P30004

64x32 White OLED
Application Notes
(for IIC Interface)

Revision History

Version	REVISION DESCRIPTION
X01	First release

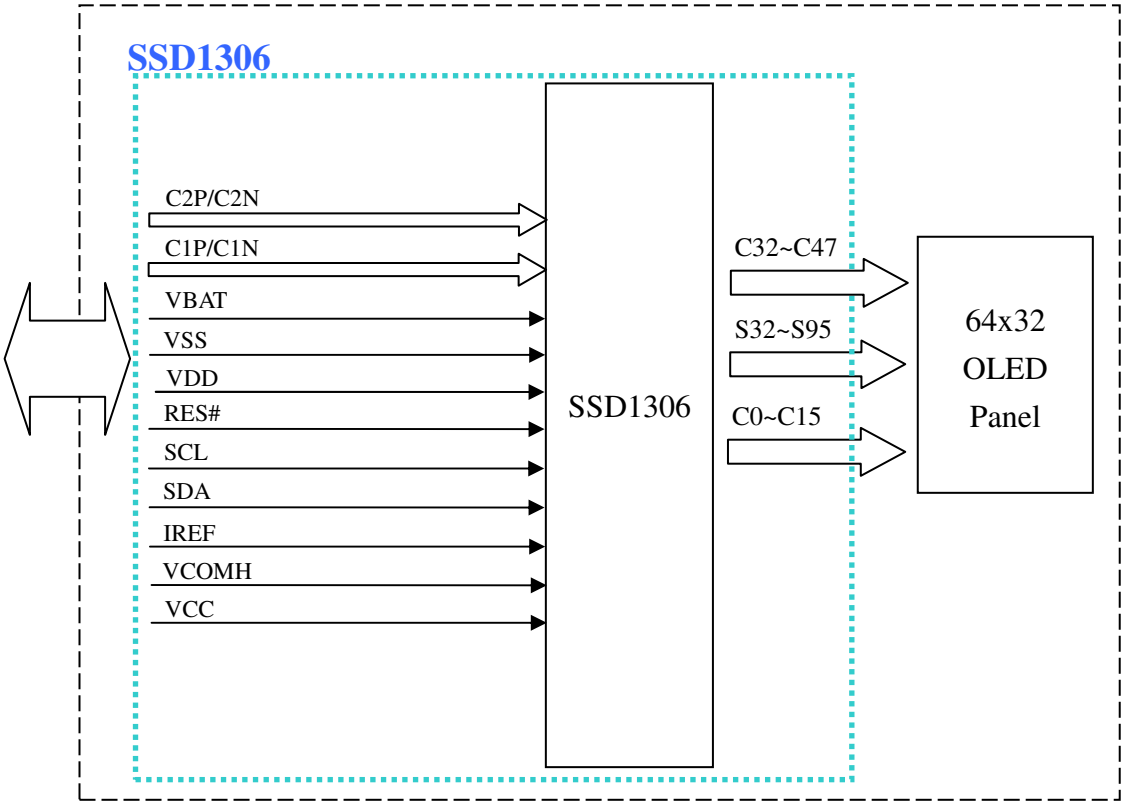
DESCRIPTION

P30004 is a 64x32 dot matrix white passive OLED module with controller for many compact portable applications.

FEATURE

- Panel matrix 64x32
- Driver IC: SSD1306
- VBAT=3.5~4.2V(For Charge Pump)
- VDD = 1.65~3.3V
- I²C Interface
- Embedded 128 x 64 bit SRAM display buffer.
- Screen saving continuous scrolling function in both horizontal and vertical direction.
- On-Chip Oscillator.
- Row Re-mapping and Column Re-mapping.

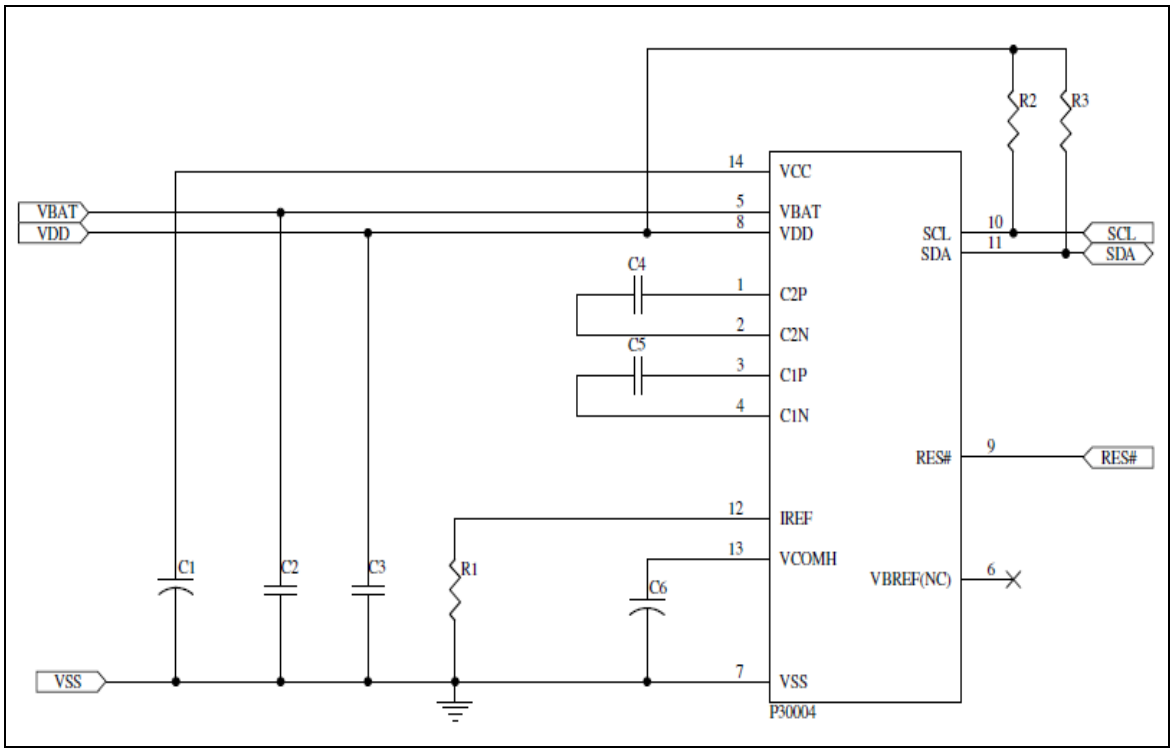
FUNCTION BLOCK DIAGRAM



RITdisplay 64X32 OLED Module

APPLICATION CIRCUIT

(Charge Pump)



Recommended components：

C1,C6: 4.7uF/16V(0805)

C2,C3,C4,C5: 1uF/16V(0603)

R1: 560Kohm (0603) 1%

R2, R3: 10K ohm(0603)

This circuit is for I^2C interface.

PIN ASSIGNMENTS

PIN NAME	PIN NO.	DESCRIPTION	Setting at each interface		
			8080 parallel	SPI	IIC
1	C2P	C2P/C2N – Pin for charge pump capacitor; Connect to each other with a capacitor.			
2	C2N				
3	C1P	C1P/C1N – Pin for charge pump capacitor; Connect to each other with a capacitor.			
4	C1N				
5	VBAT	Power supply for charge pump regulator circuit.			
6	VBREF(NC)	No connection.			
7	VSS	Ground pin.			
8	VDD	Power supply pin for core logic operation.			
9	RES#	This pin is reset signal input.	NA	NA	RES#
10	SCL	I^2C serial clock pin.	NA	NA	SCL
11	SDA	I^2C serial data pin.	NA	NA	SDA
12	IREF	This pin is the segment output current reference pin. A resistor should be connected between this pin and VSS.			
13	VCOMH	COM signal deselected voltage level. A capacitor should be connected between this pin and VSS.			
14	VCC	Power supply for panel driving voltage.			

Note

(1) Low is connected to V_{SS}

(2) High is connected to V_{DD}

Application Initial Setting

/*64x32 OLED driver program (For Charge Pump) */

/* The more detail of I^2C sequence please refer to the SSD1306 datasheet */

```
void initial(void)
{
start ( );
comm_out(0x78);//mast code
comm_out(0x00);

comm_out(0xae);//Set Display OFF

comm_out(0x40);//Set Display Start Line

comm_out(0x81);//Set Contrast Control
comm_out(0x75);

comm_out(0xa1);//Set Segment Re-map

comm_out(0xa4);//Resume to RAM content display

comm_out(0xa6);//Set Normal Display

comm_out(0xa8);//Set Multiplex Ratio
comm_out(0x1f);

comm_out(0xc8);//Set COM Output Scan Direction

comm_out(0xd3);//Set Display Offset
comm_out(0x00);

comm_out(0xd5);//Set Display Clock Divide Ratio/Oscillator Frequency
comm_out(0xc1);

comm_out(0xd9);//Set Pre-charge Period
comm_out(0xf1);
```

```
comm_out(0xda);//Set COM Pins Hardware Configuration
comm_out(0x12);

comm_out(0xdb);//Set VCOMH Deselect Level
comm_out(0x40);

comm_out(0x8d);//Charge Pump Setting
comm_out(0x14);//Enable Charge Pump

stop ( );

cleanDDR(); //Clear the whole DDRAM

start ( );

comm_out(0x78);//mast code
comm_out(0x00);

comm_out(0xaf);//Set Display ON

stop ( );
}
```

```
void cleanDDR(void)
{
int i,j;
for(i=0;i<8;i++)
      {
      start ( );
      comm_out(0x78);//mast code
      comm_out(0x00);

      comm_out(0xb0+i); //Set Page
      comm_out(0x00); //Lower Column Address
      comm_out(0x10); //Higher Column Address

      stop ( );
      start ( );
      comm_out(0x78);//mast code
      comm_out(0x40);
      for(j=0;j<128;j++)
            {
            data_out(0x00);
            }
            stop ( );
      }
}
```

After initial the driver IC, user can display all pixels on.

```
void show_data(void)
{
int i,j;
for(i=0;i<4;i++)
      {
      start ( );
      comm_out(0x78);//mast code
      comm_out(0x00);

      comm_out(0xb0+i); //Set Page
      comm_out(0x00); //Lower Column Address
      comm_out(0x12); //Higher Column Address
```

```
        stop ( );
        start ( );
        comm_out(0x78);//mast code
        comm_out(0x40);
        for(j=0;j<64;j++)
              {
              data_out(0xff);
              }
              stop ( );
        }
}
```

Power ON / OFF Sequence

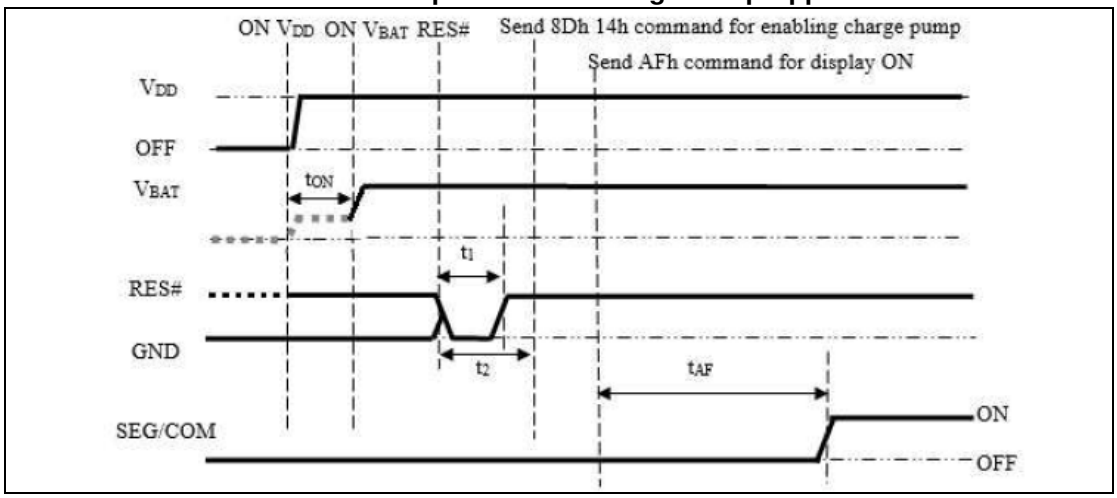
Power ON and OFF sequence with Charge Pump Application

The following figures illustrate the recommended power ON and power OFF sequence of SSD1306 with charge pump application.

Power ON sequence **:**

1. Power ON V_{DD}
2. Wait for t_{ON}. Power ON V_{BAT}.[1] (where Minimum t_{ON} = 0ms)
3. After V_{BAT} become stable, set RES# pin LOW (logic low) for at least 3us (t_1) [4] and then HIGH (logic high).
4. After set RES# pin LOW (logic low), wait for at least 3us (t_2). Then input commands with below sequence:
   a. 8Dh 14h for enabling charge pump at 7.5V mode
   b. AFh for display ON
5. SEG/COM will be ON after 100ms (t_{AF}).

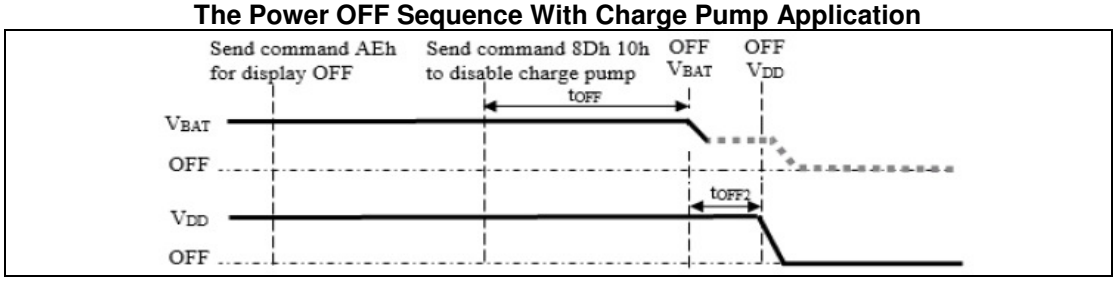
The Power ON Sequence With Charge Pump Application



Power OFF sequence **:**

1. Send command AEh for display OFF
2. Send command 8Dh 10h to disable charge pump
3. Power OFF V_{BAT} after t_{OFF}. .[1], [2], [3] (Typical t_{OFF} =100ms)
4. Power OFF V_{DD} after t_{OFF2}. (where Minimum t_{OFF2} = 0ms [5], Typical t_{OFF2}=5ms)

The Power OFF Sequence With Charge Pump Application



Note:

(1) Since an ESD protection circuit is connected between V_{DD} and V_{BAT}, V_{BAT} becomes lower than V_{DD} whenever V_{DD} is ON and V_{BAT} is OFF as shown in the dotted line of V_{BAT}.

(2)V_{BAT} should be kept float (i.e. disable) when it is OFF.

(3) Power Pins (V_{BAT}) can never be pulled to ground under any circumstance.

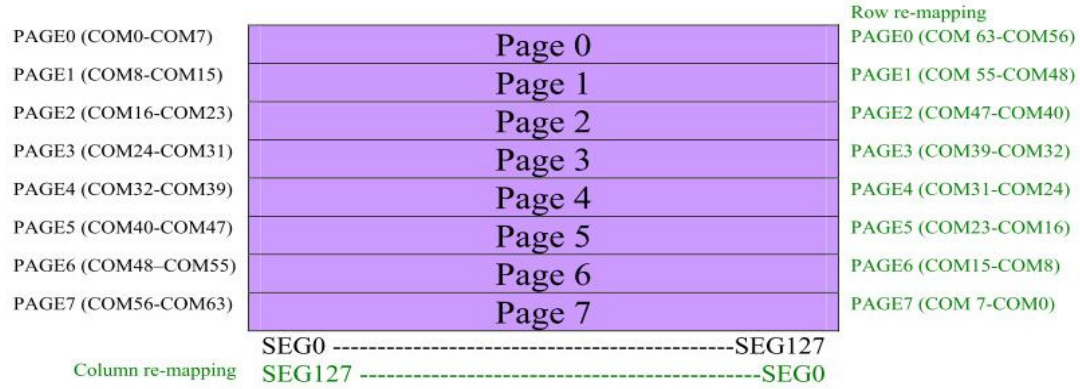
(4) The register values are reset after t_1.

(5) V_{DD} should not be Power OFF before V_{BAT} Power OFF

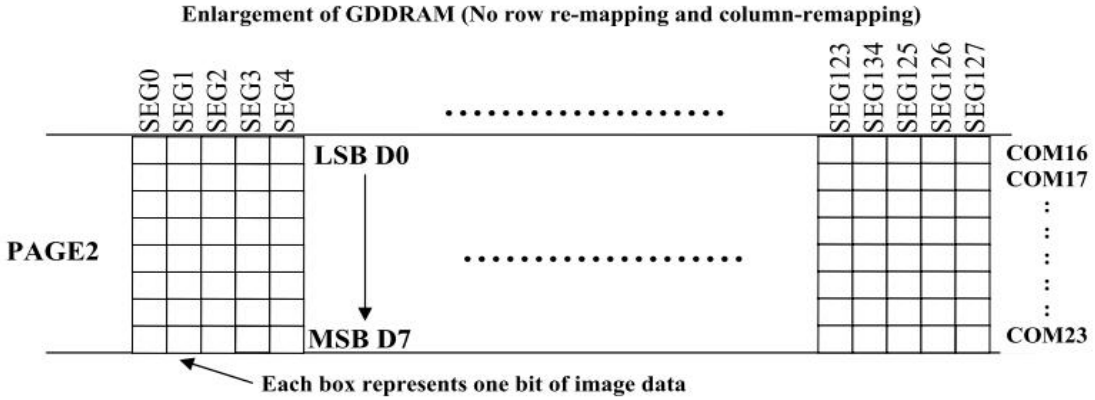
Graphic Display Data RAM Address Map

The GDDRAM is a bit mapped static RAM holding the bit pattern to be displayed. The size of the RAM is 128 x 64 bits and the RAM is divided into eight pages, from PAGE0 to PAGE7, which are used for monochrome 128x64 dot matrix display, as shown in below figures.

GDDRAM pages structure of SSD1306



When one data byte is written into GDDRAM, all the rows image data of the same page of the current column are filled (i.e. the whole column (8 bits) pointed by the column address pointer is filled.). Data bit D0 is written into the top row, while data bit D7 is written into bottom row as shown in below figures.



For mechanical flexibility, re-mapping on both Segment and Common outputs can be selected by software.

For vertical shifting of the display, an internal register storing the display start line can be set to control the portion of the RAM data to be mapped to the display (command D3h).

Thank You