**GLCD**

1. **Sơ đồ nguyên lý**

* Các chân cơ bản kết nối giống LCD text đã học chỉ thêm một số chân mới sau:

+ PSB: Muốn chọn chế độ giao tiếp nối tiếp thì nối chân này với GND, ngược lại muốn giao tiếp song song thì nối chân này lên Vcc. Board đã nối sẵn chân này lên Vcc nên khi lập trình ta phải lập trình theo giao tiếp song song.

+ Chân RST tích cực mức thấp, nếu không dùng tới thì nối chân này lên nguồn.

+ Chân VOUT xuất ra điện áp âm, chân này được nối với chân 1 của biến trở 10K, chân 3 của biến trở nối GND, chân 2 nối vào chân số 3( chân chỉnh độ tương phản).

1. **Chế độ text**

* Chế độ này hoạt động tương tự như ở LCD text SV đã được học chỉ khác những điểm sau:  
  + Tổ chức vùng nhớ DDRAM 1 địa chỉ hiển thị được 2 ký tự và 1 hàng hiển thị được 16 ký tự.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0x80 | 0x80 | 0x81 | 0x81 | 0x82 | 0x82 | 0x83 | 0x83 | 0x84 | 0x84 | 0x85 | 0x85 | 0x86 | 0x86 | 0x87 | 0x87 |
| 0x90 | 0x90 | 0x91 | 0x91 | 0x92 | 0x92 | 0x93 | 0x93 | 0x94 | 0x94 | 0x95 | 0x95 | 0x96 | 0x96 | 0x97 | 0x97 |
| 0x88 | 0x88 | 0x89 | 0x89 | 0x8a | 0x8a | 0x8b | 0x8b | 0x8c | 0x8c | 0x8d | 0x8d | 0x8e | 0x8e | 0x8f | 0x8f |
| 0x98 | 0x98 | 0x99 | 0x99 | 0x9a | 0x9a | 0x9b | 0x9b | 0x9c | 0x9c | 0x9d | 0x9d | 0x9e | 0x9e | 0x9f | 0x9f |

+ Để sử dụng được chế độ text ta phải cấu hình Function Set là 0x30 và trong lập trình ta sử dụng chương trình con sau: SETUP\_GLCD (0x30);

* Các lệnh xóa màn hình, bật tắt con trỏ… chỉ có tác dụng đối với chế độ **Text**. Muốn sử dụng được các lệnh này thì ta phải cấu hình Function Set là 0x30 trước đó đã. Chú ý ta không thể xóa màn hình của chế độ Graphic bằng lệnh 0x01.

1. **Chế độ Graphic**

* Để sử dụng được chế độ Graphic ta phải khởi tạo Functin Set là 0x36 bằng lệnh SETUP\_GLCD (0x36);
* Để xóa màn hình chế độ Graphic ta dùng lệnh GLCD\_MAU\_NEN(0hoac1);. Khi dùng lệnh trên thì chỉ những nội dung liên quan đến chế độ Graphic mới được xóa còn nội dung của chế độ text thì không bị ảnh hưởng.
* Ta có thể sử dụng thư viện GRAPHICS.C trong thư mục cài đặt phần mềm để vẽ các đường cơ bản:

+ Đường thẳng: glcd\_line(x1, y1, x2, y2, color); - Cung cấp tọa độ 2 điểm sẽ vẽ được đoạn thẳng nối 2 điểm đó

+ Chữ nhật: glcd\_rect(x1, y1, x2, y2, fill, color);- Cung cấp tọa độ 2 đỉnh chéo góc sẽ vẽ ra hình chữ nhật

+ Thanh: glcd\_bar(x1, y1, x2, y2, width, color) ;– Là đường thẳng nhưng cho phép chọn độ dày width.

+ Tròn: glcd\_circle(x, y, radius, fill, color);- Cung cấp tọa độ tâm và bán kính sẽ vẽ được đường(hình) tròn

**Với :**

X,y: Là tọa độ cần hiển thị

Fill: Tô đậm bên trong hay không – 1 là tô đậm và 0 là không tô

Fill =0 =0

Fill =1

Color: Màu cần vẽ

* Sau khi vẽ xong lúc này màn hình vẫn chưa hiển thị ta phải gọi chương trình con GDRAM\_VDK\_TO\_GDRAM\_GLCD\_ALL() ;để xuất dữ liệu từ RAM ra màn hình- **SV hay rớt vì quên gọi chương trình này.**

1. **Bài mẫu về GLCD**

#include"E:\Teaching\Day TTVXL\NHOM\_1\_ST3\_THOAN\TV\_PICKIT2\_SHIFT\_1.c"

#include"E:\Teaching\Day TTVXL\NHOM\_1\_ST3\_THOAN\TV\_PICKIT2\_SHIFT\_GLCD128X64.c"

#include"C:\Program Files (x86)\PICC\Drivers\GRAPHICS.C"

void main()

{

SET\_UP\_PORT\_IC\_CHOT();

SETUP\_GLCD (0x30); // Khởi tạo ở chế độ Text

GLCD\_COMMAND(Lenh); // Các lệnh của chế độ text

GLCD\_DATA(kytu); // Hiển thị các ký tự cố định ngoài while(true)

SETUP\_GLCD (0x36); // Khởi tạo ở chế độ Graphic

GLCD\_MAU\_NEN(mau); // Xóa màu nền

// Các lệnh để vẽ những nội dung cố định

GDRAM\_VDK\_TO\_GDRAM\_GLCD\_ALL() ; // Cho phép hiển thị ra màn hình

while(true)

{

// SETUP\_GLCD (0x30); // Khởi tạo ở chế độ text

// Các lệnh chế độ text để hiển thị những nội dung text thay đổi

//SETUP\_GLCD (0x36); // Khởi tạo ở chế độ Graphic

// Các lệnh chế độ Graphic để cập nhật những hình ảnh thay đổi

// GDRAM\_VDK\_TO\_GDRAM\_GLCD\_ALL() ; // Cập nhật hình ảnh đã thay đổi

}

}

* **Chú ý**: Chữ màu tím để chỉ những lệnh liên quan đến chế độ text, chữ màu xanh lá là của chế độ Graphic còn chữ màu đỏ là bắt buộc phải đó khi dùng GLCD