**LT768x**

***High Performance TFT-LCD Graphics Controller***

**Application Notes for GCBASIC**

**V0.9a**

**Directory**

[1. Preface 6](#_Toc154040033)

[1.1 System Architecture 7](#_Toc154040034)

[2. Model Options 9](#_Toc154040035)

[3. Reset 12](#_Toc154040036)

[3.1 Power-on Reset 12](#_Toc154040037)

[3.2 External Reset 12](#_Toc154040038)

[3.3 Software Reset 13](#_Toc154040039)

[3.4 Test Signals 13](#_Toc154040040)

[4. Geometric Drawing 14](#_Toc154040041)

[4.1 Drawing Line 14](#_Toc154040042)

[4.1.1 Drawing a Thin Line 14](#_Toc154040043)

[4.1.2 Drawing a Thick Line 14](#_Toc154040044)

[4.2 Drawing Circle 15](#_Toc154040045)

[4.2.1 Drawing a Hollow Circle 15](#_Toc154040046)

[4.2.2 Drawing a Solid Circle 15](#_Toc154040047)

[4.2.3 Drawing a Solid Circle with Frame 15](#_Toc154040048)

[4.3 Drawing Ellipse 17](#_Toc154040049)

[4.3.1 Drawing a Hollow Ellipse 17](#_Toc154040050)

[4.3.2 Drawing a Solid Ellipse 17](#_Toc154040051)

[4.3.3 Drawing a Solid Ellipse with Frame 17](#_Toc154040052)

[4.4 Drawing Rectangle 19](#_Toc154040053)

[4.4.1 Drawing a Hollow Rectangle 19](#_Toc154040054)

[4.4.2 Drawing a Solid Rectangle 19](#_Toc154040055)

[4.4.3 Drawing a Solid Rectangle with Frame 19](#_Toc154040056)

[4.5 Drawing Rounded-Rectangle 21](#_Toc154040057)

[4.5.1 Drawing a Hollow Rounded-Rectangle 21](#_Toc154040058)

[4.5.2 Drawing a Solid Rounded-Rectangle 21](#_Toc154040059)

[4.5.3 Drawing a Rounded-Rectangle with Frame 22](#_Toc154040060)

[4.6 Drawing Triangle 23](#_Toc154040061)

[4.6.1 Drawing a Hollow Triangle 23](#_Toc154040062)

[4.6.2 Drawing a Solid Triangle 23](#_Toc154040063)

[4.6.3 Drawing a Solid Rectangle with Frame 24](#_Toc154040064)

[4.7 Drawing Curve 25](#_Toc154040065)

[4.7.1 Drawing an Upper-Left Curve 25](#_Toc154040066)

[4.7.2 Drawing a Lower-Left Curve 25](#_Toc154040067)

[4.7.3 Drawing an Upper-Right Curve 26](#_Toc154040068)

[4.7.4 Drawing a Lower-Right Curve 26](#_Toc154040069)

[4.8 Drawing 1/4 Ellipse 27](#_Toc154040070)

[4.8.1 Drawing an Upper-Left 1/4 Ellipse 27](#_Toc154040071)

[4.8.2 Drawing an Lower-Left 1/4 Ellipse 27](#_Toc154040072)

[4.8.3 Drawing an Upper-Right 1/4 Ellipse 28](#_Toc154040073)

[4.8.4 Drawing an Lower-Right 1/4 Ellipse 28](#_Toc154040074)

[4.9 Drawing Quadrilateral 29](#_Toc154040075)

[4.9.1 Drawing a Hollow Quadrilateral 29](#_Toc154040076)

[4.9.2 Drawing a Solid Quadrilateral 29](#_Toc154040077)

[4.10 Drawing Pentagonal 31](#_Toc154040078)

[4.10.1 Drawing a Hollow Pentagonal 31](#_Toc154040079)

[4.10.2 Drawing a Solid Pentagonal 31](#_Toc154040080)

[4.11 Drawing Cylinder 33](#_Toc154040081)

[4.12 Drawing Cube 34](#_Toc154040082)

[4.13 Drawing Table 35](#_Toc154040083)

[5. Function Library List 37](#_Toc154040084)

index.pngindex.pngVersion History ……………………………...…………………..………… 40

Copyright Notice ……………………………….…………………..……… 40

# Preface

This application note is based upon the application note provided by Levetop. Levetop have supported the development of the GCBASIC library and Levetop have been so supportive.

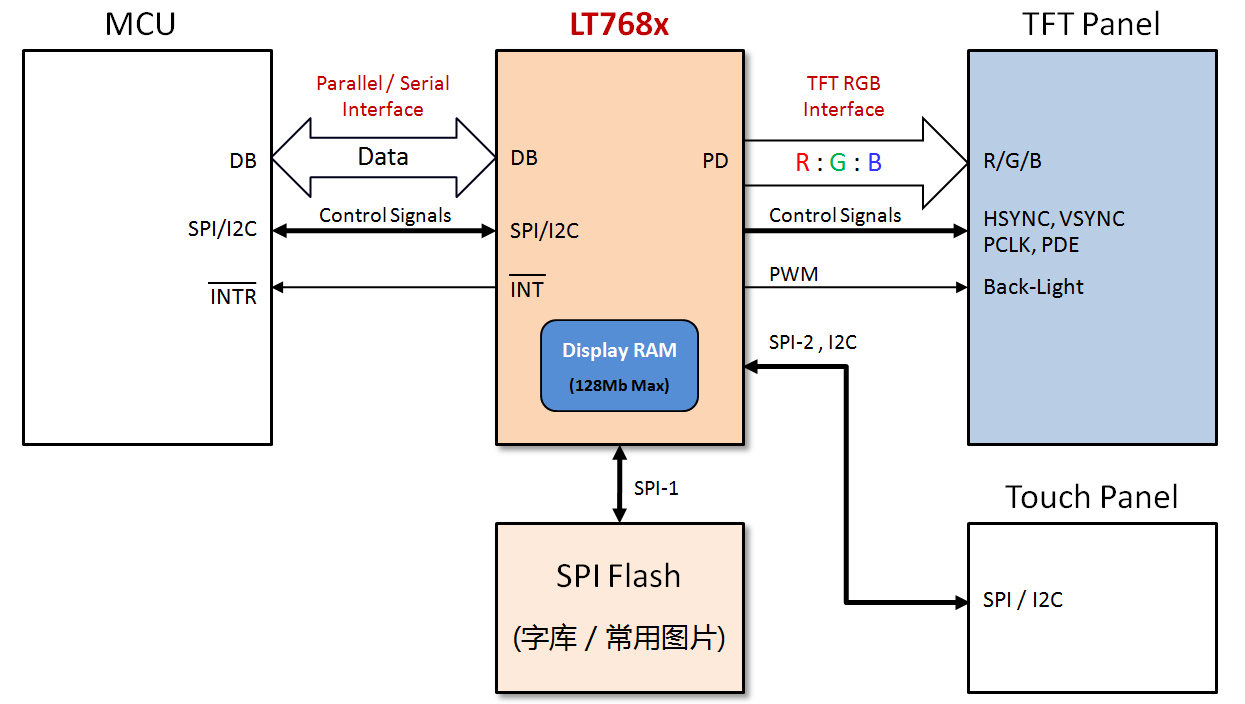
This application note mainly describes the hardware interface, software libraries of LT768x, and the realization of the internal function. In the meantime, by utilizing the demo program, function library, and schematic diagram, customers can quickly set up the design environment of LT768x and develop their application with the TFT panel product. This manual can help users to get a quick start and shorten their development time on exploring LT768x.

## System Architecture

To display an image on TFT panel, the image data must first be converted into electrical signals, and then the TFT driver transmits and continuously updates(scan) these signals to the LCD panel. As a result of the continuous transmission of scanned data, and the visual retention effect of human eyes, the image displayed on the TFT panel will be stable and complete as seen by human eyes. However, since the TFT driver does not have data store functions, it must continuously get the image data from the system side (such as MCU). The main function of the TFT controller is to assist the system to provide the image data to the TFT driver, and keep uploading the image data to the TFT panel continuously.

LT768x is a high-performance TFT-LCD graphics accelerated display controller. In addition to assist the MCU transfering display data to TFT driver as mentioned above, LT768x also provides 2D graphics acceleration, PIP (picture-in-picture), geometric graphics and other functions. In order to reduce the time taken by MCU to transmit image data, LT768x provides a SPI Master Interface to retrieve image data from SPI Flash through DMA transfer mode, and then save the data to the embedded display RAM of LT768x. SPI Flash is used to save image data such as pictures, fonts etc. LT768x will then transmit the specified display memory data of the selected display window through the RGB interface to the driver inside the external TFT panel. Therefore, LT768x not only enhance the display performance, but also greatly alleviate the processing burden on MCU.

Even if an 8bits MCU is used as the host, it can work well with a TFT display if LT768x is applied properly. The following diagram is the basic application architecture for LT768x:



**Figure 1-2: System Architecture of LT768x**

LT768x provides 2 sets of SPI master interfaces and 1 set of I2C interfaces. If the system wants to provide a touch screen function, the interface of touch controller can either connect to I2C or SPI interface of LT768x, and then the MCU can read or control the touch controller through LT768x. This will simplifies the interface of touch controller connection.

# Model Options

**Table 2-1: LT768x Model Options**

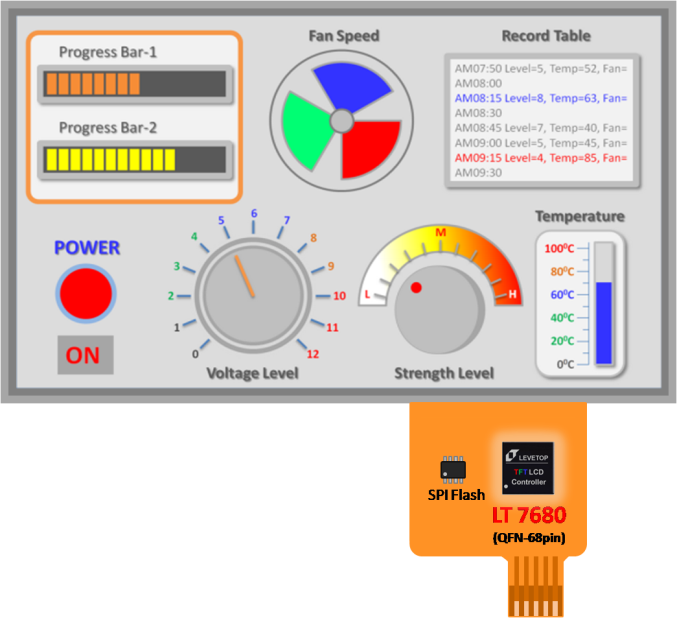
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Support Features** | | **LT7680A** | **LT7680B** | **LT7681** | **LT7683+** | **LT7686** |
| **Items** | **Function** |
| **Package** | **LQFP / QFN** | QFN-68 | QFN-68 | LQFP-128 | LQFP-128 | LQFP-128 |
| **LCD Spec.** | **Resolution/Max.** | 800\*600 | 480\*320 | 640\*480 | 1024\*768 | 1280\*1024 |
| **Colors** | 262K | 262K | 16.7M | 16.7M | 16.7M |
| **TFT Interface** | RGB  (18bits/Max) | RGB  (18bits/Max) | RGB  (24bits/Max) | RGB  (24bits/Max) | RGB  (24bits/Max) |
| **Display RAM** | **Embed. RAM Size** | 64Mbit | 64Mbit | 128Mbit | 128Mbit | 128Mbit |
| **Layers** | 7 Layers/Min | 12 Layers /Min | 18 Layers/Min | 11 Layers /Min | 4 Layers /Min |
| **MCU Interface** | **8080 8bit** |  |  | V | V | V |
| **6800 8bit** |  |  | V | V | V |
| **8080 16bit** |  |  | V | V | V |
| **6800 16bit** |  |  | V | V | V |
| **3 wire SPI** | V | V | V | V | V |
| **4 wire SPI** |  |  | V | V | V |
| **I2C** | V | V | V | V | V |
| **Others Interface** | **SPI Master (DMA Flash)** | V (2) | V (2) | V (2) | V (2) | V (2) |
| **I2C Master** |  |  | V | V | V |
| **PWM O/P** | 2 | 2 | 2 | 2 | 2 |
| **GPIO O/P** | 7/max | 7/max | 28/max | 28/max | 28/max |
| **Smart Key-scan** |  |  | 5\*5 | 5\*5 | 5\*5 |
| **Graphics Features** | **2D BTE Engine** | V | V | V | V | V |
| **Geometric Drawing Engine** | V | V | V | V | V |
| **Polygon Drawing** | V | V | V | V | V |
| **Picture in Picture (PIP)** | V | V | V | V | V |
| **Virtual Display** | V | V | V | V | V |
| **Vertical Scrolling** | V | V | V | V | V |
| **Horizontal Scrolling** | V | V | V | V | V |
| **Mirror and Rotation** | V | V | V | V | V |
| **Alpha-Blending** | V | V | V | V | V |
| **Graphic Cursor** | V | V | V | V | V |
| **Power-on Display** | V | V | V | V | V |
| **Color-Bar Test** | V | V | V | V | V |
| **Text Features** | **Embed. ISO/IEC 8859** | ISO8259 | ISO8259 | ISO8259 | ISO8259 | ISO8259 |
| **Ext. Font**  **(ext. SPI Flash)** | V | V | V | V | V |
| **Text Enlargement** | 4\*4 | 4\*4 | 4\*4 | 4\*4 | 4\*4 |
| **Text Rotate** | V | V | V | V | V |
| **Text Cursor** | V | V | V | V | V |
| **User-defined Character** | V | V | V | V | V |
| **Power** | **Sleep Mode (Standby/Suspend/Sleep)** | V | V | V | V | V |
| **Power Source** | 3.3V | 3.3V | 3.3V | 3.3V | 3.3V |

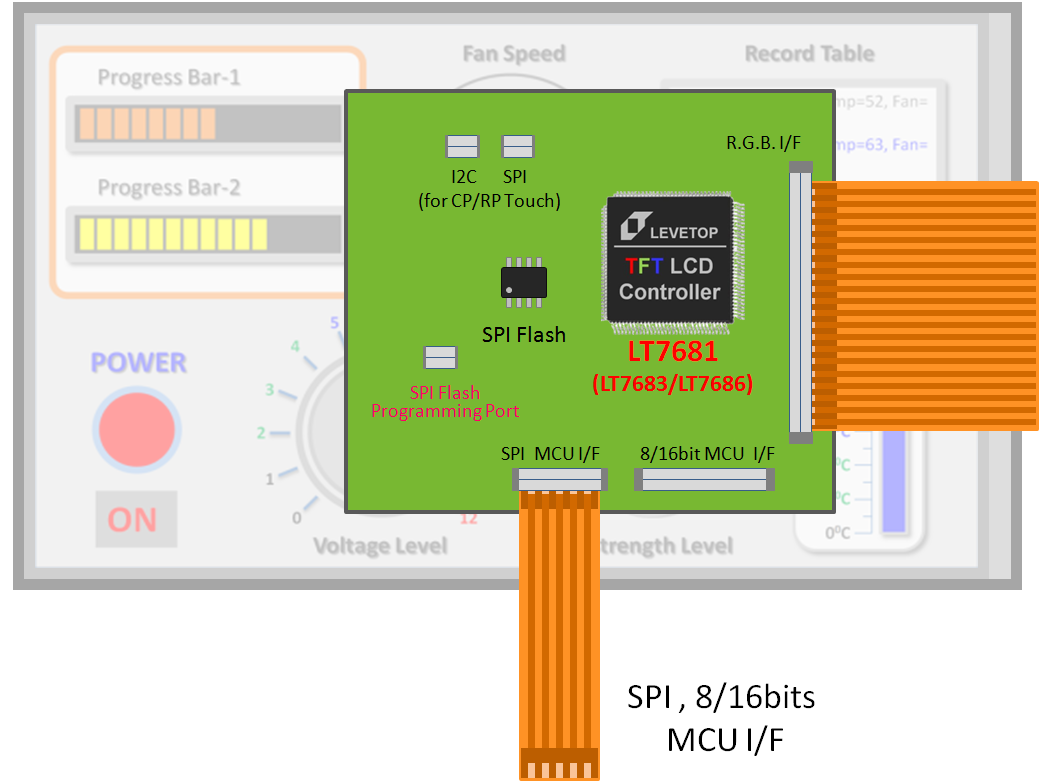
**Figure 2-1: LT768x Series**

The LT768x of the same package are compatible. For example, LT7681, LT7683+ and LT7686 are all 128Pin LQFP packages, and their pin-assignments are compatible. The LT7680A and LT7680B are 68Pin QFN package, their pin-assignment are also compatible. The resolution is backward compatible. For example, the resolution of the LT7686 is 1280\*1024, it can also be used on the lower resolution of the TFT screen.

LT7680 is a 68Pin QFN (8mm\*8mm) package with a smaller dimension. It can be used in the system PCB board or LCM PCB. It can also be welded on the FPC to form a standard TFT module with SPI interface, as shown in Figure 2-2 below. The designers can also integrate a standard TFT module and a LT768x control board to form a complete module so that most 8/16/32bits MCU can connect to this complete module directly, as shown in Figure 2-3.

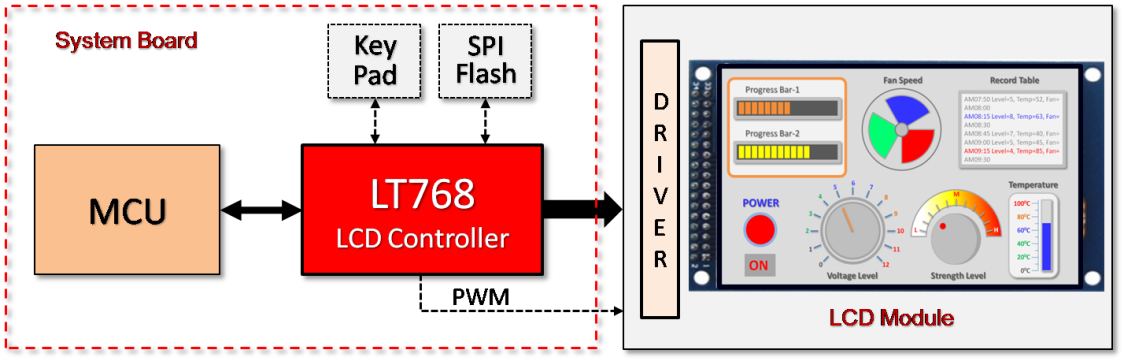
****

**Figure 2-2: Standard SPI Interface TFT Module with LT7680**

****

**Figure 2-3: Standard TFT Module with LT7681/7683+/7686**

LT768x is controlled by the MCU so it can also be placed in the system board, and then connect with the standard RGB type TFT Panel. Show as below Figure 2-4.

****

**Figure 2-4: Design-in LT768x on System Board**

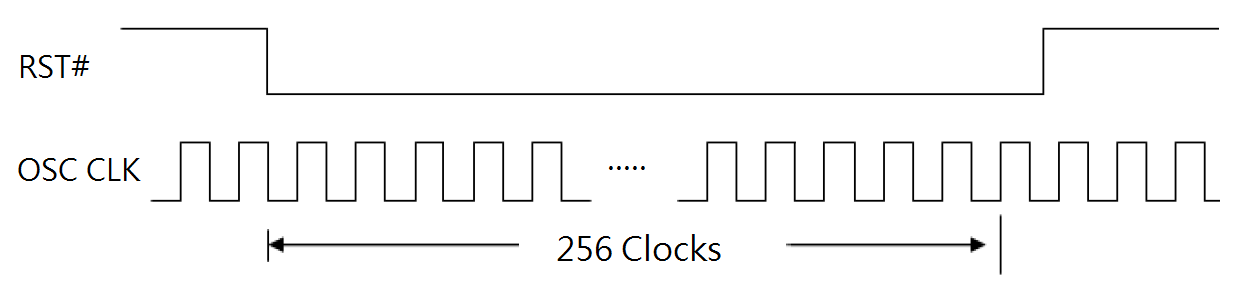
# Reset

## Power-on Reset

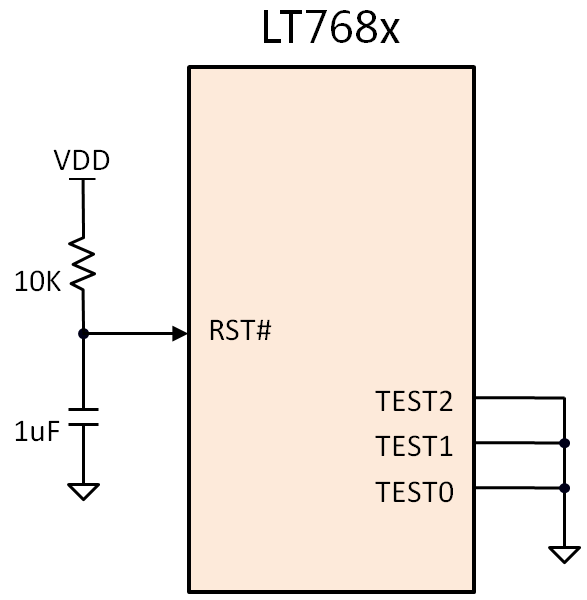
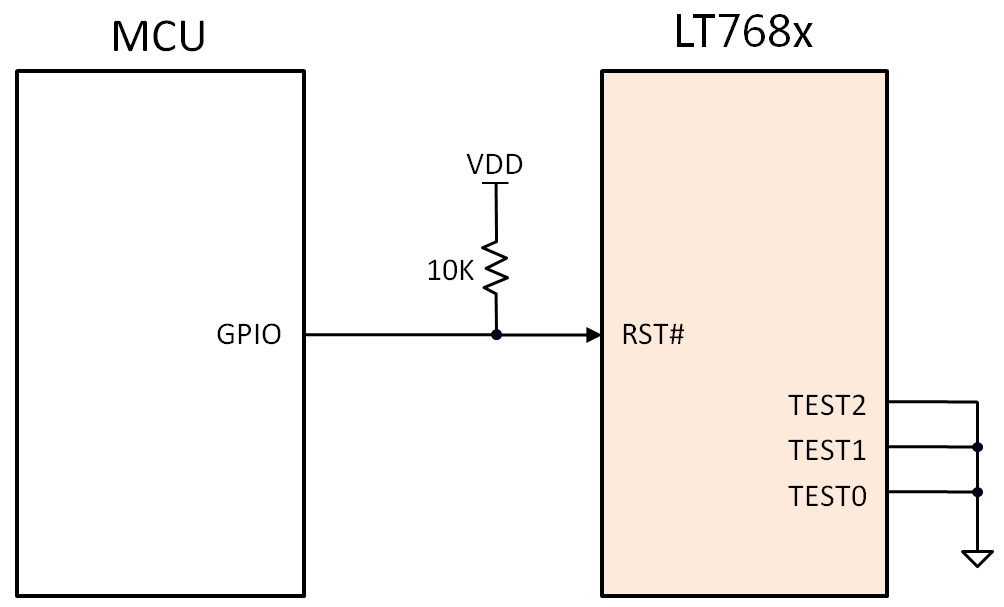
LT768x has an embedded Power-On-Reset circuit. POR can issue an active low signal to synchronize the whole system through RST# pin. When system power (3.3V) on, internal reset will be active for at least 256 OSC clocks until the internal power is stable.

## External Reset

External reset signal RST# allows LT768x to synchronize with external systems. The external reset signal must be stabilized for at least 256 crystal (OSC) clocks to be approved as shown in Figure 3-1. The MCU should check the BIT1(working mode status indication bit) of the state register STSR before setting up LT768x to ensure that LT768x is currently in "Normal Running State". External reset can be done through power-on reset or hardware reset issued by MCU, as shown in Figure 3-2, Figure 3-3.



**Figure 3-1: External Reset Signal - RST#**

** **

**Figure 3-2: External Reset (1) Figure 3-3: External Reset (2)**

## Software Reset

If the Host writes 1 to registers REG[00h] bit0, LT768x will be reset by software. The software reset will only reset the internal state machine of LT768x, and the other registers values will not be affected or cleared. After the software reset is complete, the REG[00h] bit0 will automatically be cleared to 0.

To perform a software reset, simply call the following function:

SW\_Reset(void)

## Test Signals

TEST[2:0] are the test signals of LT768x, which is provided to LT768x for testing purposes. These pins should be connected to ground (GND) in normal use. As shown in above Figure 3-2 and 3-3.

If the system is off, and users want to update image datat to SPI Flash that connects to LT768x, then TEST[2] should be pulled low, and TEST[1] should be pulled high so that LT768x can enter TEST mode and disconnect external SPI Flash. This action will allow the data to be programmed to SPI Flash without being affected by LT768x. Please refer to Sections 18.4, 19.4, 21.3, and 22.3 for more details.

# Geometric Drawing

## Drawing Line

### 1.pngDrawing a Thin Line

Line

(

[Word] X1, // X1-Axis

[Word] Y1, // Y1-Axis

[Word] X2, // X2-Axis

[Word] Y2, // Y2-Axis

**Figure 4-1: Drawing a Thin Line**

[Long] LineColor // Line Color

)

**Example:** Drawing a red Thin Line from (100, 100) to (200, 200)

Line(100, 100, 200, 200, TFT\_TFT\_RED)

### 2.pngDrawing a Thick Line

Line

(

[Word] X1, // X1-Axis

[Word] Y1, // Y1-Axis

[Word] X2, // X2-Axis

[Word] Y2, // Y2-Axis

**Figure 4-2: Drawing a Thick Line**

[Word] Width, // Line Width

[Long] LineColor // Line Color

)

**Example:** Drawing a red Thick Line from (120, 140) to (220, 260), and set the line width = 10.

Line(120, 140, 220, 260, 10, TFT\_TFT\_RED)

## Drawing Circle

### 3.pngDrawing a Hollow Circle

Circle

(

[Word] XCenter, // Center X-Axis

[Word] YCenter, // Center Y-Axis

[Word] R, // Radius

[Long] CircleColor // Color

**Figure 4-3: Drawing a Hollow Circle**

)

**Example:** Drawing a red Circle with a radius of 100, and the center is at (200, 200)

Circle(200, 200, 100, TFT\_TFT\_RED)

### 4.pngDrawing a Solid Circle

FilledCircle

(

[Word] XCenter, // Center X-Axis

[Word] YCenter, // Center Y-Axis

[Word] R, // Radius

[Long] ForegroundColor // Color

**Figure 4-4: Drawing a Solid Circle**

)

**Example:** Drawing a red Solid Circle with a radius of 100, and the center is at (200, 200)

FilledCircle (200, 200, 100, TFT\_RED)

### 5.pngDrawing a Solid Circle with Frame

FramedFilledCircle (

[Word] XCenter, // Center X-Axis

[Word] YCenter, // Center Y-Axis

[Word] R, // Radius

[Word] Width, // Frame Width

[Long] CircleColor, // Frame Color

**Figure 4-5: Drawing a Solid Circle with Frame**

[Long] ForegroundColor, // Foreground Color

)

**Example:** Drawing a white solid circle with a red frame, the frame is 10, the radius is 100, and the center is at (200, 200)

FramedFilledCircle (200, 200, 100, 10, TFT\_RED, TFT\_WHITE)

**Note:** This function is completed by drawing two Solid Circle. The frame color is determined by the first Solid Circle, and foreground color is determined by the second Solid Circle.

## 6.pngDrawing Ellipse

### Drawing a Hollow Ellipse

Ellipse

(

[Word] XCenter, // Center X-Axis

[Word] YCenter, // Center Y-Axis

[Word] X\_R, // X-Axis Radius

[Word] Y\_R, // Y-Axis Radius

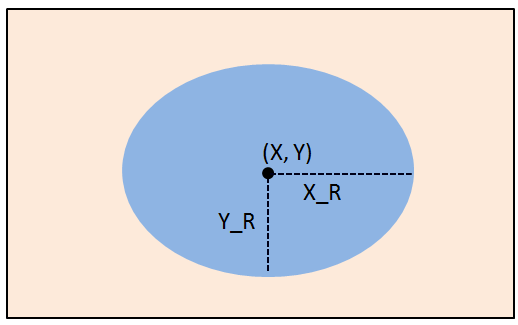
**Figure 4-6: Drawing a Hollow Ellipse**

[Long] EllipseColor // Color

)

**Example:** Drawing a red Ellipse with one radius = 80 and another = 50, and the center is at (100, 100)

Ellipse(100, 100, 80, 50, TFT\_RED)



### Drawing a Solid Ellipse

FilledEllipse

(

[Word] XCenter, // Center X-Axis

[Word] YCenter, // Center Y-Axis

[Word] X\_R, // X-Axis Radius

[Word] Y\_R, // Y-Axis Radius

**Figure 4-7: Drawing a Solid Ellipse**

[Long] ForegroundColor // Foreground Color

)

**Example:** Drawing a red Solid Ellipse with one radius = 80 and another = 50, and the center is at (100, 100)

FilledEllipse (100, 100, 80, 50, TFT\_TFT\_RED)

### 8.png Drawing a Solid Ellipse with Frame

FramedFilledEllipse

(

[Word] XCenter, // Center X-Axis

[Word] YCenter, // Center Y-Axis

[Word] X\_R, // X-Axis Radius

[Word] Y\_R, // Y-Axis Radius

**Figure 4-8: Drawing a Solid Ellipse** **with Frame**

[Word] Width, // Frame Width

[Long] EllipseColor, // Frame Color

[Long] ForegroundColor // Foreground Color

)

**Example:** Drawing a white Solid Ellipse with a red frame, the frame is 10, the radiuses are 80 and 50, and the center is at (100, 100)

FramedFilledEllipse (100, 100, 80, 50, 10, TFT\_RED, TFT\_WHITE )

**Note:** This function is completed by drawing two Solid Ellipses. The frame color is determined by the first Solid Ellipse, and foreground color is determined by the second Solid Ellipse.

## Drawing Rectangle

### 9.pngDrawing a Hollow Rectangle

Box

(

[Word] X1, // Corner X1-Axis

[Word] Y1, // Corner Y1-Axis

[Word] X2, // Corner X1-Axis

[Word] Y2, // Corner Y2-Axis

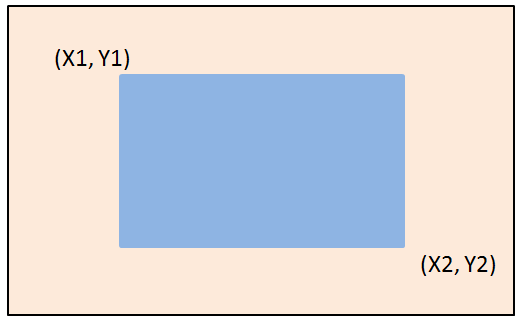
**Figure 4-9: Drawing a Hollow Rectangle**

[Long] SquareColor // Color

)

**Example:** Drawing a red Rectangle with one corner = (50, 60) and another = (200,150).

Box(50, 60, 200, 150, TFT\_RED)



### Drawing a Solid Rectangle

FilledBox

(

[Word] X1, // Corner X1-Axis

[Word] Y1, // Corner Y1-Axis

[Word] X2, // Corner X2-Axis

[Word] Y2, // Corner Y2-Axis

**Figure 4-10: Drawing a Solid Rectangle**

[Long] ForegroundColor // Foreground Color

)

**Example:** Drawing a red Solid Rectangle with one corner = (50, 60) and another = (200,150).

FilledBox (50, 60, 200, 150, TFT\_RED)

### 11.pngDrawing a Solid Rectangle with Frame

FramedFilledBox

(

[Word] X1, // Corner X1-Axis

[Word] Y1, // Corner Y1-Axis

[Word] X2, // Corner X2-Axis

[Word] Y2, // Corner Y2-Axis

**Figure 4-11: Drawing a Solid Rectangle with Frame**

[Word] Width, // Frame Width

[Long] SquareColor, // Frame Color

[Long] ForegroundColor // Foreground Color

)

**Example:** Drawing a white solid Rectanglewith a red frame. The frame width is 10, and the two corners are (50, 60) and (200, 150).

FramedFilledBox (50, 60, 200, 150, 10, TFT\_RED, TFT\_WHITE)

**Note:** This function is completed by drawing two Solid Rectangles. The frame color is determined by the first Solid Rectangle, and foreground color is determined by the second Solid Rectangle.

## 12.pngDrawing Rounded-Rectangle

### Drawing a Hollow Rounded-Rectangle

RoundRect

(

[Word] X1, // Corner X1-Axis

[Word] Y1, // Corner Y1-Axis

[Word] X2, // Corner X2-Axis

[Word] Y2, // Corner Y2-Axis

**Figure 4-12: Drawing a Hollow Rounded-Rectangle**

[Word] X\_R, // X-Axis Radius

[Word] Y\_R, // Y-Axis Radius

[Long] CircleSquareColor // Color

)

**Example:** Drawing a red Rounded-Rectangle with one corner radius = 30, and another = 20, , and the corners are at (50, 60) and (200, 150).

RoundRect (50, 60, 200, 150, 30, 20, TFT\_RED)

### 13.pngDrawing a Solid Rounded-Rectangle

FilledRoundRect

(

[Word] X1, // Corner X1-Axis

[Word] Y1, // Corner Y1-Axis

[Word] X2, // Corner X2-Axis

[Word] Y2, // Corner Y2-Axis

**Figure 4-13: Drawing a Solid Rounded-Rectangle**

[Word] X\_R, // X-Axis Radius

[Word] Y\_R, // Y-Axis Radius

[Long] ForegroundColor, // Foreground Color

)

**Example:** Drawing a red Solid Rounded-Rectangle with one corner radius = 30, and another = 20, and the corners are at (50, 60) and (200, 150).

FilledRoundRect (50, 60, 200, 150, 30, 20, TFT\_RED)

### 14.png Drawing a Rounded-Rectangle with Frame

FramedFilledRoundRect

(

[Word] X1, // Corner X1-Axis

[Word] Y1, // Corner Y1-Axis

[Word] X2, // Corner X2-Axis

[Word] Y2, // Corner Y2-Axis

**Figure 4-14: Drawing a Solid Rounded-Rectangle with Frame**

[Word] X\_R, // X-Axis Radius

[Word] Y\_R, // Y-Axis Radius

[Word] Width, // Frame Width

[Long] CircleSquareColor, // Frame Color

[Long] ForegroundColor // Foreground Color

)

**Example:** Drawing a white Rounded-Rectanglewith a red frame, the radiuses are 30 and 20, the frame width is10, and the corners are at (50, 60) and (200, 150).

FramedFilledRoundRect (50, 60, 200, 150, 30, 20, 10, TFT\_RED, TFT\_WHITE)

**Note:** This function is completed by drawing two Rounded-Rectangles. The frame color is determined by the first solid Rounded-Rectangle, and foreground color is determined by the second Rounded-Rectangle.

## 15.pngDrawing Triangle

### Drawing a Hollow Triangle

Triangle

(

[Word] X1, // Corner X1-Axis

[Word] Y1, // Corner Y1-Axis

[Word] X2, // Corner X2-Axis

[Word] Y2, // Corner Y2-Axis

**Figure 4-15: Drawing a Hollow Triangle**

[Word] X3, // Corner X3-Axis

[Word] Y3, // Corner Y3-Axis

[Long] TriangleColor // Color

)

**Example:** Drawing a red Hollow Triangle with corners = (100,100), (50, 200), and (150, 150) respectively.

Triangle(100, 100, 50, 200, 150, 150, TFT\_RED)

### 16.pngDrawing a Solid Triangle

FilledTriangle

(

[Word] X1, // Corner X1-Axis

[Word] Y1, // Corner Y1-Axis

[Word] X2, // Corner X2-Axis

[Word] Y2, // Corner Y2-Axis

**Figure 4-16: Drawing a Solid Rectangle**

[Word] X3, // Corner X3-Axis

[Word] Y3, // Corner Y3-Axis

[Long] ForegroundColor // Foreground Color

)

**Example:** Drawing a red Solid Triangle with corners = (100,100), (50, 200), and (150, 150) respectively.

FilledTriangle (100, 100, 50, 200, 150, 150, TFT\_RED)

### 17.pngDrawing a Solid Rectangle with Frame

FramedFilledTriangle

(

[Word] X1, // Corner X1-Axis

[Word] Y1, // Corner X1-Axis

[Word] X2, // Corner X1-Axis

[Word] Y2, // Corner X1-Axis

[Word] X3, // Corner X1-Axis

**Figure 4-17: Drawing a Solid Rectangle with Frame**

[Word] Y3, // Corner X1-Axis

[Long] TriangleColor, // Frame Color

[Long] ForegroundColor // Foreground Color

)

**Example:** Drawing a white Solid Triangle with a red Frame, and corners = (100,100), (50, 200), and (150, 150) respectively.

FramedFilledTriangle (100, 100, 50, 200, 150, 150, TFT\_RED, TFT\_WHITE)

**Note:** This function is completed by drawing a Solid and a Hollow Triangle. The foreground color is determined by the Solid Rectangle, and frame color is determined by the Hollow Rectangle.

## 18.pngDrawing Curve

### Drawing an Upper-Left Curve

LeftUpCurve

(

[Word] XCenter, // Center X-Axis

[Word] YCenter, // Center Y-Axis

[Word] X\_R, // X-Axis Radius

[Word] Y\_R, // Y-Axis Radius

**Figure 4-18: Drawing an Upper-Left Curve**

[Long] CurveColor // Color

)

**Example:** Drawing a red Upper-Left Curve with one radius = 100 and another = 70, and the center is at (100, 100)

LeftUpCurve(100, 100, 100, 70, TFT\_RED)

### Drawing a Lower-Left Curve19.png

LeftDownCurve

(

[Word] XCenter, // Center X-Axis

[Word] YCenter, // Center Y-Axis

[Word] X\_R, // X-Axis Radius

[Word] Y\_R, // Y-Axis Radius

**Figure 4-19: Drawing a Lower-Left Curve**

[Long] CurveColor // Color

)

**Example:** Drawing a red Lower-Left Curve with one radius = 100 and another = 70, and the center is at (100, 100)

LeftDownCurve(100, 100, 100, 70, TFT\_RED)

### 20.pngDrawing an Upper-Right Curve

RightUpCurve

(

[Word] XCenter, // Center X-Axis

[Word] YCenter, // Center Y-Axis

[Word] X\_R, // X-Axis Radius

[Word] Y\_R, // Y-Axis Radius

**Figure 4-20: Drawing an Upper-Right Curve**

[Long] CurveColor // Color

)

**Example:** Drawing a red Upper-Right Curve with one radius = 100 and another = 70, and the center is at (100, 100)

RightUpCurve(100, 100, 100, 70, TFT\_RED)

### 21.pngDrawing a Lower-Right Curve

RightDownCurve

(

[Word] XCenter, // Center X-Axis

[Word] YCenter, // Center Y-Axis

[Word] X\_R, // X-Axis Radius

[Word] Y\_R, // Y-Axis Radius

**Figure 4-21: Drawing a Lower-Right Curve**

[Long] CurveColor // Color

)

**Example:** Drawing a red Lower-Right Curve with one radius = 100 and another = 70, and the center is at (100, 100)

RightDownCurve(100, 100, 100, 70, TFT\_RED)

## 22.pngDrawing 1/4 Ellipse

### Drawing an Upper-Left 1/4 Ellipse

FilledLeftUpCurve

(

[Word] XCenter, // Center X-Axis

[Word] YCenter, // Center Y-Axis

[Word] X\_R, // X-Axis Radius

[Word] Y\_R, // Y-Axis Radius

**Figure 4-22: Drawing an Upper-Left 1/4 Ellipse**

[Long] ForegroundColor // Color

)

**Example:** Drawing a red Upper-Left 1/4 Ellipse with one radius = 100 and another = 70, and the center is at (100, 100)

FilledLeftUpCurve(100, 100, 100, 70, TFT\_RED)

### 23.png Drawing an Lower-Left 1/4 Ellipse

FilledLeftDownCurve

(

[Word] XCenter, // Center X-Axis

[Word] YCenter, // Center Y-Axis

[Word] X\_R, // X-Axis Radius

[Word] Y\_R, // Y-Axis Radius

**Figure 4-23: Drawing a Lower-Left 1/4 Ellipse**

[Long] ForegroundColor // Color

)

**Example:** Drawing a red Lower-Left 1/4 Ellipse with one radius = 100 and another = 70, and the center is at (100, 100)

FilledLeftDownCurve(100, 100, 100, 70, TFT\_RED)

### 24.png Drawing an Upper-Right 1/4 Ellipse

FilledRightUpCurve

(

[Word] XCenter, // Center X-Axis

[Word] YCenter, // Center Y-Axis

[Word] X\_R, // X-Axis Radius

[Word] Y\_R, // Y-Axis Radius

**Figure 4-24: Drawing an Upper-Right 1/4 Ellipse**

[Long] ForegroundColor // Color

)

**Example:** Drawing a red Upper-Right 1/4 Ellipse with one radius = 100 and another = 70, and the center is at (100, 100)

FilledRightUpCurve(100, 100, 100, 70, TFT\_RED)

### 26.png Drawing an Lower-Right 1/4 Ellipse

FilledRightDownCurve

(

[Word] XCenter, // Center X-Axis

[Word] YCenter, // Center Y-Axis

[Word] X\_R, // X-Axis Radius

[Word] Y\_R, // Y-Axis Radius

**Figure 4-25: Drawing a Lower-Right 1/4 Ellipse**

[Long] ForegroundColor // Color

)

**Example:** Drawing a red Lower-Right 1/4 Ellipse with one radius = 100 and another = 70, and the center is at (100, 100)

FilledRightDownCurve(100, 100, 100, 70, TFT\_RED)

## Drawing Quadrilateral

### 27.pngDrawing a Hollow Quadrilateral

Quadrilateral

(

[Word] X1, // Corner X1-Axis

[Word] Y1, // Corner Y1-Axis

[Word] X2, // Corner X2-Axis

[Word] Y2, // Corner Y2-Axis

**Figure 4-26: Drawing a Hollow Quadrilateral**

[Word] X3, // Corner X3-Axis

[Word] Y3, // Corner Y3-Axis

[Word] X4, // Corner X4-Axis

[Word] Y4, // Corner Y4-Axis

[Long] ForegroundColor // Frame Color

)

**Example:** Drawing a red Hollow Quadrilateral with the corners = (50, 50), (200, 80), (150, 130) and (60, 100) respectively.

Quadrilateral(50, 50, 200, 80, 150, 130, 60, 100, TFT\_RED)

**Note:** The quadrilateral can be arbitrarily set coordinates of four corners. But the rectangle only need sets the coordinates of two corners.

### 28.png Drawing a Solid Quadrilateral

FilledQuadrilateral

(

[Word] X1, // Corner X1-Axis

[Word] Y1, // Corner Y1-Axis

[Word] X2, // Corner X2-Axis

[Word] Y2, // Corner Y2-Axis

**Figure 4-27: Drawing a Solid Quadrilateral**

[Word] X3, // Corner X3-Axis

[Word] Y3, // Corner Y3-Axis

[Word] X4, // Corner X4-Axis

[Word] Y4, // Corner Y4-Axis

[Long] ForegroundColor // Foreground Color

)

**Example:** Drawing a red Solid Quadrilateral with the corners = (50, 50), (200, 80), (150, 130) and (60, 100) respectively.

FilledQuadrilateral (50, 50, 200, 80, 150, 130, 60, 100, TFT\_RED)

## Drawing Pentagonal

### 29.pngDrawing a Hollow Pentagonal

Pentagon

(

[Word] X1, // Corner X1-Axis

[Word] Y1, // Corner Y1-Axis

[Word] X2, // Corner X2-Axis

[Word] Y2, // Corner Y2-Axis

**Figure 4-28: Drawing a Hollow Pentagonal**

[Word] X3, // Corner X3-Axis

[Word] Y3, // Corner Y3-Axis

[Word] X4, // Corner X4-Axis

[Word] Y4, // Corner Y4-Axis

[Word] X5, // Corner X5-Axis

[Word] Y5, // Corner Y5-Axis

[Long] ForegroundColor // Frame Color

)

**Example:** Drawing a red Hollow Pentagonal with the coners = (50, 100), (120, 130), (150, 160), (100, 180) and (80, 140) respectively.

Pentagon(50, 100, 120, 130, 150, 160, 100, 180, 80, 140, TFT\_RED)

### 30.pngDrawing a Solid Pentagonal

FilledPentagon

(

[Word] X1, // Corner X1-Axis

[Word] Y1, // Corner Y1-Axis

[Word] X2, // Corner X2-Axis

[Word] Y2, // Corner Y2-Axis

**Figure 9-29: Drawing a Solid Pentagonal**

[Word] X3, // Corner X3-Axis

[Word] Y3, // Corner Y3-Axis

[Word] X4, // Corner X4-Axis

[Word] Y4, // Corner Y4-Axis

[Word] X5, // Corner X5-Axis

[Word] Y5, // Corner Y5-Axis

[Long] ForegroundColor // Foreground Color

)

**Example:** Drawing a red Solid Pentagonal with corners = (50, 100), (120, 130), (150, 160), (100, 180) and (80, 140) respectively.

FilledPentagon (50, 100, 120, 130, 150, 160, 100, 180, 80, 140, TFT\_RED)

## Drawing Cylinder

Cylinder

(

[Word] XCenter, // Center X-Axis

[Word] YCenter, // Center X-Axis

[Word] X\_R, // X Radius

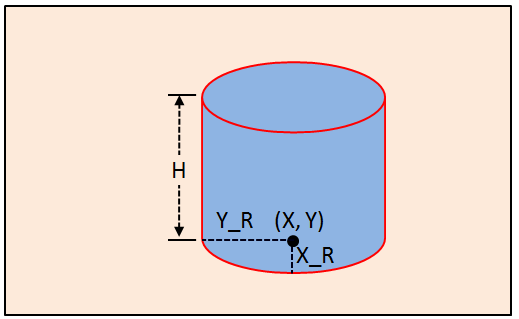
[Word] Y\_R, // Y Radius

[Word] H, // Cylinder Height

[Long] CylinderColor, // Frame Color

[Long] ForegroundColor // Foreground Color

)



**Figure 4-30: Drawing a Cylinder**

**Example:** Drawing a blue Cylinderwith a red frame, the radiuses are 100 and 50, the height is 150, and the center is at (200, 300).

Cylinder (200, 300, 50, 100, 150, TFT\_RED, TFT\_BLUE)

**Note:** The implementation of drawing a Cylinder is integrated by some of the drawing functions mentioned in this Chapter, and the parameters of the Ellipse in this function are based on the bottom Ellipse.

## Drawing Cube

FilledCube

(

[Word] X1, // X1-Axis

[Word] Y1, // Y1-Axis

[Word] X2, // X2-Axis

[Word] Y2, // Y2-Axis

[Word] X3, // X3-Axis

[Word] Y3, // Y3-Axis

[Word] X4, // X4-Axis

[Word] Y4, // Y4-Axis

[Word] X5, // X5-Axis

[Word] Y5, // Y5-Axis

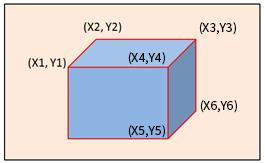
[Word] X6, // X6-Axis

[Word] Y6, // Y6-Axis

[Long] QuadrangularColor, // Frame Color

[Long] ForegroundColor // Foreground Color

)

****

**Figure 4-31: Drawing a Cube**

**Example:** Drawing a blue Cubewith a red frame, the 6 points are (300, 300), (350, 250), (400, 250), (350, 300), (350, 350), and (400, 300)

FilledCube (300, 300, 350, 250, 400, 250, 350, 300, 350, 350, 400, 300, TFT\_RED, TFT\_BLUE)

**Note:** The implementation of drawing a Cube is integrated by some of the drawing functions mentioned in this Chapter.

## Drawing Table

GLCDMakeTable (

[Word] X1, // Start Corner X-Axis

[Word] Y1, // Start Corner Y-Axis

[Word] W, // Cell Width (W)

[Word] H, // Cell Height (H)

[Word] Cols, // Column Number (total) (CN)

[Word] Rows, // Row Number (total) (RN)

[Word] width1, // Inner Frame Width

[Word] width2, // Outer Frame Width

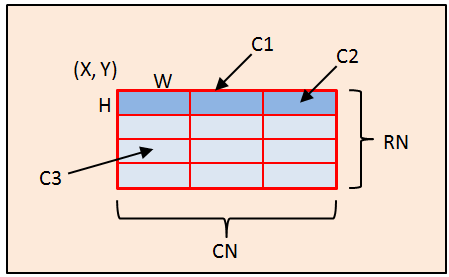
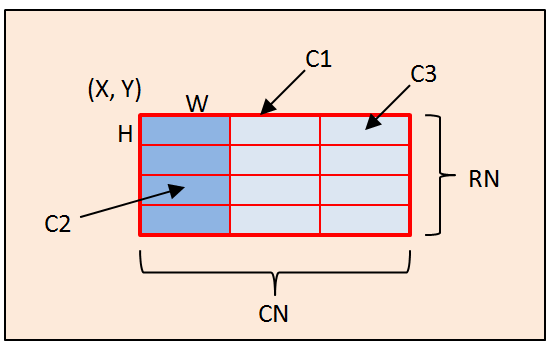
[Byte] mode, // Bit0: Item Table is Vertical, Bit1: Item Table is Horizontal

[Long] TableColor, // Frame Color (C1)

[Long] ItemColor, // Item Table Color (C2)

[Long] ForegroundColor // Content Table Color (C3)

)

**Figure 4-32: Drawing a Horizontal Accent Figure 4-33: Drawing a Vertical Accent**

**Example:** Example: Drawing a horizontal table window at (5, 20) position. The cell size is 31\*20, table number of rows is 22, table number of columns is 25, the frame color is red, foreground color of Item table is green, foreground color of content table is white, inner frame width is 1, outer frame width is 3.

#define TOPL 5

#define TOPR 20

#define CELLW 31

#define CELLH 20

#define NUMOFCOLS 25

#define NUMOFROWS 22

#define INNERFRAME 1

#define OUTERFRAME 3

// These constants control the accent of the table. These are additive.

#define TOPROWACCENT 2

#define LEFTCOLUMNACCENT 1

#define NOACCENT 0

GLCDMakeTable ( TOPL, TOPR, CELLW, CELLH, NUMOFCOLS, NUMOFROWS, INNERFRAME, OUTERFRAME, TOPROWACCENT + LEFTCOLUMNACCENT, TFT\_RED, TFT\_GREEN, TFT\_WHITE ).

**Note:** The implementation of drawing a Table is integrated by some of the drawing functions mentioned in this Chapter.

# Function Library List

Table 24-1: Function Library List

| **No.** | **Function Name** | **Function Description** |
| --- | --- | --- |
| 1 | LT7686\_Software\_Reset () | Software Reset |
| 2 | InitGLCD\_LT7686() | Setup Clock & PLL |
| 63 | LT7686\_Display\_ColorBar( TRUE ) | Color Bar Display Enable |
| 64 | LT7686\_Display\_ColorBar( FALSE ) | Color Bar Display Disable |
| 65 | Line() | Drawing a Thin Line |
| 66 | Line() | Drawing a Thick Line |
| 67 | Circle() | Drawing a Hollow Circle |
| 68 | FilledCircle() | Drawing a Solid Circle |
| 69 | FramedFilledCircle() | Drawing a Solid Circle with Frame |
| 70 | Ellipse() | Drawing a Hollow Ellipse |
| 71 | FilledEllipse() | Drawing a Solid Ellipse |
| 72 | FramedFilledEllipse() | Drawing a Solid Ellipse with Frame |
| 73 | Square() | Drawing a Hollow Rectangle |
| 74 | FilledSquare() | Drawing a Solid Rectangle |
| 75 | FramedFilledSquare() | Drawing a Solid Rectangle with Frame |
| 76 | CircleSquare() | Drawing a Hollow Rounded-Rectangle |
| 77 | FilledCircleSquare() | Drawing a Solid Rounded-Rectangle |
| 78 | FramedFilledCircleSquare() | Drawing a Rounded-Rectangle with Frame |
| 79 | Triangle() | Drawing a Hollow Triangle |
| 80 | FilledTriangle() | Drawing a Solid Triangle |
| 81 | FramedFilledTriangle\_Frame() | Drawing a Solid Rectangle with Frame |
| 82 | LeftUpCurve() | Drawing an Upper-Left Curve |
| 83 | LeftDownCurve() | Drawing a Lower-Left Curve |
| 84 | RightUpCurve() | Drawing an Upper-Right Curve |
| 85 | RightDownCurve() | Drawing a Lower-Right Curve |
| 86 | FilledLeftUpCurve() | Drawing an Upper-Left 1/4 Ellipse |
| 87 | FilledLeftDownCurve() | Drawing an Lower-Left 1/4 Ellipse |
| 88 | FilledRightUpCurve() | Drawing an Upper-Right 1/4 Ellipse |
| 89 | FilledRightDownCurve() | Drawing an Lower-Right 1/4 Ellipse |
| 90 | Quadrilateral() | Drawing a Hollow Quadrilateral |
| 91 | FilledQuadrilateral() | Drawing a Solid Quadrilateral |
| 92 | Pentagon() | Drawing a Hollow Pentagonal |
| 93 | FilledPentagon() | Drawing a Solid Pentagonal |
| 94 | Cylinder() | Drawing Cylinder |
| 95 | Quadrangular() | Drawing Cube |
| 96 | GCLDMakeTable() | Drawing Table |
| 106 | Select\_Internal\_Font\_Init() | Internal Font Initialize |
| 107 | Print\_Internal\_Font\_String() | Setup Internal Font |
| 108 | Select\_Outside\_Font\_Init() | External Font Initialize |
| 109 | Print\_Outside\_Font\_String() | Setup External Font |
| 110 | Print\_Outside\_Font\_String\_BIG5() | 显示外建字库(BIG5) |
| 111 | Print\_Outside\_Font\_GB2312\_48\_72() | 显示大的外建字库 |
| 112 | Print\_Outside\_Font\_BIG5\_48\_72() | 显示大的外建字库(BIG5) |
| 113 | Font\_Line\_Distance() | 使用外建字库 – 文字行距 |
| 114 | Text\_cursor\_Init() | Text Cursor Initialize |
| 115 | Enable\_Text\_Cursor() | Enable Text Cursor |
| 116 | Disable\_Text\_Cursor() | Disable Text Cursor |
| 117 | Graphic\_cursor\_Init() | Graphic Cursor Initialize |
| 118 | Set\_Graphic\_cursor\_Pos() | Change Graphic Cursor Position |
| 119 | Enable\_Graphic\_Cursor() | Enable Graphic Cursor |
| 120 | Disable\_Graphic\_Cursor() | Disable Graphic Cursor |
| 121 | PWM0\_Init() | PWM0 Initialize |
| 122 | PWM1\_Init() | PWM1 Initialize |
| 123 | PWM0\_Duty([Word] Compare\_Buffer) | Setup PWM0 Duty Rate |
| 124 | PWM1\_Duty([Word] Compare\_Buffer) | Setup PWM1 Duty Rate |
| 129 | Standby() | Enter Standby Mode |
| 130 | Suspend() | Enter Suspend Mode |
| 131 | SleepMode() | Enter Sleep Mode |
| 132 | Wkup\_Standby() | Wakeup from Standby Mode |
| 133 | Wkup\_Suspend() | Wakeup from Suspend Mode |
| 134 | Wkup\_Sleep() | Wakeup from Sleep Mode |

**index.pngVersion History**

Table A-1: Version History

|  |  |  |
| --- | --- | --- |
| **Version** | **Date** | **Revision Description** |
|  |  |  |

**index.pngCopyright Notice**