

# Bài 6: goto - setjmp.h

Nguyễn Hoàng Anh

# Goto

goto là một từ khóa trong ngôn ngữ lập trình C/C++, cho phép chương trình nhảy đến một nhãn (label) đã được đặt trước đó trong cùng một hàm. Mặc dù nó cung cấp khả năng kiểm soát flow của chương trình, nhưng việc sử dụng goto thường được xem là không tốt vì nó có thể làm cho mã nguồn trở nên khó đọc và khó bảo trì.

# Goto

```
#include <stdio.h>
```

```
int main() {
```

```
    int i = 0; //0x01: 0
```

```
    // Đặt nhãn
```

```
start:
```

```
    if (i >= 5) {
```

```
        goto end; // Chuyển control đến nhãn "end"
```

```
    }
```

```
    printf("%d ", i);
```

```
    i++;
```

```
    goto start; // Chuyển control đến nhãn "start"
```

```
    // Nhãn "end"
```

```
end:
```

```
    printf("\n");
```

```
    return 0;
```

```
}
```

```
// 0
```

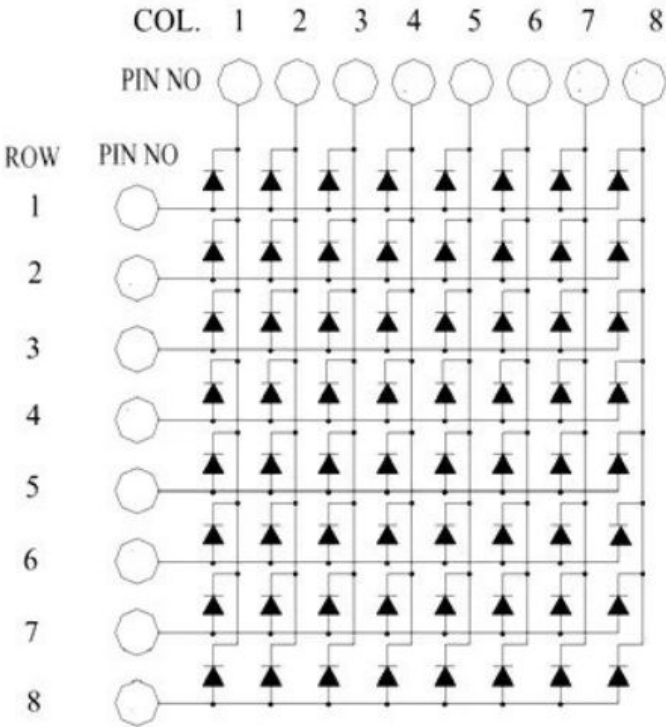
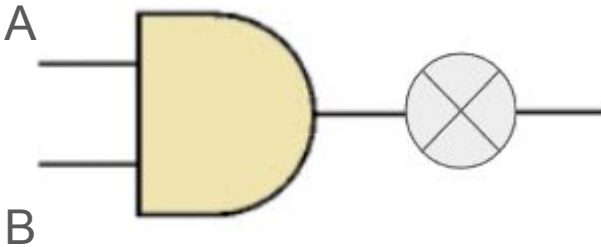
```
// 1
```

# Goto

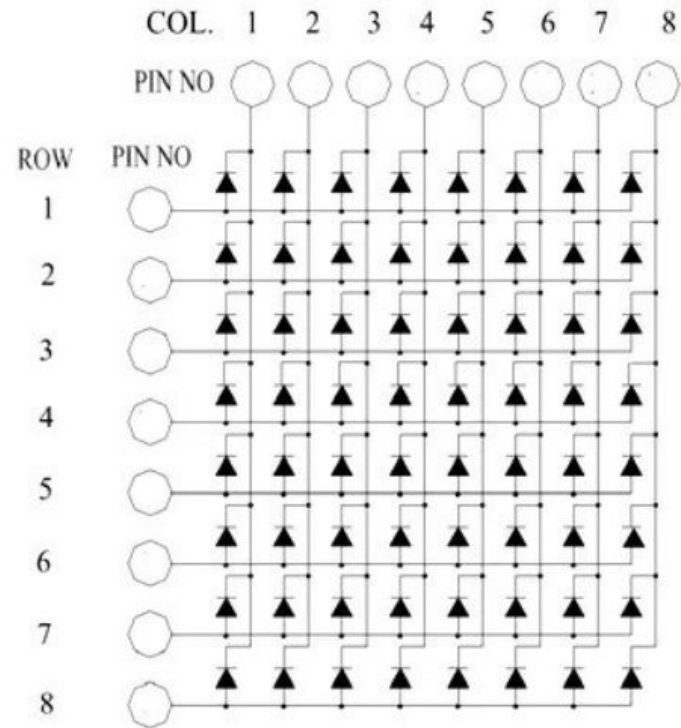
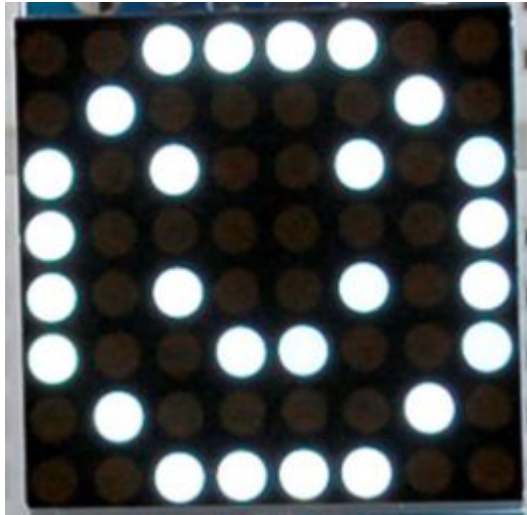
```
#include <stdio.h>
void delay(double second)
{
    double start = 0;
    while (start < second * 6000000)
    {
        start++;
    }
}

// Khai báo các trạng thái đèn giao thông
typedef enum
{
    RED,
    YELLOW,
    GREEN
} TrafficLightState;
```

Goto



# Goto



Goto



	0	1	2	3	4	5	6	7
0								
1								
2								
3								
4								
5								
6								
7								

# Goto



	0	1	2	3	4	5	6	7
0								
1								
2								
3								
4								
5								
6								
7								



# Goto



	0	1	2	3	4	5	6	7
0								
1								
2								
3								
4								
5								
6								
7								

# Goto



	0	1	2	3	4	5	6	7
0								
1								
2								
3								
4								
5								
6								
7								

# Goto



	0	1	2	3	4	5	6	7
0								
1								
2								
3								
4								
5								
6								
7								

# Goto



	0	1	2	3	4	5	6	7
0								
1								
2								
3								
4								
5								
6								
7								

# Goto



	0	1	2	3	4	5	6	7
0								
1								
2								
3								
4								
5								
6								
7								

# Goto



	0	1	2	3	4	5	6	7
0								
1								
2								
3								
4								
5								
6								
7								

# Goto



	0	1	2	3	4	5	6	7
0								
1								
2								
3								
4								
5								
6								
7								

# Goto

	0	1	2	3	4	5	6	7
0								
1								
2								
3								
4								
5								
6								
7								



# Goto

	0	1	2	3	4	5	6	7
0								
1								
2								
3								
4								
5								
6								
7								

# Goto

	0	1	2	3	4	5	6	7
0								
1								
2								
3								
4								
5								
6								
7								

# Goto

	0	1	2	3	4	5	6	7
0								
1								
2								
3								
4								
5								
6								
7								

# Goto

	0	1	2	3	4	5	6	7
0								
1								
2								
3								
4								
5								
6								
7								

# Goto

	0	1	2	3	4	5	6	7
0								
1								
2								
3								
4								
5								
6								
7								

# Goto

	0	1	2	3	4	5	6	7
0								
1								
2								
3								
4								
5								
6								
7								

# Goto

	0	1	2	3	4	5	6	7
0								
1								
2								
3								
4								
5								
6								
7								

# Goto

	0	1	2	3	4	5	6	7
0								
1								
2								
3								
4								
5								
6								
7								



# Goto

```
#include <stdio.h>

void delay()
{
    double start;
    while (start < 60000000)
    {
        start++;
    }
}

char letter = 'A';

char first_sentence[] = "HELLO";
char second_sentence[] = "FASHION SUIT";
char third_sentence[] = "SUITABLE PRICE";
```

# Goto

```
#include <stdio.h>
#include <stdbool.h>
#include <stdint.h>
#include <time.h>
#include <conio.h>
#include <unistd.h>

char first_sentence[] = "HI";
char second_sentence[] = "FASHION";
char third_sentence[] = "SUITABLE";

uint8_t letter_H[8][8] =
{
    {1,0,0,0,0,0,0,1},
    {1,0,0,0,0,0,0,1},
    {1,0,0,0,0,0,0,1}
```

# setjmp.h

setjmp.h là một thư viện trong ngôn ngữ lập trình C, cung cấp hai hàm chính là setjmp và longjmp. Cả hai hàm này thường được sử dụng để thực hiện xử lý ngoại lệ trong C, mặc dù nó không phải là một cách tiêu biểu để **xử lý ngoại lệ** trong ngôn ngữ này.

# setjmp.h

```
#include <stdio.h>
#include <setjmp.h>

jmp_buf buf;

int exception = 0;

void func2()
{
    printf("This is function 2\n");
    longjmp(buf, 2);
}

void func3()
{
    printf("This is function 2\n");
```

# setjmp.h

```
#include <stdio.h>
#include <setjmp.h>

jmp_buf buf;

int exception_code;

typedef enum
{
    NO_ERROR,
    NO_EXIT,
    DIVIDE_BY_0
} ErrorCodes;

#define TRY if ((exception_code = setjmp(buf)) == 0)
#define CATCH(x) else if (exception_code == x)
```

# setjmp.h

[https://docs.google.com/document/d/1xmmRqDHt0Ewai3tRQ-eVVE\\_VvHZ7kIF2GQvNv8Ayaps/edit](https://docs.google.com/document/d/1xmmRqDHt0Ewai3tRQ-eVVE_VvHZ7kIF2GQvNv8Ayaps/edit)