_7654\_3210**.

*Trong đó : 4 bit FEDC ko được sử dụng.*

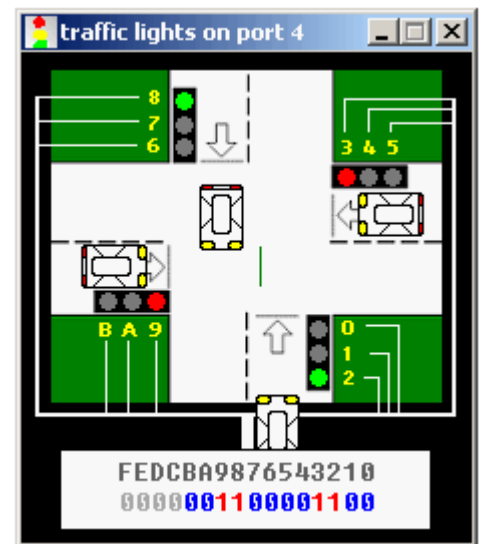
*4 bit B,8,5,2 tương ứng đèn xanh*

*4 bit 9,6,3,0 tương ứng đèn đỏ*

*4 bit A,7,4,1 tương ứng đèn vàng*

Nếu bit tương ứng có giá trị 1 thì đèn tương ứng sẽ bật. Ngược lại sẽ tắt.

Chuỗi bit sẽ chuyển vào cổng 4 của ứng dụng để hoạt động.



4. **Tình huống giả tưởng và cách xử lý.**

- Trong quá trình lưu thông, nếu giao thông gặp sự cố ở tuyến đường có đèn 5.
- Ta sẽ bật đèn 5 thành đèn đỏ để tuyến đường đó ngừng hoạt động.
- Các tuyến đường còn lại tiếp tục hoạt động như sau:
- Đèn đỏ → Đèn xanh. Đèn vàng → Đèn đỏ. Đèn xanh → Đèn vàng.
- Đồng thời đèn led sẽ đếm ngược thời gian và hiển thị lên màn hình.

## III. CẤU TRÚC ỨNG DỤNG.

1. **Hàm tổng.**

**Onđinh:**

```
Call SINH_SUCO      ; tao suco.
;Check if exist SUCO
  cmp SUCO,0
  je LED:
```

```
;Thong bao len CONSOLE neu co
Call THONGBAO
```

**LED:**

```
CALL _PORT4
CALL WAITT
```

```
CALL SUCO_KETTHUC
```

```
CALL WAITT
```

**Uonglap:**

- Chương trình là một vòng lặp vô hạn.
- Công việc 1: Sinh sự cố  $\rightarrow$  Thúc tạo giá trị cho biến sự cố.
- Công việc 2: Thông báo nếu có sự cố.
- Công việc 3: Đưa trạng thái hiện tại sang trạng thái mới.
- Công việc 4: Hiển thị đèn đếm ngược.
- Công việc 5: Kiểm tra sự cố đã kết thúc chưa?

2. **Các biến toàn cục:** Đây là các biến quan trọng ảnh hưởng trực tiếp đến giao thông. Ngoài ra, có thể cần một vài biến khác để xử lý dễ dàng hơn.

	SUCO	DW	?
	TT	DW	?

### 3. Các hàm con:

**Hàm SINH\_SUCO: Tạo một biến SUCO.**

Ta sẽ tạo sự cố bằng hàm SINH\_SUCO.

Gồm 3 bước thực hiện:

**Bước 1:** Sinh giá trị ngẫu nhiên  $x$  ( $x$  thuộc tập  $D(0,1)$  )

**Bước 2:** Nếu  $x == 1$ . Sinh giá trị cho SUCO bằng cách sinh giá trị  $y$  cho địa chỉ SI ( khởi tạo **SI = offset SUCO0** ) ( $y$  thuộc tập  $T(0 \rightarrow 9)$  ).

$$SI = SI + y * 2$$

*Bởi vì SUCO là kiểu dữ liệu DW do vậy khoảng cách giữa 2 ô nhớ là 2*

**Bước 3:** Chuyển giá trị thuộc ô nhớ SI vào biến SUCO.

*( quá trình chọn ngẫu nhiên sự cố ngẫu nhiên mang ý nghĩa cho việc phát sinh sự cố trong GT)*

SUCO0

...

SUCO9

**Hàm \_PORT4:** Chuyển trạng thái hiện tại sang trạng thái mới bằng hàm BATDEN.

**BATDEN ENDP**

**\_PORT4 proc  
begin:**

**CALL BATDEN**

**mov ax,TT  
out 4,ax**

**ret  
\_PORT4 ENDP**

### Hàm BATDEN:

Kiểm tra xem Trạng thái hiện tại có bị xung đột không. Ví dụ nếu 2 tuyến song song đều có đèn đỏ, vậy ta phải chuyển 1 tuyến thành xanh.

Kiểm tra từng tuyến một. Khi kiểm tra,

đèn vàng → đèn đỏ, đèn xanh → đèn vàng, đèn đỏ → đèn xanh. Nếu có sự cố, ưu tiên bật thành đỏ. Ví dụ cho tuyến 1:

### Hàm THONGBAO:

in Thông báo sự cố nếu có và in Thông báo sự cố đã kết thúc nếu sự cố kết thúc.

### Hàm WAITT:

Khiến các đèn xanh và đỏ dừng 30 giây, nếu có đèn vàng thì chỉ dừng 3 giây.

Hiển thị thời gian lên đèn LED. Ưu tiên hiển thị đèn vàng nếu có.

Ta sử dụng một chuỗi CHECK để kiểm tra trạng thái đèn.

**CHECK DW 0000 0100 1001 0001**

Tức full đèn vàng.

**CHECK AND TT = 0000 0i00 i00i 000i**

Nếu Kết quả != 0 thì LED đếm ngược 3 giây.

Với Kết quả == 0 thì LED đếm ngược 30 giây.

Ví Dụ 0010 AND 0100 = 0. Bởi vì phép toán thực hiện AND từng bit.

### Hàm SUCO\_KETTHUC:

Biến ENSUCO nhận giá trị 0 (hết SUCO) hoặc 1 (còn ) ngẫu nhiên.

Nếu hết SỨC thì ENSUCO = 0, nếu ko giữ yên sự cố.

Nếu sự cố kết thúc thông báo lên màn hình.

Sau đó sử dụng hàm \_POST 4 để chuyển trạng thái.

```
SUCO_KETTHUC PROC
    mov ah,0 ;lay thoi thoi gian chay cua he thong luu vao dx
    int 1ah ;lay thoi thoi gian chay cua he thong luu vao dx

    mov al,dl ;luu gia tri dl vao al
    mov dl,0 ;clear gia tri dl
    mov bl,2 ;mod cho 2 de lay 0 hoac 1
    div bl
    cmp ah,1 ; neu phan du la 1 vay la su co ket thuc
    je tieptuc
    mov SUCO,0000_0000_0000b
    CALL THONGBAO
    tiectuc:

    CALL BATDEN

Ketthuc_SUCO:
```

**END.**

---



