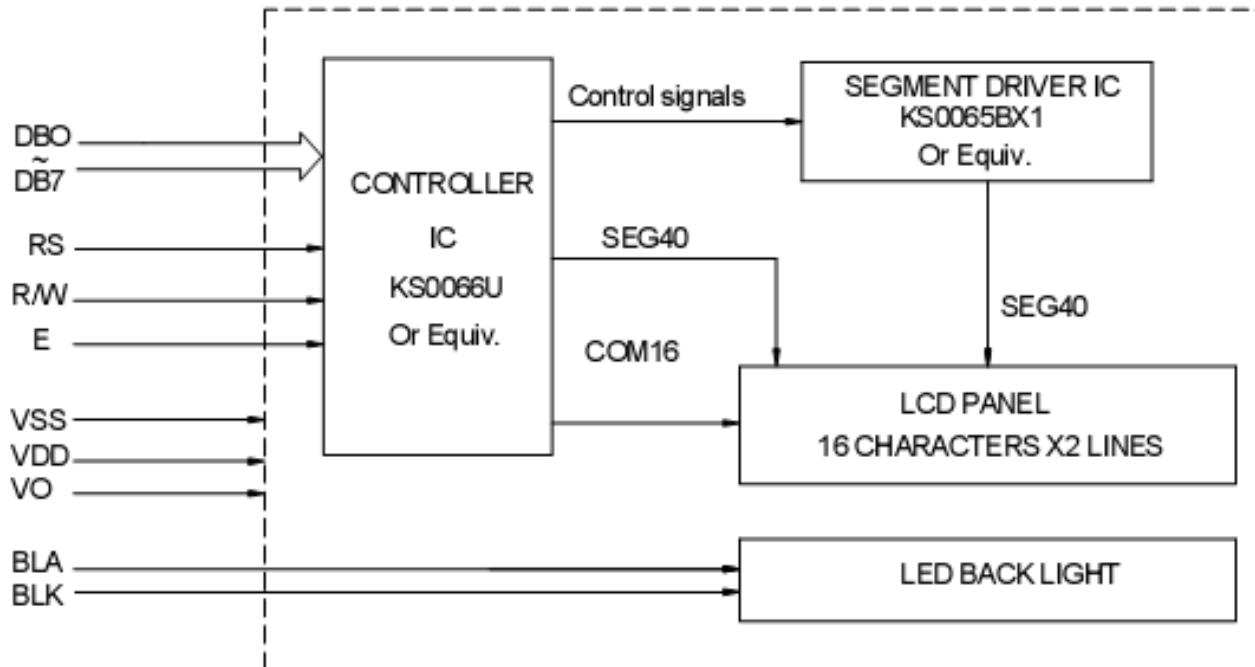


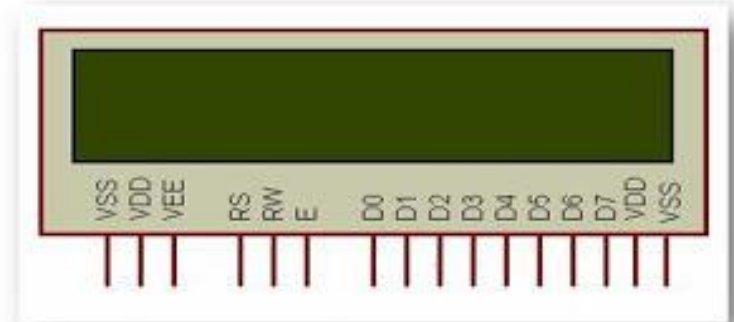
# Lập trình ghép nối text LCD

- LCD LM016L:
  - Display characters
  - 2 hàng, 16 cột
- Sơ đồ khối:

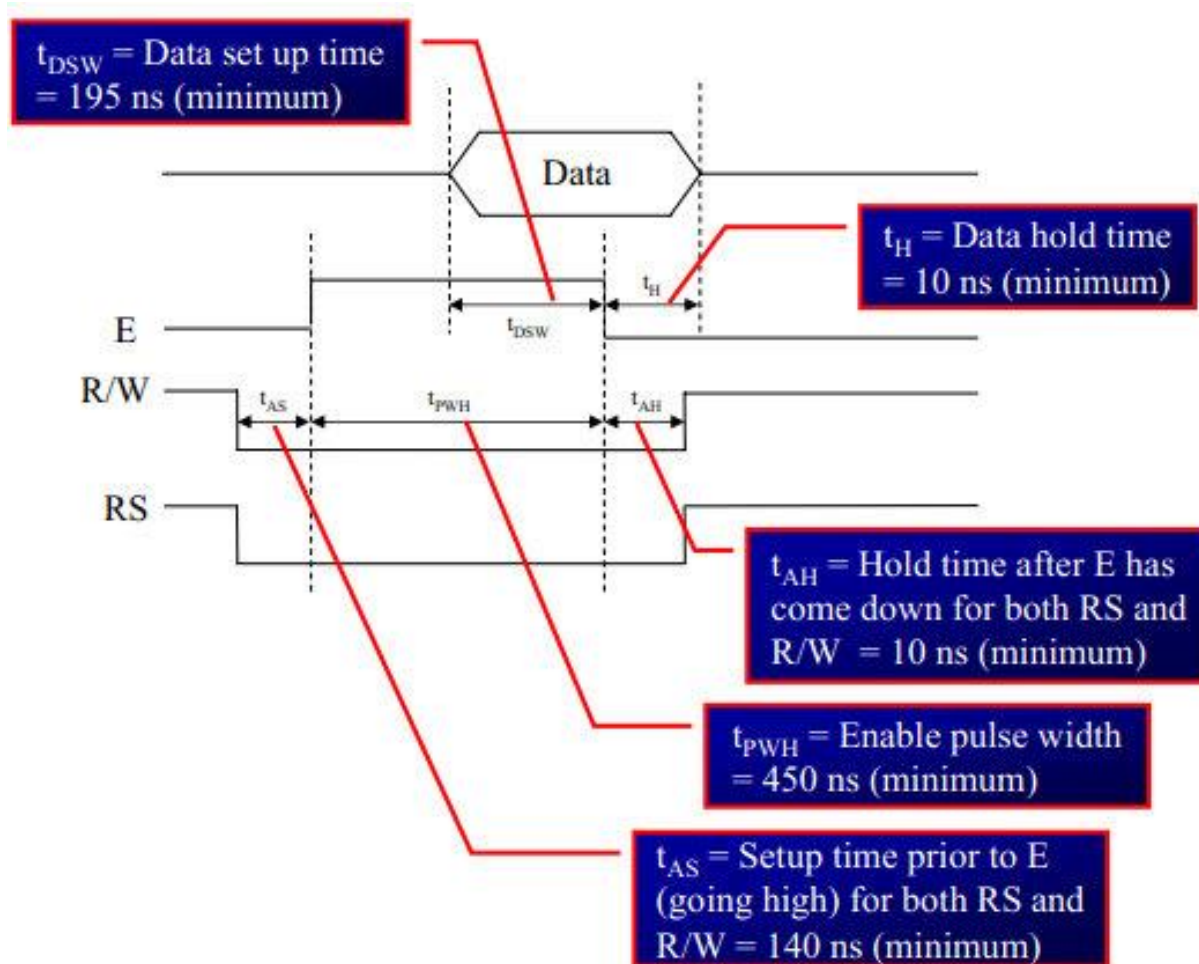


# Sơ đồ chân LCD 16x2

Pin Number	Symbol	Pin Function
1	VSS	Ground
2	VCC	+5v
3	VEE	Contrast adjustment (VO)
4	RS	Register Select. 0:Command, 1: Data
5	R/W	Read/Write R/W=0: Write R/W=1: Read
6	EN	Enable. Falling edge triggered
7	D0	Data Bit 0
8	D1	Data Bit 1
9	D2	Data Bit 2
10	D3	Data Bit 3
11	D4	Data Bit 4
12	D5	Data Bit 5
13	D6	Data Bit 6
14	D7	Data Bit 7/Busy Flag
15	A/LED+	Back-light Anode(+)
16	K/LED-	Back-Light Cathode(-)

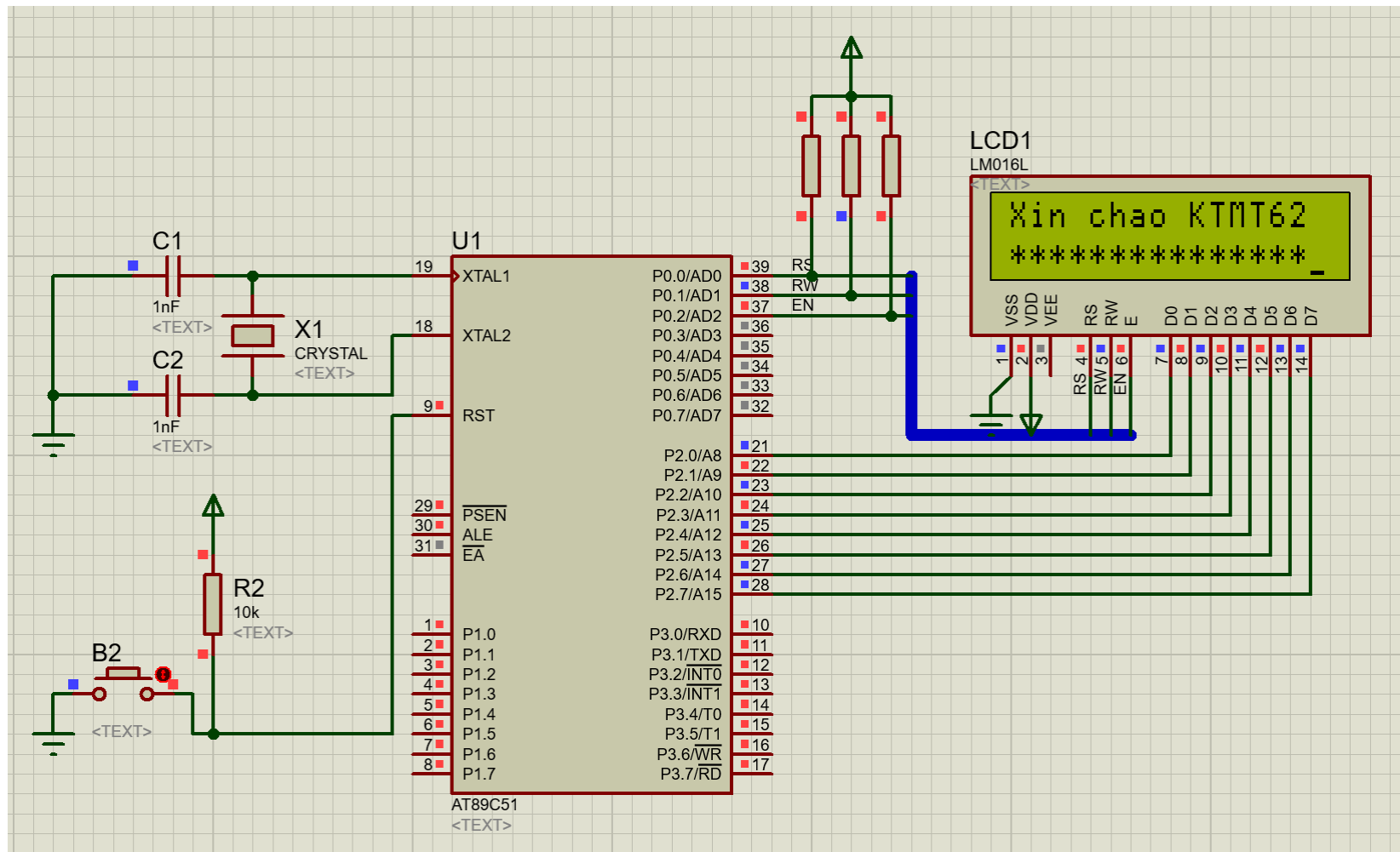


## ■ Timing diagram



# Các bước lập trình ghép nối LCD

- Bước 1: Thiết lập cấu hình làm việc cho LCD
  - $R/W = 0$  (write)
  - $RS = 0$  (command)
- Bước 2: Hiển thị ký tự trên LCD
  - Chọn vị trí hiển thị
  - $R/W = 0$  (write)
  - $RS = 1$  (data)
  - Send ASCII code to LCD
  - Send negative edge to LCD



# Hàm gửi lệnh điều khiển

```
void LCD_Send_Command(unsigned char x)
{
    LCD_DATA=x;
    RS=0; //Chon thanh ghi lenh
    RW=0; //De ghi du lieu
    EN=1;
    Delay_ms(1);
    EN=0;
    Wait_For_LCD(); //Doi cho LCD san sang
    EN=1;
}
```

```
void LCD_Write_One_Char(unsigned char c)
{
    LCD_DATA=c; //Dua du lieu vao thanh ghi
    RS=1; //Chon thanh ghi du lieu
    RW=0;
    EN=1;
    Delay_ms(1);
    EN=0;
    Wait_For_LCD();
    EN=1;
}
```

# Tập lệnh điều khiển LCD



## LCD Command Codes

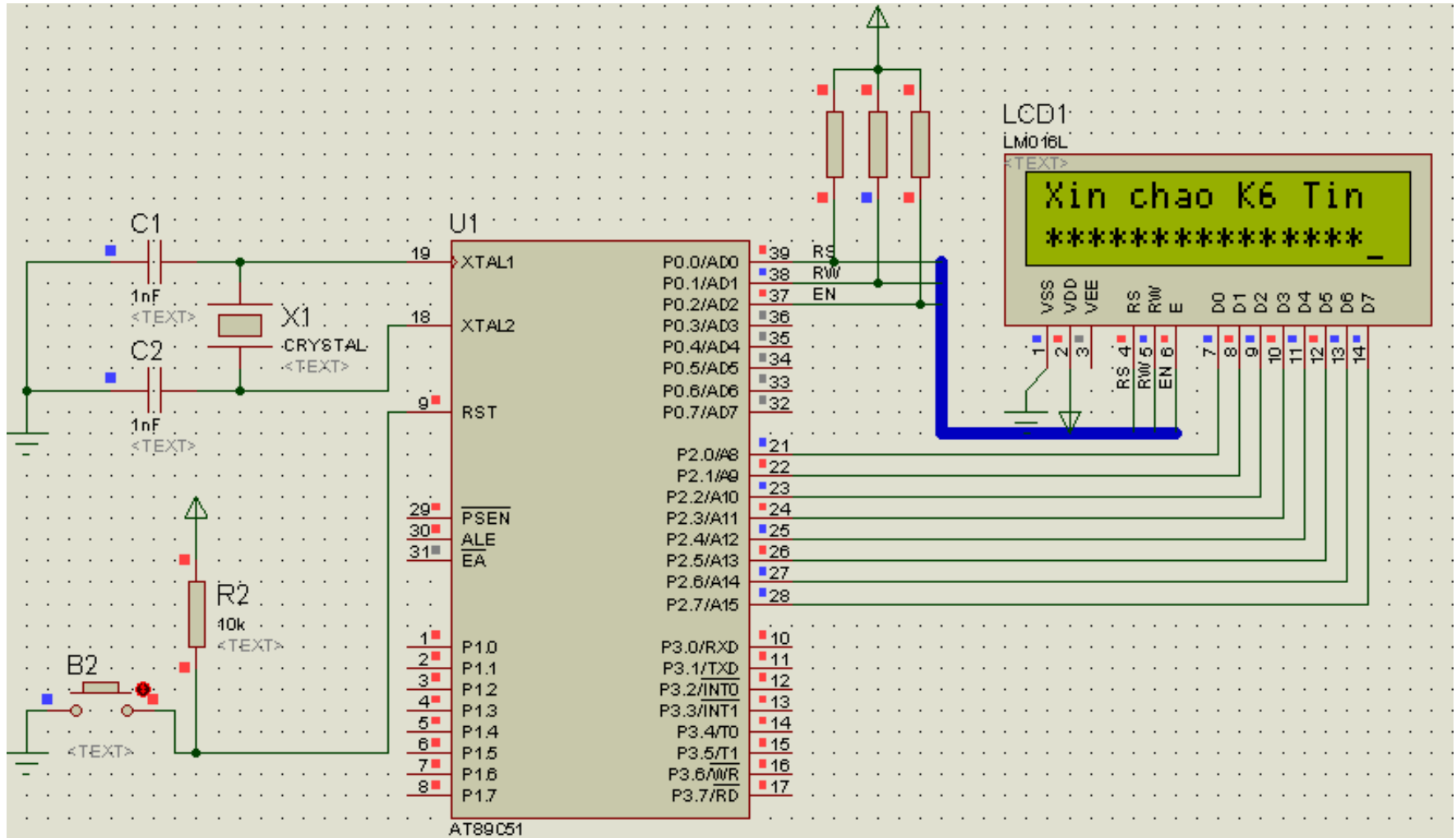
Code (Hex)	Command to LCD Instruction Register
1	Clear display screen
2	Return home
4	Decrement cursor (shift cursor to left)
6	Increment cursor (shift cursor to right)
5	Shift display right
7	Shift display left
8	Display off, cursor off
A	Display off, cursor on
C	Display on, cursor off
E	Display on, cursor blinking
F	Display on, cursor blinking
10	Shift cursor position to left
14	Shift cursor position to right
18	Shift the entire display to the left
1C	Shift the entire display to the right
80	Force cursor to beginning to 1st line
C0	Force cursor to beginning to 2nd line
38	2 lines and 5x7 matrix



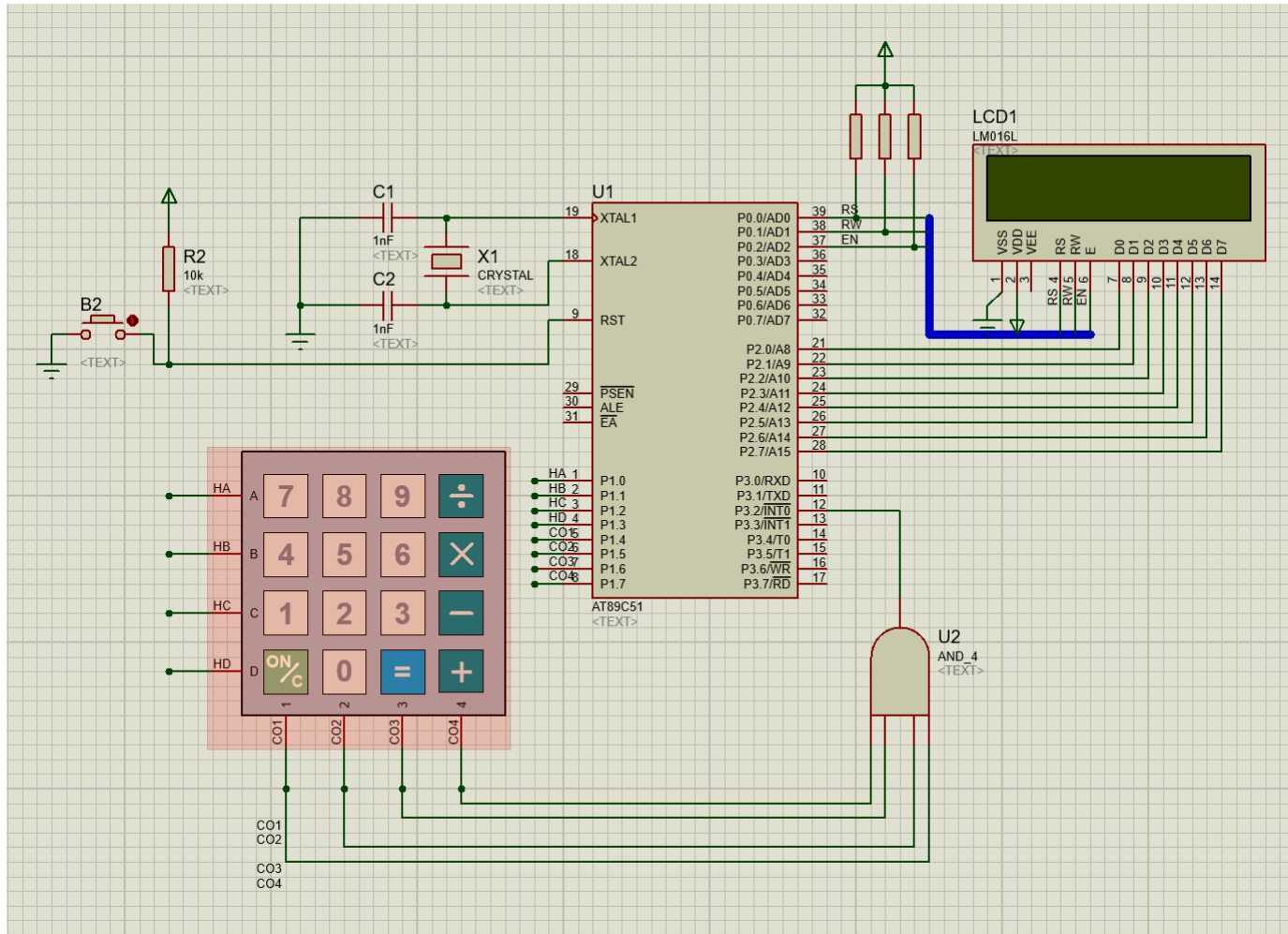
# Hàm khởi tạo LCD

```
void LCD_init()  
{  
    //Chọn chế độ 5x7 bit, 2 hàng cho LCD  
    LCD_Send_Command(0x38);  
    //Bật hiển thị, nhập nhảy con trỏ  
    LCD_Send_Command(0x0E);  
    LCD_Send_Command(0x01); //Xóa màn hình  
    LCD_Send_Command(0x80); //Vẽ dấu dòng  
}
```

# Ví dụ ghép nối LCD



- Ghép nối cả keypad và LCD với 8051
- Lập trình hiện số lên màn hình LCD dựa theo phím bấm trên keypad



# Ghép nối bảng quang báo

- Ứng dụng của bảng quang báo: bảng led quảng cáo, thông báo...



- Nguyên lý:
  - Bảng quang báo hoạt động dựa trên hiện tượng lưu ảnh trên mắt người
  - Các hàng/cột LED được quét tốc độ cao (giống quét LED 7 thanh)

# Bảng quang báo

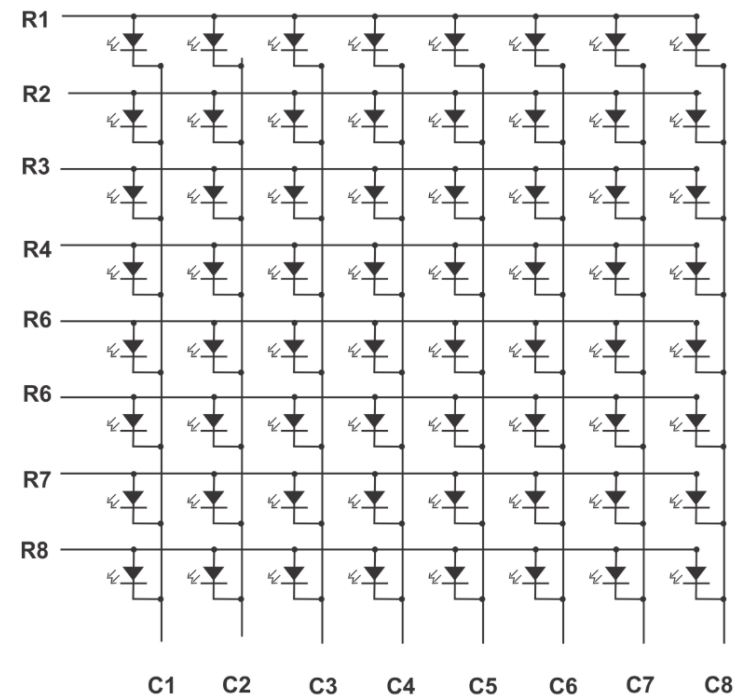
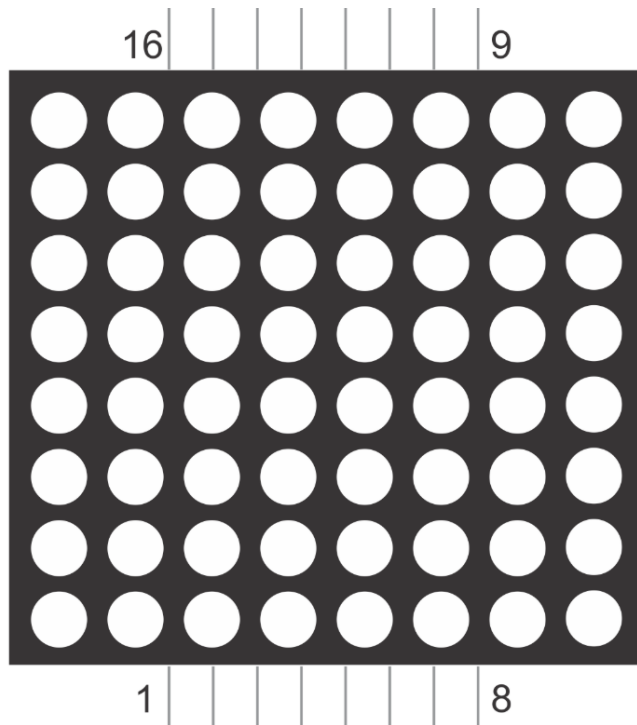
- Bảng quang báo kích thước nhỏ: một hoặc một vài module led ma trận 8x8



- Bảng quang báo kích thước lớn: hàng chục/hàng trăm module led được ghép nối với nhau
- Mỗi một cụm module led sẽ có vi điều khiển
- Các vi điều khiển tại các cụm sẽ được kết nối với nhau theo các chuẩn truyền tin: RS485, CAN, I2C, ...

# Module led ma trận

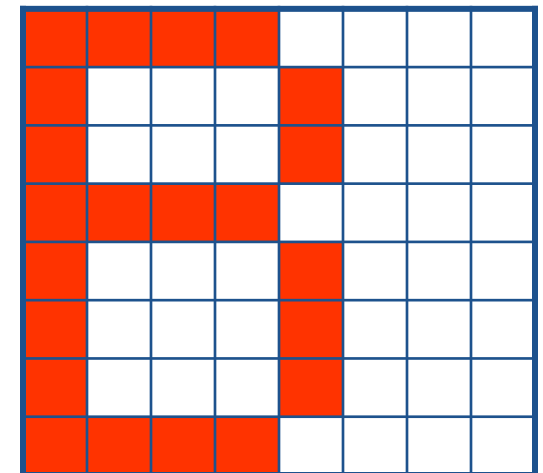
- Mỗi module led sẽ bao gồm các chân điều khiển hàng và điều khiển cột
- VD: module led 8x8 có 8 chân điều khiển hàng và 8 chân điều khiển cột



# Tạo font chữ

- Để hiển thị số 0 trên module led 8x8
  - Bước 1: Tạo font chữ (thường có kích thước 5x8 – chiều rộng 5 pixel và cao 8 pixel)
  - Bước 2: quét module LED theo từng hàng

0	1	1	1	0	0	0	0
1	0	0	0	1	0	0	0
1	0	0	0	1	0	0	0
1	0	0	0	1	0	0	0
1	0	0	0	1	0	0	0
1	0	0	0	1	0	0	0
1	0	0	0	1	0	0	0
0	1	1	1	0	0	0	0





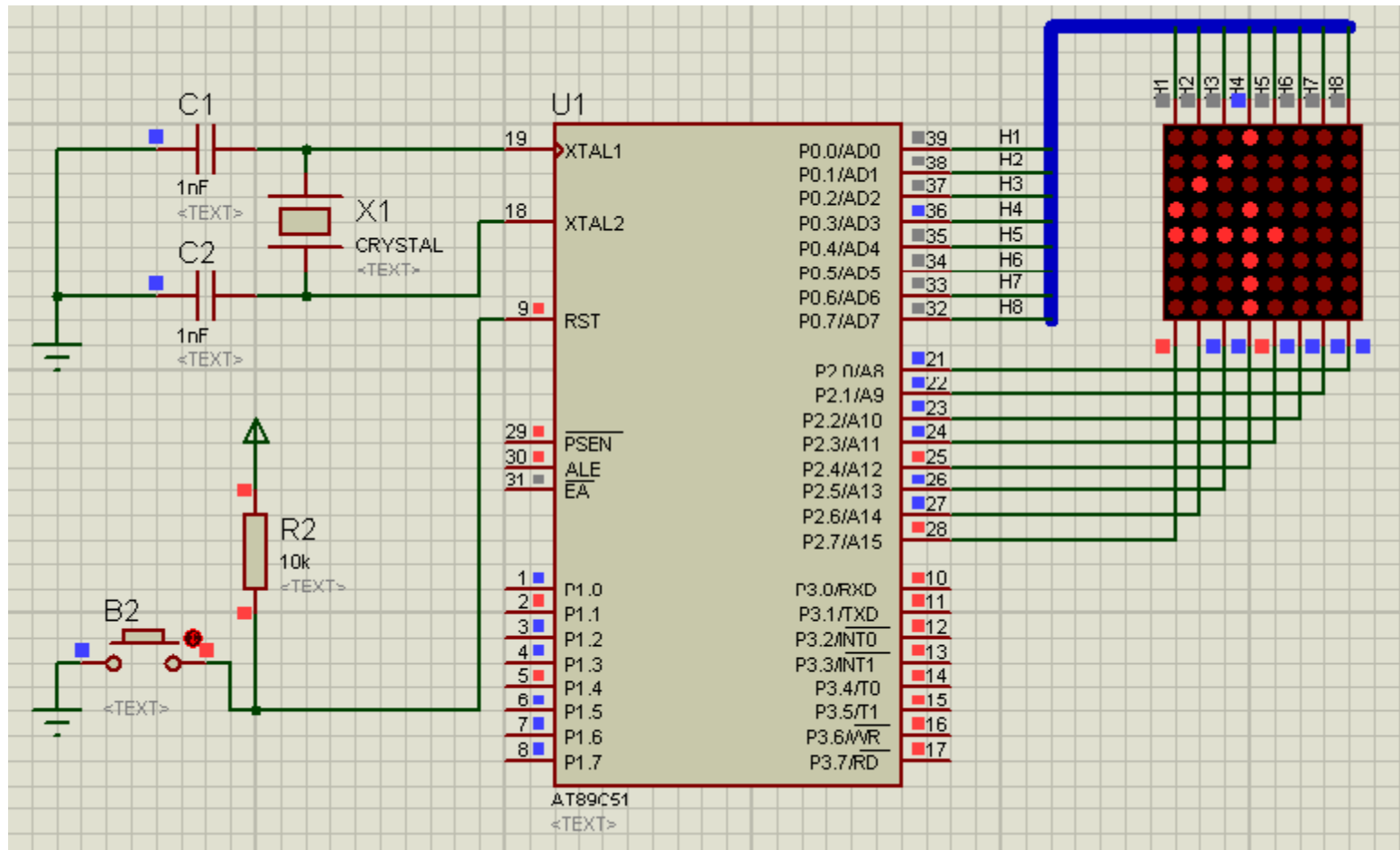
## ■ Tạo font chữ

0	0	1	0	0	0	0	0
0	1	0	1	0	0	0	0
1	0	0	0	1	0	0	0
1	0	0	0	1	0	0	0
1	1	1	1	1	0	0	0
1	0	0	0	1	0	0	0
1	0	0	0	1	0	0	0
1	0	0	0	1	0	0	0

1	1	1	1	0	0	0	0
1	0	0	0	1	0	0	0
1	0	0	0	1	0	0	0
1	1	1	1	0	0	0	0
1	0	0	0	1	0	0	0
1	0	0	0	1	0	0	0
1	0	0	0	1	0	0	0
1	1	1	1	0	0	0	0

# Minh họa giao tiếp led matrix

## ■ Sơ đồ ghép nối 1 led matrix



- Lập trình hiển thị chữ A lên ma trận LED