



深圳市拓普微科技开发有限公司
Shenzhen TOPWAY Technology Co., Ltd.

TOPWAY

Smart TFT LCD Module

RGTools Handbook

FAST

Shorten
Development Time

VALUE

Simplifies Hardware
Saves BOM Costs

RELIABLE

High Endurance of noise
Industrial Level

in minutes
LOAD & PLAY



Content

1	Quick Start.....	5
1.1	How to Use a Smart TFT LCD Module	5
1.2	Smart TFT LCD Module Connection Examples	5
1.3	Quick Start Example	5
2	Basic Descriptions	6
2.1	Smart TFT LCD Module Highlight.....	6
2.2	RGTools Highlight.....	6
2.3	Functional Block.....	6
3	Smart TFT LCD Module Features and Functions	7
3.1	Basic Concept.....	7
3.1.1	PAGE	7
3.1.2	Page Elements.....	7
3.1.3	Image Recourses.....	7
3.1.4	VP Variables	7
3.1.5	Call Functions	7
3.2	Display and Functional Relationships.....	7
3.3	Page Elements.....	8
3.3.1	Action Type Elements	8
3.3.2	Character (Alpha Numeric) Type Elements	8
3.3.3	Image Type Elements	8
3.3.4	Graphics Type Elements Details	9
3.4	Page, IMAGE Resource and VP Variables.....	10
3.4.1	Page and IMAGE Resource	10
3.4.2	VP Variable	10
4	RGTools	11
4.1	RGTools Layout	11
4.2	General Operations.....	11
4.3	RGTools Menu	12
4.3.1	Start.....	12
4.3.2	New Project (Menu - File - New Project)	12
4.3.3	Open Project (Menu - File - Open Project)	12
4.3.4	Save (Menu - File - Save).....	12
4.3.5	Save As (Menu - File - Save As)	12
4.3.6	Close Project (Menu - File - Close).....	12
4.3.7	RGTools Options (Menu - Tools - Options)	13
4.3.8	Project Setting (Menu - Tools - Project Setting)	13
4.3.9	Font Setting (Menu - Tools - Font Setting)	14
4.3.10	Built Project Files (Menu - Tools - Built Project Files)	14
4.3.11	Download to Module (Menu - Tools - Download to Module)	14
4.4	Elements Configurations.....	15
4.4.1	Touch Key (TPK)	15
4.4.2	Virtual Key (VPK).....	17
4.4.3	Static String (STS).....	18
4.4.4	String Element (STR).....	19
4.4.5	Number Elements (N16, N32, N64).....	20
4.4.6	Timer Display (TMR).....	22
4.4.7	Real Time Clock (RTC).....	23
4.4.8	Static Icon (ICO)	24
4.4.9	Animation Element (ANI)	25
4.4.10	Bit Icon (IDX_BIT).....	26
4.4.11	Indexed Icon (IDX).....	27
4.4.12	Tachometer (TCM).....	28
4.4.13	Decimal Icon (I16, I32).....	30
4.4.14	Progress Bar (B16).....	32
4.4.15	Graph Element (G16).....	34
4.4.16	Bitmap Element (BP1)	35
4.4.17	QR Code Element (QRC)	36
4.4.18	Draw Pad (DPD)	36

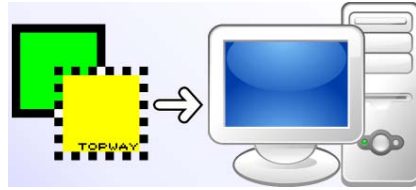
4.4.19	Predefined Touch Key (ESC)	37
4.4.20	Predefined Touch Key (DEL)	37
4.4.21	Predefined Touch Key (Left)	37
4.4.22	Predefined Touch Key (Right)	37
4.4.23	Predefined Touch Key (ESC)	37
4.4.24	Predefined Touch Key (CapLock)	37
4.4.25	Predefined Touch Key (char)	38
4.4.26	Predefined Touch Key - Set (number keyboard 1)	38
4.4.27	Predefined Touch Key - Set (number keyboard 2)	39
4.4.28	Predefined Touch Key - Set (English keyboard)	39
4.4.29	Page Properties	40
4.4.30	Page Functions	40
4.5	Elements Sub Functions Details	41
4.5.1	Draw Pad (DPD) in-memory command Functions	41
4.5.2	TPK, VPK Call Functions	42
5	Serial Communication	44
5.1.1	Hardware connection	44
5.1.2	Communication Packet Structure	44
5.1.3	Packet Acknowledgment	44
5.1.4	Color Data Value Configuration	44
5.1.5	Data / Address / Page_ID / Location Values Configuration	44
5.2	Command Summary	45
5.3	Command Details	46
5.3.1	Config / Status Functions	46
5.3.2	Display Control / Draw Functions	47
5.3.3	VP Functions	48
6	Basic Function Examples	50
6.1	Display a <u>PAGE</u> with <u>background image</u> (IMG_BKG)	50
6.2	Using <u>Touch Key</u> (TPK) to Jump to a Page	51
6.3	Show a <u>Static Icon</u> (ICO)	52
6.4	Show a <u>Static String</u> (STS)	53
6.5	Show an <u>Animation Element</u> (ANI)	54
7	VP_Variables and Keyboard Examples	55
7.1	Using <u>Internal Number Keyboard</u> to input a value to a <u>VP_N32</u>	55
7.2	Using <u>Internal English Keyboard</u> to input a string to a <u>VP_STR</u>	56
7.3	Using <u>Internal Password Keyboard</u> to input a number pin	57
7.4	Using <u>Internal Real-Time-Clock Keyboard</u> to set the <u>Real-Time-Clock(RTC)</u> time	58
7.5	Using <u>PIP Menu</u> to input a String Value	59
7.6	Using <u>PIP (Number) Keyboard</u> to input a value	60
7.7	Using <u>PIP Keyboard</u> to input a value	61
7.8	Using <u>TPK</u> to <u>operate VP</u> value	62
8	VP and COM communication Examples	63
8.1	Using <u>Serial Command</u> to <u>update a Number variable</u>	63
9	Advance element Examples	64
9.1	Show a <u>Progress Bar</u>	64
9.2	Using <u>Index Icon</u>	65
9.3	Using <u>Tachometer Element</u> (Angle Open Mode) Application Example	66
9.4	Using <u>Tachometer Element</u> (Rotation Mode) Application Example (Meter)	68
9.5	Using <u>Tachometer Element</u> (Rotation Mode) Application Example (Radar)	69
9.6	Using <u>Decimal Icon</u>	70
9.7	Using <u>Graph Element</u>	71
9.8	Show a <u>Bitmap Element</u>	72
10	Appendix	73
10.1	RGTools Shortcut Keys	73
10.2	Project Limitation	74
10.2.1	PAGES, Image-Resources and VP-Variables Limitation	74
10.2.2	Page's Elements Limitation	75

10.2.3	System Registers	76
11	Precautions of using LCD Modules	77
12	Revisions	78

1 Quick Start

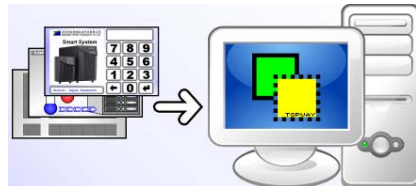
1.1 How to Use a Smart TFT LCD Module

1 Install TOPWAY RGTools



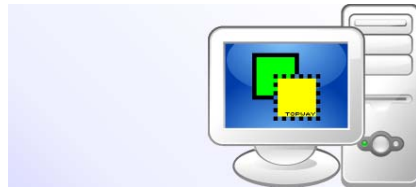
Note: RGTools support Windows XP, Windows Vista and Window 7 (Administrator Mode)

2 Import pictures design UI flow



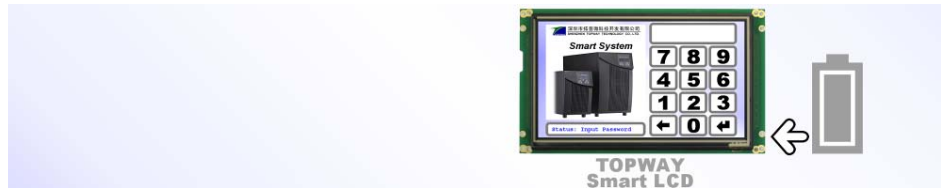
Note: RGTools support BMP(32bit), BMP(24bit), JPG, PNG, etc picture format

3 Download to Smart TFT LCD Module



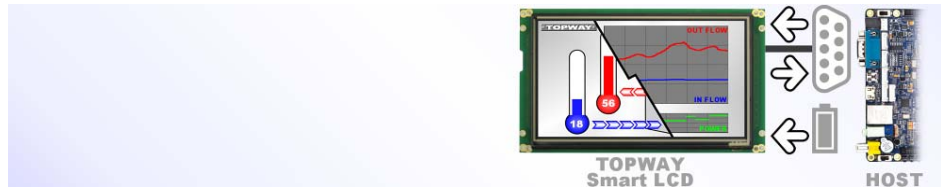
Note: Use High quality USB(A) to USB(mini) cable directly attach to the PC motherboard for better power and signaling

4 Power on & display



Note: Please refer to the user manual for connection polarity and voltage.

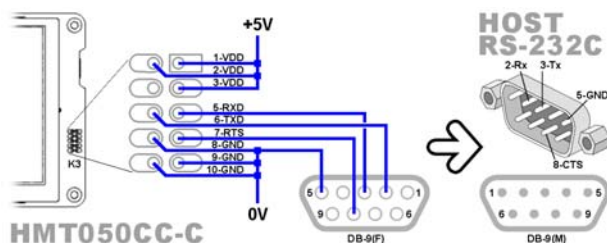
5 Connect to host Show real time data



Note: RS-232C terminal is common grounded with the supply (some models providing logic level UART interface)

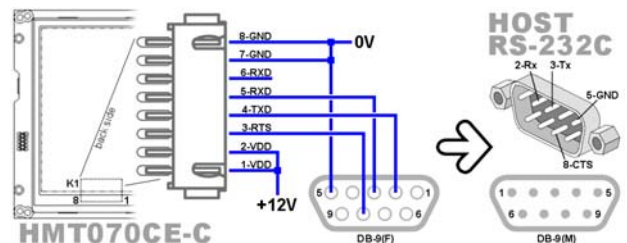
Please also refer to Product User Manual for details.

1.2 Smart TFT LCD Module Connection Examples



HMT050CC-C connection example

Please also refer to Product User Manual for details.



HMT070CE-C connection example

1.3 Quick Start Example

Please refer to "Basic Function Example" section

2 Basic Descriptions

TOPWAY **Smart TFT LCD Modules** provide a display engine with a versatile range of TFT display sizes and Touch-Panels to support a wide range of industrial and instrumentation applications.

The pre-loaded User Interface (UI) reduces the host system's workload and provides a much faster interaction with the user.

UI designs are implemented under usage of **RGTools** allowing designs with zero coding. It dramatically simplifies and speeds up the whole product design process.

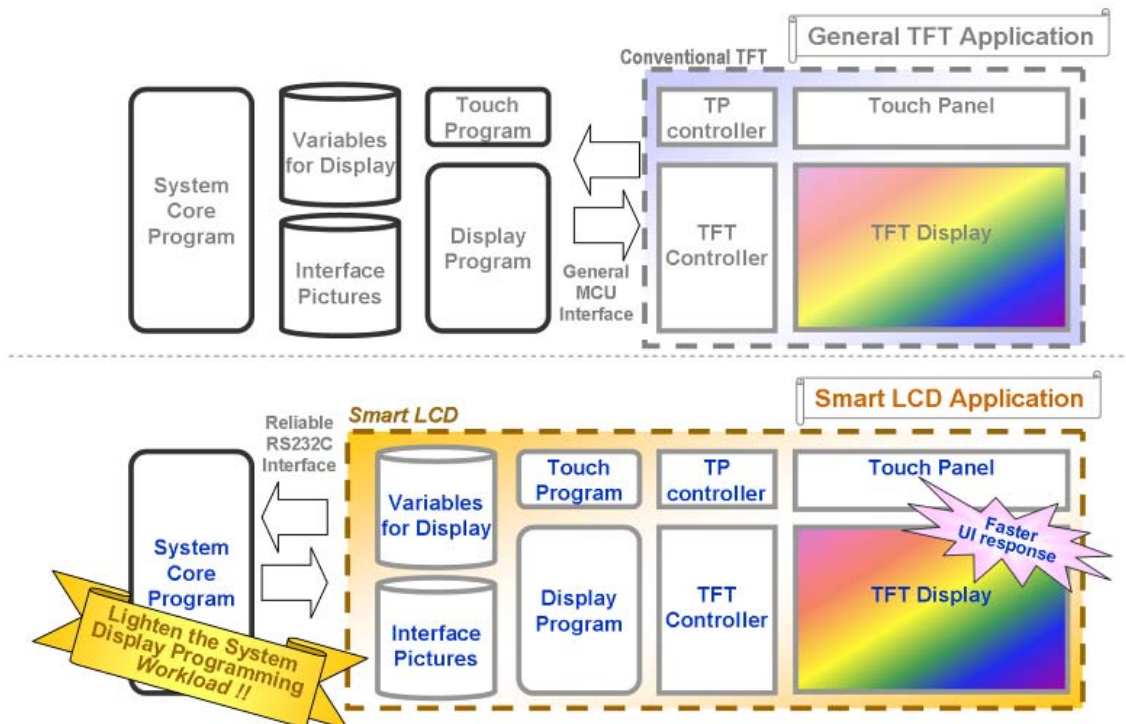
2.1 Smart TFT LCD Module Highlight

- Standard RS232-C communication interface
- Reliable packet protocol ensure reliable communication
- Host data could be accepted at any moment
- Free the host from interface response and handling
- Direct connect the USB terminal to PC for development
- USB thumb drive with OTG cable can be used for data preload in production stage (options)
- 256Mbyte Flash (vary by model) for interface pictures preload (more than 300 pictures(800x480))

2.2 RGTools Highlight

- Professional Visual design interface
- Web page like page base UI design
- Various kind of touch screen effect available
- User friendly work space
- Support various countries font ASCII and extended char
- Support font import and fine font adjustment
- Only need few minutes to show a UI sample with zero coding
- One click download

2.3 Functional Block



3 Smart TFT LCD Module Features and Functions

3.1 Basic Concept

3.1.1 PAGE

Each display interface is a PAGE.

It could contain / link with all the predefined content (e.g. background image, page elements, etc.)

3.1.2 Page Elements

18 type page elements could be use on screen

2 action type elements, 5 Character (Alpha Numeric) type elements, 6 image type elements, 5 graphics type elements

Most of them are linked with VP variable to generate dynamic display content.

3.1.3 Image Recourses

Image Recourses include background images (IMG_BKG), icon images (IMG_ICO) and animation images (IMG_ANI).

They could link by PAGE or Elements to be show on screen.

3.1.4 VP Variables

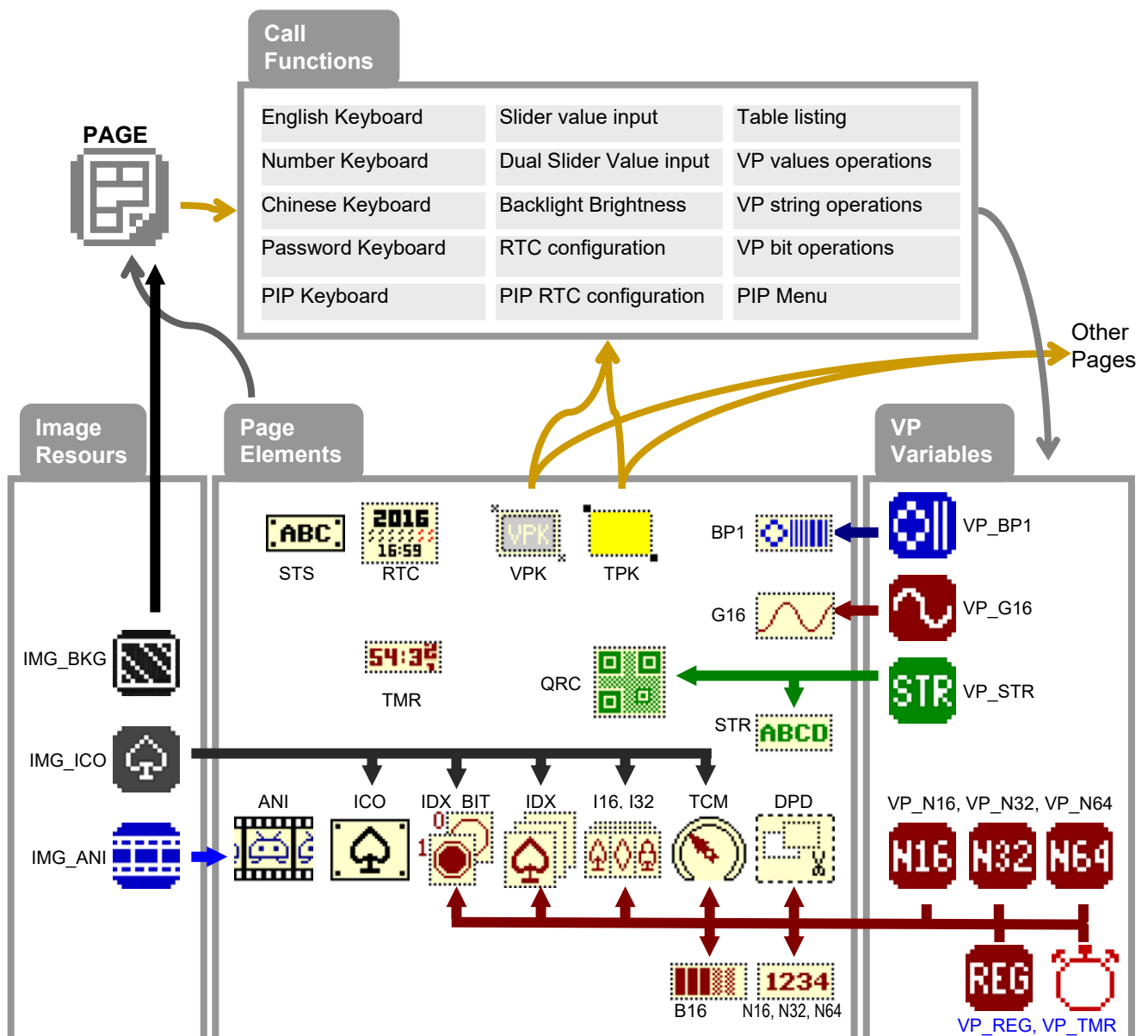
VP Variables is a RAM area to hold functional variable values.

They could be accessed by host via serial port command, value operations or on screen Keyboard etc.

3.1.5 Call Functions

Call Functions are functions (e.g. keyboard, values operations, etc) that could be triggered by a Page start, Touch Key or Virtual Touch Key.

3.2 Display and Functional Relationships



3.3 Page Elements

3.3.1 Action Type Elements

Icon	Mnemonic	Name	Descriptions
	TPK	Touch Key	Touch Key Element could place inside a page. It will be triggered by user touching on its assigned area <ul style="list-style-type: none"> - Touch Key assigned area could provide visual touch response (e.g. inverse color or show a icon, etc) - It will also report to the host machine about its ID (Page_ID / Key_ID, operated VP_address and value, etc) - Its call functions could provide various kinds of actions (e.g. keyboards, value operations, page jump, etc.)
	VPK	Virtual Key	Virtual Key is a non-display element. It could be defined in a PAGE and trigged by a condition like "content of a VP equal to constant value" <ul style="list-style-type: none"> - Virtual Key assigned area will NOT provide visual response - Its call functions could provide various kinds of actions (e.g. keyboards, value operations, page jump, etc.) - After being triggered, the monitored VP will be cleared as 0 <p>(note: only the PAGE with VPK showing on screen could be trigged)</p>

note: Elements overlapped in the PAGE might affect the display result

3.3.2 Character (Alpha Numeric) Type Elements

Icon	Mnemonic	Name	Descriptions
	STS	Static String	Static String element shows text string defined by its properties directly.
	STR	String Element	String Element is for displaying VP_STR content <ul style="list-style-type: none"> - Display the VP_STR content in PAGE with alignment and formatting - String Element will be updated correspondingly in real-time, if the VP_STR content changed.
	N16 N32 N64	Number Element	Number Element is for displaying VP_N16, VP_N32, VP_N64 content <ul style="list-style-type: none"> - Display the VP_N16, VP_N32 and VP_N64 content in PAGE with alignment and formatting - Number Element will be updated correspondingly in real-time, if the VP_N16, VP_N32 and VP_N64 content changed.
	TMR	Timer Display	Timer Display element links to a timer addressed inside VP_N32. It Display the timer value with alignment and formatting
	RTC	Real Time Clock	Real Time Clock element shows the Smart LCM internal clock with formatting.


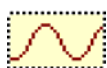



note: Elements overlapped in the PAGE might affect the display result

3.3.3 Image Type Elements

Icon	Mnemonic	Name	Descriptions
	ICO	Static Icon	Static Icon element links to an IMG_ICO and shows it on screen
	ANI	Animation Element	Animation Element links to an IMG_ANI and shows it on the screen It is animation speed could be defined.
	IDX_BIT	Bit Icon	Bit Icon element links with one bit value of VP_N16 or VP_N32 which could point to an IMG_ICON and display accordingly with its value
	IDX	Indexed Icon	Indexed Icon element links with VP_N16 or VP_N32 which could point to an IMG_ICON and display accordingly If the value in side the VP is outside the defined max/min rang, it will display nothing
	TCM	Tachometer	Tachometer element link with VP_N16 or VP_N32 which show as a meter display accordingly. If the value in side the VP is outside the defined max/min rang, it will display nothing
	I16 I32	Decimal Icon	Decimal Icon element links with VP_N16 and display as decimal formatted icons It should linked with 12 icons for full operation (where the indexed picture sequence is 0~9, dot, minus)

note: Elements overlapped in the PAGE might affect the display result





3.3.4 Graphics Type Elements Details

Icon	Mnemonic	Name	Descriptions
	B16	Progress Bar	Progress Bar element links with VP_N16 and display a bar based on its value. The bar length is a percentage of the defined max/min value with the VP_N16 value. An ICON pointer could also be link with it as an option
	G16	Graph Element	Graph Element display graphical plot of the VP_G16 content - It provide real time update of the VP_G16 value
	BP1	Bitmap Element	Bitmap Element display a mono picture content which store inside VP_BP1 - it show picture in 1bpp - It provide LUT for mapping 0 and 1 to different defined color - It provide real time update of the VP_BP1 value
	QRC	QR Code Element	QR Code Element generates a QR code with a linked VP_STR
	DPD	Draw Pad	Draw Pad element is an element that execute a drawing function that store inside VP_N16 which point by its properties VP_address. Once the command value updated, the drawing will be updated in real time.

note: Elements overlapped in the PAGE might affect the display result









3.4 Page, IMAGE Resource and VP Variables

3.4.1 Page and IMAGE Resource

Icon	Mnemonic	Name	Descriptions
	PAGE	Page	Page is the basic unit of display. It links to elements and background Image(background color) predefined in RGTools
	IMG_BKG	Background Image	Background Image is one of the Picture Resources. Picture files could be imported into RGTools as Background Image. It could also be used (automated crop) as TPK touch down effect image.
	IMG_ICO	Icon	Icon is one of the Picture Resources. Icon files could be imported into the RGTools and linked by various elements. (e.g. TPK ,IDX ,I32/16 ,B16 , ICO, etc)
	IMG_ANI	Animation	Animation is one of the Picture Resources It is a series of images file which imported into the RGTools and linked by ANI element to play.

Note: PAGEs and IMAGEs are store inside Smart TFT LCD Module Flash memory.

3.4.2 VP Variable

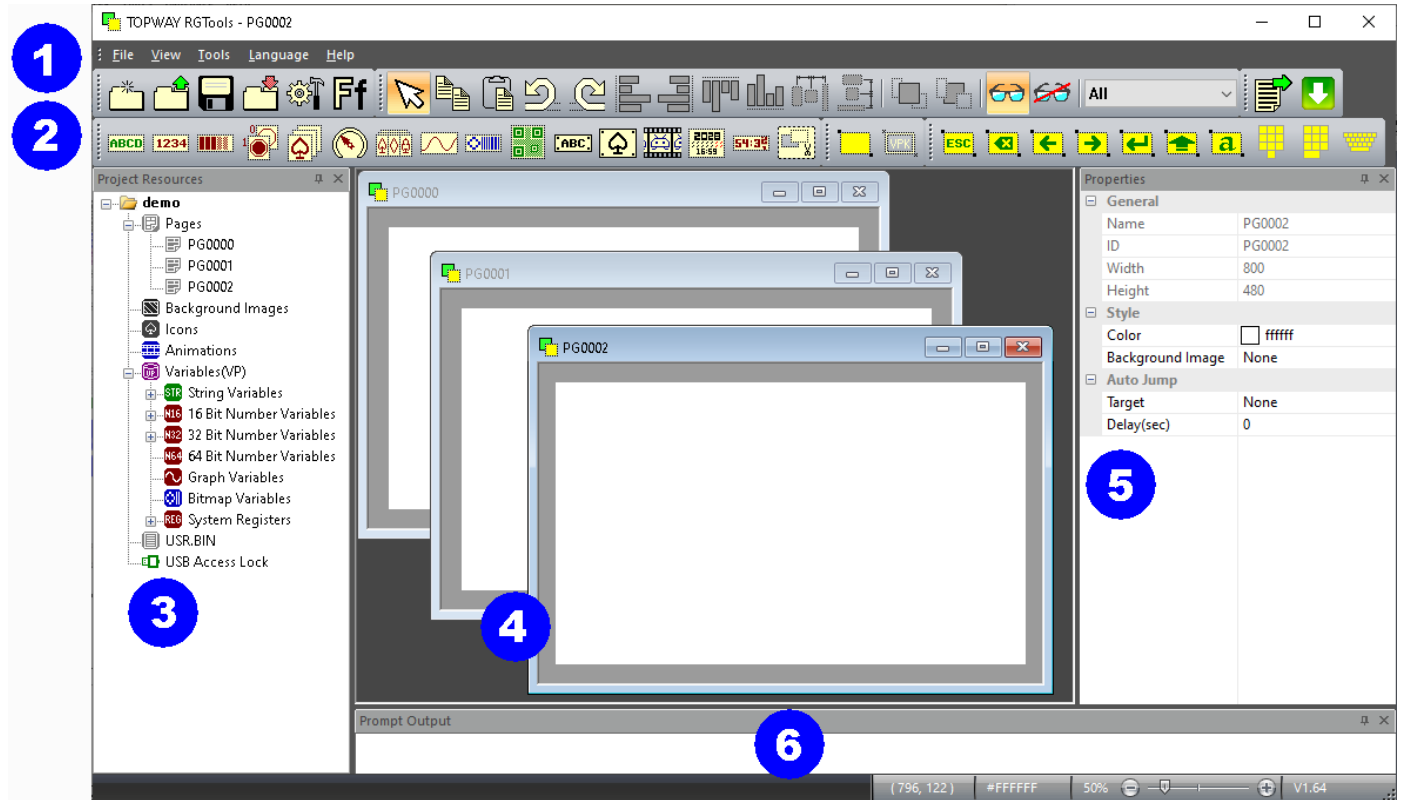
Icon	Mnemonic	Name	Descriptions
	VP_STR	String Variable	String Variable with 128bit length for character strange storage (note: content must end with "\0") The memory space could be allocated in RGTools, and link and display by elements like STR, QR code, etc.
	VP_N16	16bit Integer Variable	range: -32767~32767 The memory space could be allocated in RGTools, and linked and display by elements like N16, IDX, I16, VPK, etc.
	VP_N32	32bit Integer Variable	range: -2147483647 ~ 2147483647 The memory space could be allocated in RGTools, and linked and display by elements like N32, I32, etc.
	VP_N64	64bit Integer Variable	range: -9223372036854775807 ~ 9223372036854775806 (-9223372036854775808 (0x8000 0000 0000 0000) is reserved value) (9223372036854775807 (0x7FFF FFFF FFFF FFFF) is reserved value) The memory space could be allocated in RGTools, and linked and display by elements like N64, etc.
	VP_G16	16bit Graph Variable	groups of 16bit integer values for graphical-curve display data storage Range: -32767 ~32767 (-32768d (0xFFFF) is reserved value) The memory space could be allocated in RGTools, and linked/display by elements G16, etc
	VP_BP1	Bitmap Variable	groups of picture data in 1bpp The memory space could be allocated in RGTools, and linked/display by element BP1
	VP_REG	Register Variable	Special Register Variable for some of the system configuration. Timer Control, RTC Control, Buzzer Control, Backlight Control, Screen Saver Control, Code Page Control, Country Code Control.
	VP_TMR	Timer Variable	Timer counter inside VP_N32

Note: VP variable is inside the Smart TFT LCD Module RAM.

4 RGTools

UI designs are implemented under usage of **RGTools** allowing designs with zero coding. It dramatically simplifies and speeds up the whole product design process.

4.1 RGTools Layout



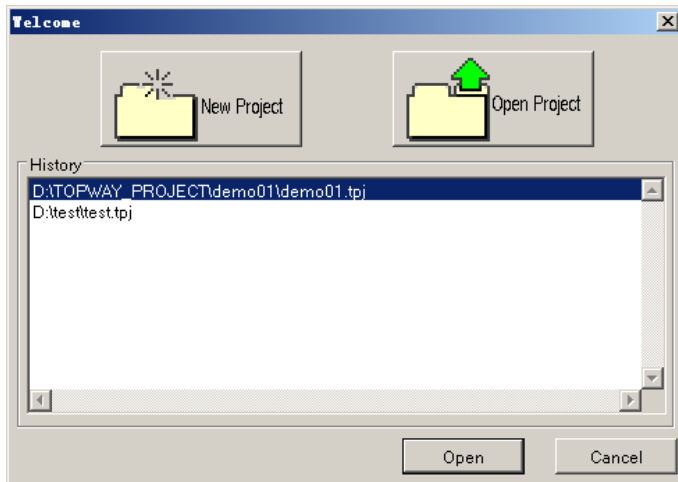
① Menu	Provide basic software operation, View options, tools options, etc
② Tools bar	There are four type of tools - file tools for project open, save, compile output, etc - alignment and display filter tools - display elements tools - compile and download tools
③ Project Resources window	Resource windows (right click on the resources) - built new page, - import pictures (IMG_BKG, IMG_ICO, IMG_ANI) - allocated VP variable (VP_N16, VP_N32, VP_N64) - user file, etc...
④ Working Area	The working area for composing the display page. User could build element onto the page.
⑤ Properties Window	Display the selected element properties or Page properties
⑥ Prompt Output Window	Prompt output window show the compiling information, warning and error information

4.2 General Operations

Please refer to the "Examples" section for general operation examples

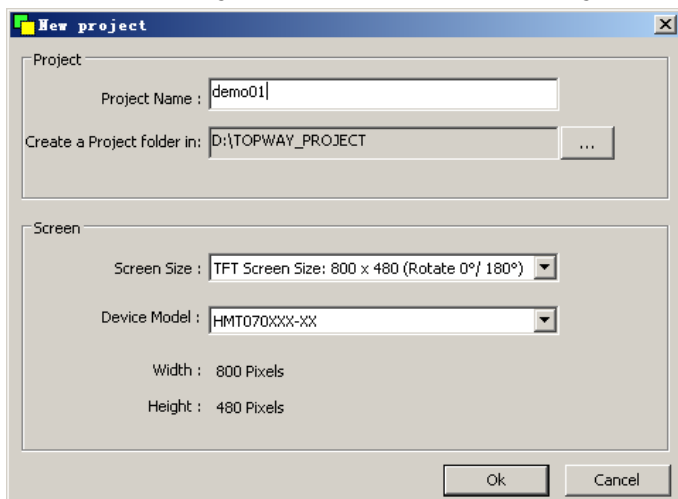
4.3 RGTools Menu

4.3.1 Start



At the start up of the RGTools, It will pop up a Welcome interface for use to select the previous Edited Project, Create a New Project or Open Project somewhere else.

4.3.2 New Project (Menu - File - New Project)



For New Project, user can type in the project name and select the folder for that project.

It is important to ensure to select the correct screen size that match the target Smart TFT LCD Module.

After click the OK button, RGTools will create two folders in to the Project Folder.

In this case, it creates
D:\TOPWAY_PROJECT\demo01
D:\TOPWAY_PROJECT\demo01.tmp
(.tmp folder is the working folder)

4.3.3 Open Project (Menu - File - Open Project)



Open Project look for the project folder with .tpj file inside

4.3.4 Save (Menu - File - Save)



Save the current project

4.3.5 Save As (Menu - File - Save As)

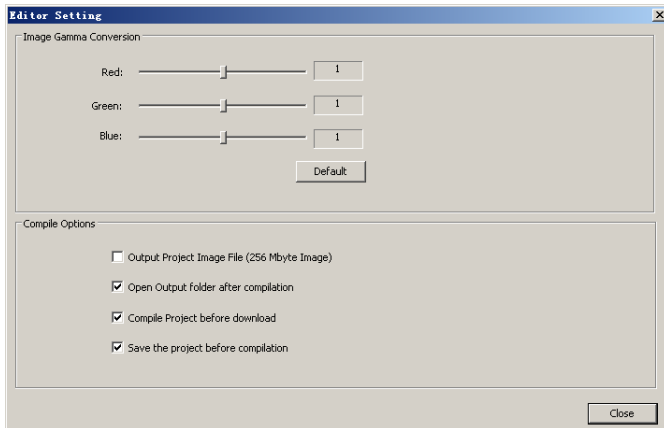
Save the current project with another name

4.3.6 Close Project (Menu - File - Close)



Close the current project

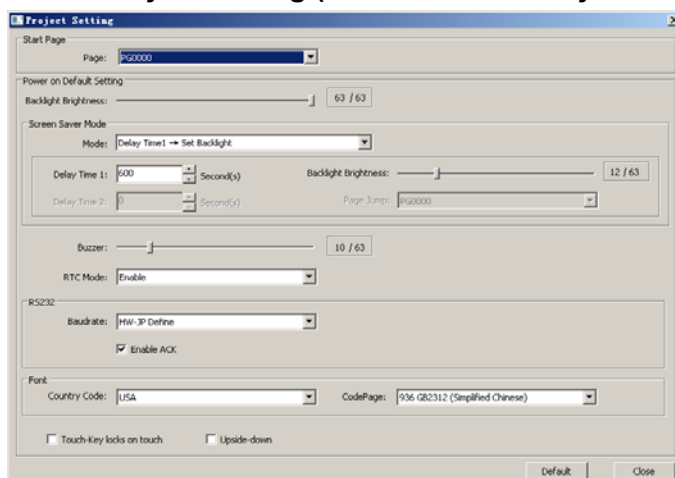
4.3.7 RGTools Options (Menu - Tools - Options)



In this window, user can

- fine adjust all the image color tone during compile by setting Image Gamma
- Select Compile Options
- Output Project Image File
- Open Output folder after compile
- Compile Project before download
- Save the project before compile

4.3.8 Project Setting (Menu - Tools - Project Setting)



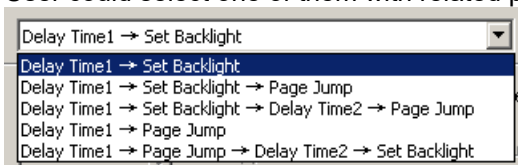
In this window, user can set

- "Start Page" to show at power on
- power on "Backlight Brightness" level
- select one of the "Screen Saver Mode" (*1)
- set "Buzzer" beeping length (0=no sound)
- set RTC (real time clock) operation mode
- set power on "RS232 Baudrate"
- set power on "Country Code" for ASCII display
- set power on "Code Page" for ext. char & decoding
- set "Touch-Key lock on touch" to lock the touch down key and action on release
- set "Upside-down" to show the generate 180 deg rotated display content

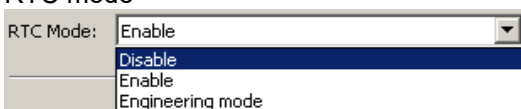
Note:

*1. Screen Saver Mode

User could select one of them with related parameter



*2. RTC mode



RTC Mode	Operating with RTC battery	Operating without RTC battery	Descriptions
Enable	1st power up takes 6s; next power up takes 1s	every power up takes 6s	RTC run with correct init
Disable	every power up takes 1s	every power up takes 1s	RTC stop
Engineering Mode	every power up takes 1s	every power up takes 1s	RTC run without init RTC might not run correctly It is NOT suggested for production setting

4.3.9 Font Setting (Menu - Tools - Font Setting)

Ff

There are 2 sections of Fonts.

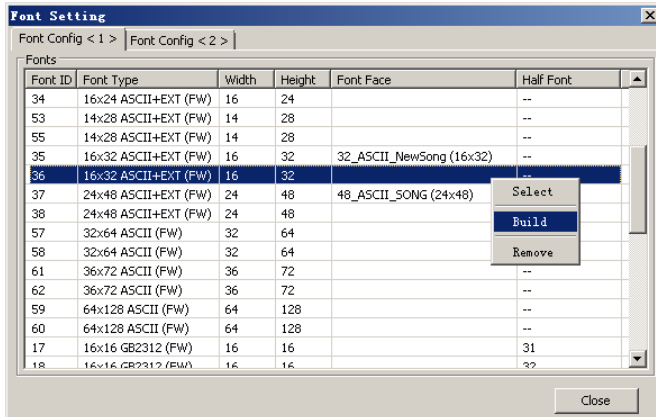
Font Config <1>, the Font Width and Height are fixed and mainly for ASCII Char.

(with some default font, vary by model)

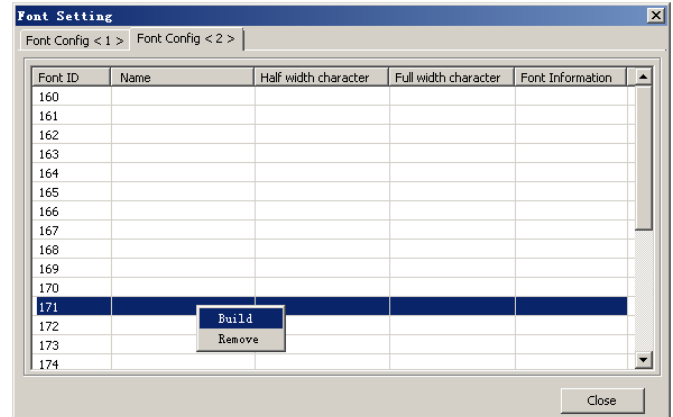
Font Config <2>, provide more free option about Size and Decoding selection. (without font by default)

Right click on the font table, select "build" could build the font with several options.

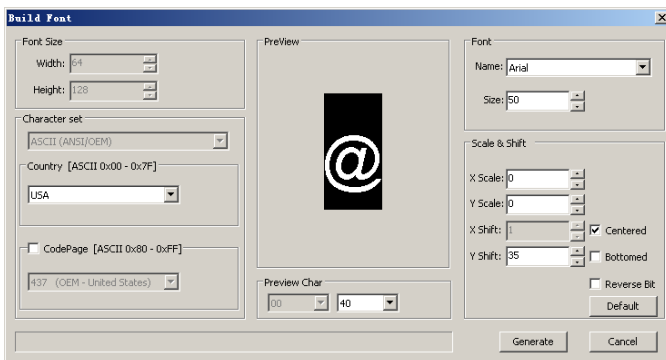
It is suggested to delete the un-used Font area for reduce the memory allocation.



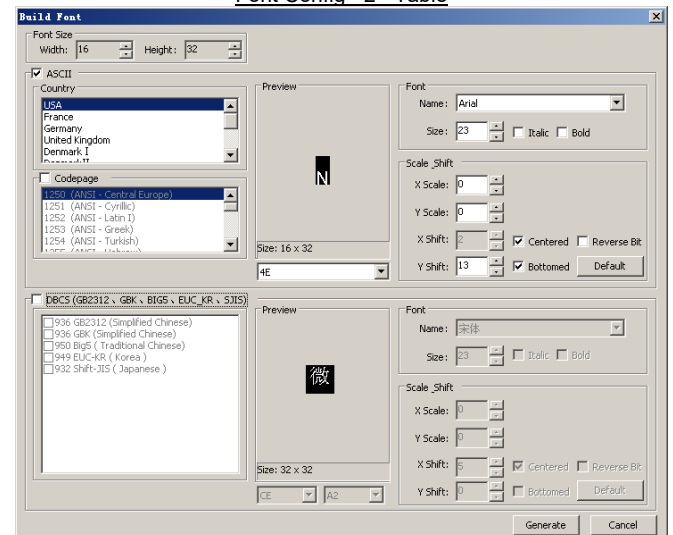
Font Config <1> Table



Font Config <2> Table



Build Font <1>



Build Font <2>

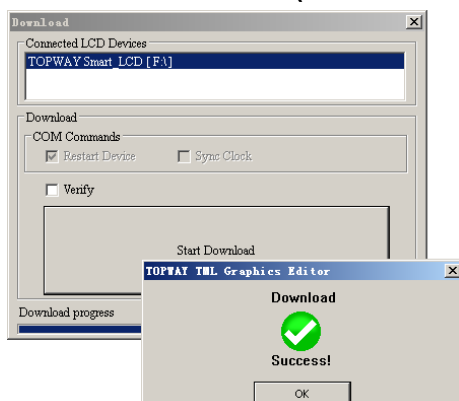
4.3.10 Built Project Files (Menu - Tools - Built Project Files)



After finish the layout, "Built Project Files" will compile all the resource and font etc in to a set of files. It could download or copy to the Smart TFT LCD Module at any time.

It will pop-up the output folder inside the Project Folder

4.3.11 Download to Module (Menu - Tools - Download to Module)

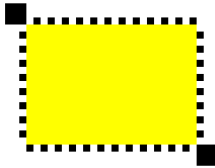


After connect the Smart TFT LCD Module USB to user PC, the Smart TFT LCD Module will appear as a removable drive. Using "Download to Module" could copy the compiled project file into the Smart TFT LCD Module.

Note. RGTools will refuse to download without Smart TFT LCD Module

4.4 Elements Configurations

4.4.1 Touch Key (TPK)



Properties	
General	
Name	Touch Key
ID	0
X	12
Y	21
Width	60
Height	60
Visual Effect	
On Press Down	None
Icon/BgImg	None
Target	
Target Page	None
Call	
Call Keyboard	None
Call Operation	None
Call Key	None
Content	
VP Address	None
Compile As	Auto
Data Type	Signed
Return	
Return Value	None
Real Time Properties Control	
Enable VP	None

General

- Show the basic information of the element
- X and Y is the location of element on screen (top-left corner of the screen = 0,0)
element top-left corner is the reference point
- Width and Height are in pixels

Visual Effect

- On Press Down options are
None, Inverse Color, Show Icon or Show Copped Image
- Icon/BgImage base on the above option
select a Background Image or an Icon for the visual effect.

Target

- Target Page is for page jump function after touch

Call

- Call Keyboard is for call up a keyboard after touch
- Call Operation is for simple calculation after touch
- Call Key is for building PIP_Keyboard

Context

- VP Address is for calculation or Keyboard input result
- Compile As can force VP variable work different than default (Auto)
VP_N16 (two successive VP_N16) work as VP_N32
VP_N32 (first two byte of VP_N32) work as VP_N16
VP_N64 (first two byte of VP_N64) work as VP_N16
VP_N64 (first four byte of VP_N64) work as VP_N32
- Data type should be signed by default (*1)

Return

- Return Value for select type of notification to host during touch
It can notify the Host via Touch Key ID Response code (0x78) (*3)
none, no notification (by default)
Up PGID + TPKID, send on TPK release
Down PGID + TPKID, send on TPK touch
Down Up PGID + TPKID, send on TPK touch and release

Parameter modification (advance feature)

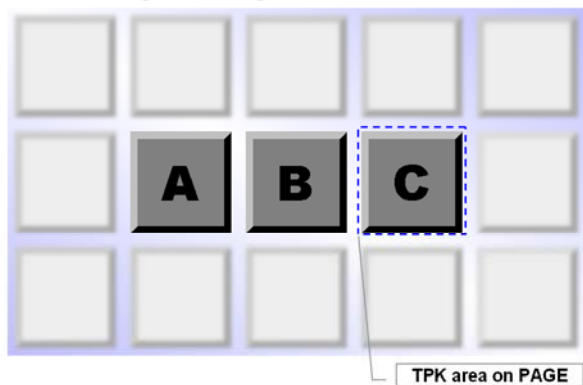
- Enable VP is none by default (element operate normally)
Select a VP for real time element control.
VP value will set to 1 as enable at power-on.
It can be disabled (hidden) by a zero value

Note:

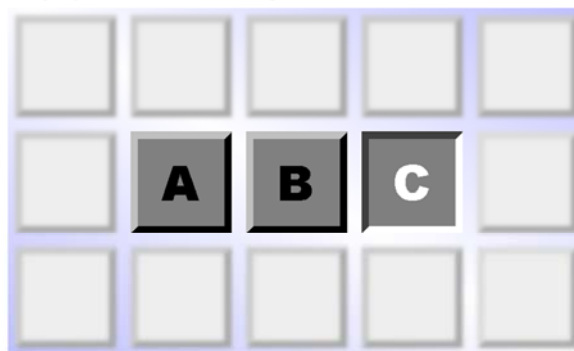
- *1. Please also see "TPK, VPK Call Functions" section for Call Features
- *2. TPK support touch and move, action on release. It could change to "Touch-Key lock on touch" in "Project Setting".
- *3. Please also see "Command" section for details

***4. TPK Visual Effect - Inverse Color Operation Idea**

PAGE's Background Image



Display Result of Touching the TPK on 'C'

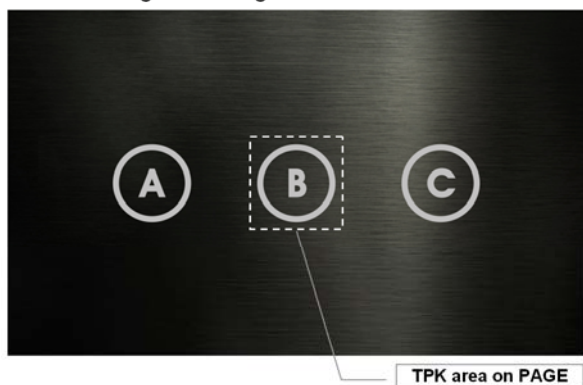


Properties Highlights

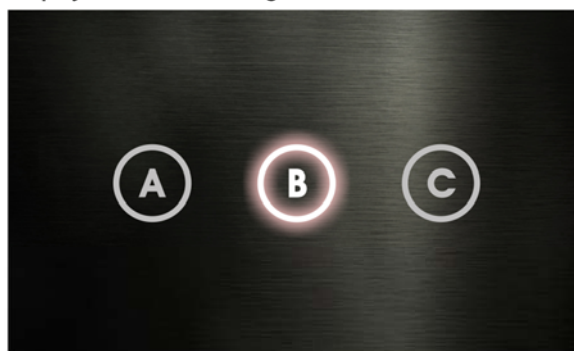
On Press Down	Inverse
Icon/BgImage	--

***5. TPK Visual Effect - Show Cropped BgImage Operation Idea**

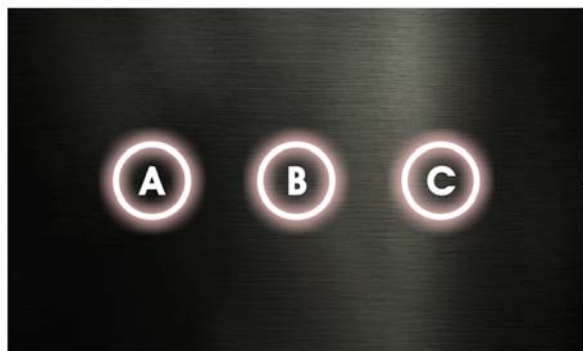
PAGE's Background Image



Display Result of Touching the TPK on 'B'



Icon / BgImage

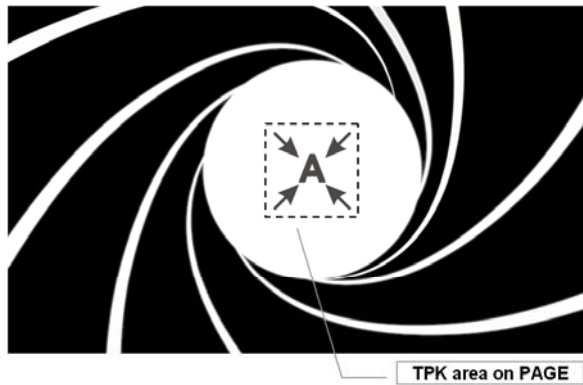


Properties Highlights

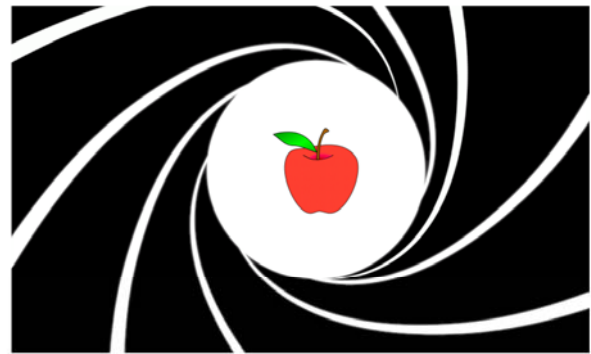
On Press Down	Show Cropped BgImg
Icon/BgImage	Background Image

*6. TPK Visual Effect - Show Icon Operation Idea

PAGE's Background Image



Display Result of Touching the TPK on 'A'



Icon / BgImage

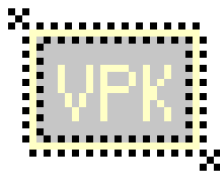


Properties Highlights

On Press Down	Show Icon
Icon/BgImage	Icon

*7. Floating point value could only be operate with N32 (or Compile As N32) and PIP Number Keyboard

4.4.2 Virtual Key (VPK)



General

- Show the basic information of the element
- X and Y is the location of element on screen (top-left corner of the screen = 0,0)
element top-left corner is the reference point
- Width and Height are in pixels

Condition

- Monitor-VP is a VP to trigger this Virtual Key
- Monitor-Value is the Value for triggering this element

Target

- Target Page is for page jump function after touch

Call

- Call Keyboard is for call up a keyboard after touch
- Call Operation is for simple calculation after touch
- Call Key is for building PIP_Keyboard

Context

- VP Address is for calculation or Keyboard input result
- Compile As can force VP variable work different than default (Auto)
VP_N16 (two successive VP_N16) work as VP_N32
VP_N32 (first two byte of VP_N32) work as VP_N16
VP_N64 (first two byte of VP_N64) work as VP_N16
VP_N64 (first four byte of VP_N64) work as VP_N32
- Data type should be signed by default (*1)

Parameter modification (advance feature)

- Enable VP is none by default (element operate normally)
Select a VP for real time element control.
VP value will set to 1 as enable at power-on.
It can be disabled (hidden) by a zero value

Properties	
General	
Name	Virtual Key
ID	0
X	86
Y	21
Width	60
Height	62
Condition	
Monitor-VP	None
Monitor-Value	1
Target	
Target Page	None
Call	
Call Keyboard	None
Call Operation	None
Call Key	None
Content	
VP Address	None
Compile As	Auto
Data Type	Signed
Real Time Properties Control	
Enable VP	None

Note:

*1. Please also see "TPK, VPK Call Functions" section for Call Features

*2. Floating point value could only be operate with N32 (or Compile As N32) and PIP Number Keyboard

4.4.3 Static String (STS)



Properties	
General	
Name	Static String
ID	0
X	12
Y	116
Width	60
Height	60
Style	
Font	24_ASCII_Sys
Font Color	000000
Background Color	ffffff
Transparent	True
Format	
Align	Left
Content	
Static Text	
Real Time Properties Control	
Enable VP	None
Font VP	None
Foreground Color VP	None
Background Color VP	None
Transparent VP	None

General

- Show the basic information of the element
- X and Y is the location of element on screen (top-left corner of the screen = 0,0)
element top-left corner is the reference point
- Width and Height are in pixels

Style

- Font face could be selected at here
- Font Color is the font face color
- Background Color only available with Transparent disable
- Transparent = true, disable the Background Color box to be show

Format

- Align could be: Left / Right / Center

Content

- Static Text is the text going to be display

Parameter modification (advance feature)

- Enable VP is none by default (element operate normally)
Select a VP for real time element control.
VP value will set to 1 as enable at power-on.
It can be disabled (hidden) by a zero value
- Font VP is none by default (using the above selected font)
Select a VP for real time font ID change
VP value will copy the above ID at power-on.
- Foreground color VP is none by default (using the above selected color)
Select a VP for real time foreground color adjustment
VP value will copy the above selected color at power-on.
- Background color VP is none by default (using the above selected color)
Select a VP for real time background color adjustment
VP value will copy the above selected color at power-on.
- Transparent VP is none by default (using the above option)
Select a VP for real time transparent background control
VP value will copy the above transparent config at power-on.
It can be enable by a non-zero value or disable by a zero

4.4.4 String Element (STR)



Properties	
General	
Name	String Variable
ID	0
X	86
Y	116
Width	60
Height	60
Style	
Font	24_ASCII_Sys
Font Color	000000
Background Color	ffffff
Transparent	True
Mask	
Format	
Align	Left
Char Spacing	Default
Content	
VP Resource	VP_STR
VP Address	None
Length	127
Real Time Properties Control	
Enable VP	None
Font VP	None
Foreground Color VP	None
Background Color VP	None
Transparent VP	None
Character Preview	
Character	Abc

General

- Show the basic information of the element
- X and Y is the location of element on screen (top-left corner of the screen = 0,0)
element top-left corner is the reference point
- Width and Height are in pixels

Style

- Font face could be selected at here
- Font Color is the font face color
- Background Color only available with Transparent disable
- Transparent = true, disable the Background Color box to be show
- Mask could be any char to "represent" as the char (Keep empty for normal operation)

Format

- Align could be:
Left / Right / Center / Multi-line Left / Multi-line Right / Multi line center
- Char spacing option: Default, Shrink

Content

- VP Resource is the type of VP hold the text content (VP_STR by default)
- VP Address is the VP that hold the string content (all the string should be end with \0 (0x00))
- Length is no. of Char going to display

Parameter modification (advance feature)

- Enable VP is none by default (element operate normally)
Select a VP for real time element control.
VP value will set to 1 as enable at power-on.
It can be disabled (hidden) by a zero value
- Font VP is none by default (using the above selected font)
Select a VP for real time font ID change
VP value will copy the above ID at power-on.
- Foreground color VP is none by default (using the above selected color)
Select a VP for real time foreground color adjustment
VP value will copy the above selected color at power-on.
- Background color VP is none by default (using the above selected color)
Select a VP for real time background color adjustment
VP value will copy the above selected color at power-on.
- Transparent VP is none by default (using the above option)
Select a VP for real time transparent background control
VP value will copy the above transparent config at power-on.
It can be enable by a non-zero value or disable by a zero

Character Preview

- Character is for simulation in RGTools environment.

4.4.5 Number Elements (N16, N32, N64)



Properties	
General	
Name	Number Vari...
ID	0
X	160
Y	116
Width	60
Height	60
Style	
Font	24_ASCII_Sys
Font Color	000000
Background Color	ffffff
Transparent	True
Mask	
Format	
Align	Left
Integer Digits	Auto
Decimal Digits	0
Content	
VP Resource	VP_N16
VP Address	None
Advance	
Compile As	Auto
Data Type	Signed
Real Time Properties Control	
Enable VP	None
Font VP	None
Foreground Color VP	None
Background Color VP	None
Transparent VP	None
Character Preview	
Character	123

General

- Show the basic information of the element
- X and Y is the location of element on screen (top-left corner of the screen = 0,0)
element top-left corner is the reference point
- Width and Height are in pixels

Style

- Font face could be selected at here
- Font Color is the font face color
- Background Color only available with Transparent disable
- Transparent = true, disable the Background Color box to be show
- Mask could be any char to "represent" as the char (Keep empty for normal operation)

Format

- Align could be: Left / Right / Center
- Integer Digits is the no. of digit on left side of the dot (default Auto) (*1)(*2)
- Decimal Digits is the no of digit on right side of the dot (*1)(*2)

Content

- VP Resource is the type of VP hold the content (VP_N16 by default)
- VP Address is the VP that hold the value

Advance

- Compile As can force VP variable work different than default (Auto)
VP_N16 (two successive VP_N16) work as VP_N32
VP_N32 (first two byte of VP_N32) work as VP_N16
VP_N64 (first two byte of VP_N64) work as VP_N16
VP_N64 (first four byte of VP_N64) work as VP_N32
- Data type as signed by default for the decoding of the RAM content
It could also present unsigned value or floating value, too
(please make sure the value in the VP variable appropriate for the type)

Parameter modification (advance feature)

- Enable VP is none by default (element operate normally)
Select a VP for real time element control.
VP value will set to 1 as enable at power-on.
It can be disabled (hidden) by a zero value
- Font VP is none by default (using the above selected font)
Select a VP for real time font ID change
VP value will copy the above ID at power-on.
- Foreground color VP is none by default (using the above selected color)
Select a VP for real time foreground color adjustment
VP value will copy the above selected color at power-on.
- Background color VP is none by default (using the above selected color)
Select a VP for real time background color adjustment
VP value will copy the above selected color at power-on.
- Transparent VP is none by default (using the above option)
Select a VP for real time transparent background control
VP value will copy the above transparent config at power-on.
It can be enable by a non-zero value or disable by a zero

Character Preview

- Character is for simulation in RGTools environment.

Note:

*1. Integer Digits / Decimal Digits Formatting Examples

VP's value (10's)	Integer Digits	Decimal Digits	Display Result	Descriptions
15	Auto	3	0.015	Decimal Digits = 3, zero prefix is added. Integer Digits = Auto, a minimal 0 is shown
23762	Auto	3	23.762	Decimal Digits = 3, least 3 digit as decimal value Integer Digits = Auto, unlimited Integer digit
5629	5	1	00562.9	Decimal Digits = 1, least 1 digit as decimal value Integer Digits = 5, zero prefix is added
-87913	3	2	-879.13	Decimal Digits = 2, two decimal Integer Digits = 3, show all the rest as integer Negative signed value use "-" as prefix
-13277	2	2	-99.99	The VP's value over the maximum value that could be display, the max value (within the display format) shown with sign.
1758	3	0	999	The VP's value over the maximum value that could be display, the max value (within the display format) shown.

*2. Float type value might rounded or truncated due to hex-dec conversion

*3. Float type value is suggested to select: Integer Digits=Auto, Decimal Digits=1~9.
to prevent over maximum value presentation error

4.4.6 Timer Display (TMR)



Properties	
General	
Name	Timer
ID	0
X	234
Y	116
Width	60
Height	60
Style	
Font	24_ASCII_Sys
Font Color	000000
Background Color	ffffff
Transparent	True
Format	
Align	Left
Time Format	HH:mm:ss
Content	
Timer	None
Real Time Properties Control	
Enable VP	None
Font VP	None
Foreground Color VP	None
Background Color VP	None
Transparent VP	None

General

- Show the basic information of the element
- X and Y is the location of element on screen (top-left corner of the screen = 0,0)
element top-left corner is the reference point
- Width and Height are in pixels

Style

- Font face could be selected at here
- Font Color is the font face color
- Background Color only available with Transparent disable
- Transparent = true, disable the Background Color box to be show

Format

- Align could be: Left / Right / Center
- Time Format could be: HH:mm:ss / mm:ss / ss

Content

- Timer: select a timer specific inside VP_N32 area

Parameter modification (advance feature)

- Enable VP is none by default (element operate normally)
Select a VP for real time element control.
VP value will set to 1 as enable at power-on.
It can be disabled (hidden) by a zero value
- Font VP is none by default (using the above selected font)
Select a VP for real time font ID change
VP value will copy the above ID at power-on.
- Foreground color VP is none by default (using the above selected color)
Select a VP for real time foreground color adjustment
VP value will copy the above selected color at power-on.
- Background color VP is none by default (using the above selected color)
Select a VP for real time background color adjustment
VP value will copy the above selected color at power-on.
- Transparent VP is none by default (using the above option)
Select a VP for real time transparent background control
VP value will copy the above transparent config at power-on.
It can be enable by a non-zero value or disable by a zero

4.4.7 Real Time Clock (RTC)



Properties	
General	
Name	Real Time Cl...
ID	0
X	308
Y	116
Width	60
Height	60
Style	
Font	24_ASCII_Sys
Font Color	000000
Background Color	ffffff
Transparent	True
Format	
Date/Time Format	HH:mm:ss
Real Time Properties Control	
Enable VP	None
Font VP	None
Foreground Color VP	None
Background Color VP	None
Transparent VP	None

General

- Show the basic information of the element
- X and Y is the location of element on screen (top-left corner of the screen = 0,0)
element top-left corner is the reference point
- Width and Height are in pixels

Style

- Font face could be selected at here
- Font Color is the font face color
- Background Color only available with Transparent disable
- Transparent = true, disable the Background Color box to be show

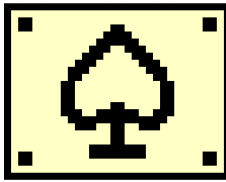
Format

- Date / Time Format could be shown as :
YYYY-MM-dd / YYYY-MM-dd HH:mm:ss /
W YYYY-MM-dd / W YYYY-MM-dd HH:mm:ss /
YYYY / MM / DD / HH / mm / ss

Parameter modification (advance feature)

- Enable VP is none by default (element operate normally)
Select a VP for real time element control.
VP value will set to 1 as enable at power-on.
It can be disabled (hidden) by a zero value
- Font VP is none by default (using the above selected font)
Select a VP for real time font ID change
VP value will copy the above ID at power-on.
- Foreground color VP is none by default (using the above selected color)
Select a VP for real time foreground color adjustment
VP value will copy the above selected color at power-on.
- Background color VP is none by default (using the above selected color)
Select a VP for real time background color adjustment
VP value will copy the above selected color at power-on.
- Transparent VP is none by default (using the above option)
Select a VP for real time transparent background control
VP value will copy the above transparent config at power-on.
It can be enable by a non-zero value or disable by a zero

4.4.8 Static Icon (ICO)



Properties	
General	
Name	Static Icon
ID	0
X	12
Y	211
Width	60
Height	60
Style	
Transparent	False
Transparent Color	<input type="checkbox"/> ffffff
Content	
Icon	None
Real Time Properties Control	
Enable VP	None
ICON VP	None
Transparent Color VP	None
Transparent VP	None

General

- Show the basic information of the element
- X and Y is the location of element on screen (top-left corner of the screen = 0,0)
element top-left corner is the reference point
- Width and Height are in pixels

Style

- Transparent = False by default
True: enable transparent effect to show the IMG_ICO
Auto: enable transparent effect and pick a high possible color as Transparent color
- Transparent Color one color in the IMG_ICO could show as transparent. (it is suggested to use pure color for transparent effect) (e.g. pure-magenta pure-black, pure-white, etc...)

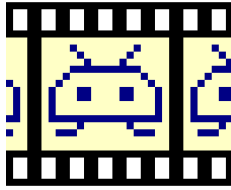
Content

- ICON select a IMG_ICO to show

Parameter modification (advance feature)

- Enable VP is none by default (element operate normally)
Select a VP for real time element control.
VP value will set to 1 as enable at power-on.
It can be disabled (hidden) by a zero value
- ICON VP is none by default (using the above selected font)
Select a VP for real time font IMG_ICON change
VP value will copy the above IMG_ICO ID at power-on.
- Transparent color VP is none by default (using the above selected color)
Select a VP for real time transparent color change
VP value will copy the above selected color at power-on.
- Transparent VP is none by default (using the above option)
Select a VP for real time transparent background control
VP value will copy the above transparent config at power-on.
It can be enable by a non-zero value or disable by a zero

4.4.9 Animation Element (ANI)



Properties													
<div> <div>General</div> <table> <tr> <td>Name</td> <td>Animation</td> </tr> <tr> <td>ID</td> <td>0</td> </tr> <tr> <td>X</td> <td>86</td> </tr> <tr> <td>Y</td> <td>211</td> </tr> <tr> <td>Width</td> <td>60</td> </tr> <tr> <td>Height</td> <td>60</td> </tr> </table> </div>		Name	Animation	ID	0	X	86	Y	211	Width	60	Height	60
Name	Animation												
ID	0												
X	86												
Y	211												
Width	60												
Height	60												
<div> <div>Style</div> <table> <tr> <td>Transparent</td> <td>False</td> </tr> <tr> <td>Transparent Color</td> <td><input type="checkbox"/> fffffff</td> </tr> </table> </div>		Transparent	False	Transparent Color	<input type="checkbox"/> fffffff								
Transparent	False												
Transparent Color	<input type="checkbox"/> fffffff												
<div> <div>Format</div> <table> <tr> <td>Loop-Play</td> <td>False</td> </tr> <tr> <td>Play Speed</td> <td>100</td> </tr> </table> </div>		Loop-Play	False	Play Speed	100								
Loop-Play	False												
Play Speed	100												
<div> <div>Content</div> <table> <tr> <td>Animation</td> <td>None</td> </tr> </table> </div>		Animation	None										
Animation	None												
<div> <div>Real Time Properties Control</div> <table> <tr> <td>Enable VP</td> <td>None</td> </tr> <tr> <td>Animation VP</td> <td>None</td> </tr> <tr> <td>Transparent Color VP</td> <td>None</td> </tr> <tr> <td>Transparent VP</td> <td>None</td> </tr> </table> </div>		Enable VP	None	Animation VP	None	Transparent Color VP	None	Transparent VP	None				
Enable VP	None												
Animation VP	None												
Transparent Color VP	None												
Transparent VP	None												

General

- Show the basic information of the element
- X and Y is the location of element on screen (top-left corner of the screen = 0,0)
element top-left corner is the reference point
- Width and Height are in pixels

Style

- Transparent = False by default
True: enable transparent effect to show the IMG_ANI
- Transparent Color one color in the IMG_ANI could show as transparent.
(it is suggested to use pure color for transparent effect)
(e.g. pure-magenta pure-black, pure-white, etc...)

Format

- Loop-Play = false by default, it shows the last frame on screen
- Play Speed = 100ms per frame by default

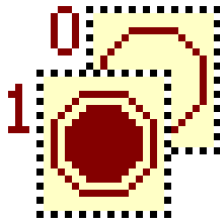
Content

- Animation select a IMG_ANI to show

Parameter modification (advance feature)

- Enable VP is none by default (element operate normally)
Select a VP for real time element control.
VP value will set to 1 as enable at power-on.
It can be disabled (hidden) by a zero value
- ICON VP is none by default (using the above selected font)
Select a VP for real time font IMG_ICON change
VP value will copy the above IMG_ICO ID at power-on.
- Transparent color VP is none by default (using the above selected color)
Select a VP for real time transparent color change
VP value will copy the above selected color at power-on.
- Transparent VP is none by default (using the above option)
Select a VP for real time transparent background control
VP value will copy the above transparent config at power-on.
It can be enable by a non-zero value or disable by a zero

4.4.10 Bit Icon (IDX_BIT)



Properties	
General	
Name	Bit Icon
ID	0
X	160
Y	211
Width	60
Height	60
Style	
Transparent	False
Transparent Color	<input type="checkbox"/> ffffff
Content	
VP Resource	VP_N16
VP Address	None
Bit Position	0
ICON_1	None
ICON_0	None
Real Time Properties Control	
Enable VP	None
ICON_1_VP	None
ICON_0_VP	None
Transparent Color VP	None
Transparent VP	None

General

- Show the basic information of the element
- X and Y is the location of element on screen (top-left corner of the screen = 0,0)
element top-left corner is the reference point
- Width and Height are in pixels

Style

- Transparent = False by default
True: enable transparent effect to show the IMG_ICO
- Transparent Color one color in the IMG_ICO could show as transparent. (it is suggested to use pure color for transparent effect)
(e.g. pure-magenta pure-black, pure-white, etc...)

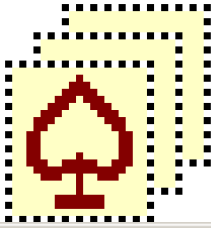
Content

- VP Resource is the type of VP hold the content (VP_N16 by default)
- VP Address is the VP that hold a value
- Bit Position select a bit in the VP as monitor bit
- ICON_1 is none by default (monitor bit content = 1; no icon to show)
Select a IMG_ICO to show when monitor bit content = 1
- ICON_0 is none by default (monitor bit content = 0; no icon to show)
Select a IMG_ICO to show when monitor bit content = 0

Parameter modification (advance feature)

- Enable VP is none by default (element operate normally)
Select a VP for real time element control.
VP value will set to 1 as enable at power-on.
It can be disabled (hidden) by a zero value
- ICON_1_VP is none by default (using the above selected font)
Select a VP for real time change the IMG_ICO ID (monitor bit content = 1)
VP value will copy the above ID at power-on.
- ICON_0_VP is none by default (using the above selected font)
Select a VP for real time change the IMG_ICO ID (monitor bit content = 0)
VP value will copy the above ID at power-on.
- Transparent color VP is none by default (using the above selected color)
Select a VP for real time transparent color change
VP value will copy the above selected color at power-on.
- Transparent VP is none by default (using the above option)
Select a VP for real time transparent background control
VP value will copy the above transparent config at power-on.
It can be enable by a non-zero value or disable by a zero

4.4.11 Indexed Icon (IDX)



Properties	
General	
Name	Indexed Icon
ID	0
X	234
Y	211
Width	60
Height	60
Style	
Transparent	False
Transparent Color	<input type="checkbox"/> ffffff
Content	
VP Resource	VP_N16
VP Address	None
First Icon	None
Min Value	0
Max Value	0
Real Time Properties Control	
Enable VP	None
ICON VP	None
Transparent Color VP	None
Transparent VP	None

General

- Show the basic information of the element
- X and Y is the location of element on screen (top-left corner of the screen = 0,0)
element top-left corner is the reference point
- Width and Height are in pixels

Style

- Transparent = False by default
True: enable transparent effect to show the IMG_ICO
- Transparent Color one color in the IMG_ICO could show as transparent. (it is suggested to use pure color for transparent effect)
(e.g. pure-magenta pure-black, pure-white, etc...)

Content

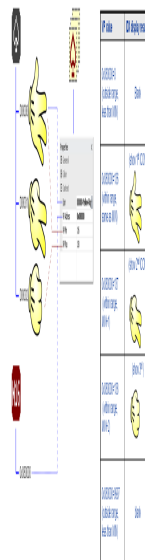
- VP Resource is the type of VP hold the content (VP_N16 by default)
- VP Address is the VP that hold the value
- First Icon is the first IMG_ICO of a series of icons to be use
- Min Value is the value that link to the first of the IMG_ICO series
- Max Value is the value that link to the last of the IMG_ICO series

Parameter modification (advance feature)

- Enable VP is none by default (element operate normally)
Select a VP for real time element control.
VP value will set to 1 as enable at power-on.
It can be disabled (hidden) by a zero value
- ICON VP is none by default (using the above selected font)
Select a VP for real time change the first IMG_ICO ID of a series of icons to be use
VP value will copy the above ID at power-on.
- Transparent color VP is none by default (using the above selected color)
Select a VP for real time transparent color change
VP value will copy the above selected color at power-on.
- Transparent VP is none by default (using the above option)
Select a VP for real time transparent background control
VP value will copy the above transparent config at power-on.
It can be enable by a non-zero value or disable by a zero

note:

*1. Indexed Icon Operation Idea



4.4.12 Tachometer (TCM)



Properties	
General	
Name	Tachometer
ID	0
X	308
Y	211
Width	60
Height	60
Background	
Icon	None
Transparent	False
Transparent Color	<input type="checkbox"/> fffffff
Foreground	
Icon	None
Transparent Mode	Normal
Transparent Color	<input checked="" type="checkbox"/> 000000
Rotation Mode	
Mode	Rotation Mode
Direction	Clockwise
Start Angle	0
End Angle	180
Rotation Center	
Rotation Center Point	Icon Center
X Coordinate	36
Y Coordinate	27
Content	
VP Resource	VP_N16
VP Address	None
Min Value	0
Max Value	10
Real Time Properties Control	
Enable VP	None

General

- Show the basic information of the element
- X and Y is the location of element on screen (top-left corner of the screen = 0,0)
element top-left corner is the reference point
- Width and Height are in pixels

Background

- Icon is an IMG_ICO selected as tachometer background
- Transparent = False by default
True: enable transparent effect to show the IMG_ICO
- Transparent Color one color in the IMG_ICO could show as transparent. (it is suggested to use pure color for transparent effect)
(e.g. pure-magenta pure-black, pure-white, etc...)

Foreground

- Icon is an IMG_ICO selected as tachometer foreground (or pointer)
The size should be same as the background one.
- Transparent = False by default
True: enable transparent effect to show the IMG_ICO
- Transparent Color one color in the IMG_ICO could show as transparent. (it is suggested to use pure color for transparent effect)
(e.g. pure-magenta pure-black, pure-white, etc...)

Rotation mode

- Mode could be rotation mode or angle opening mode
- Direction could be Clockwise or Anticlockwise
- Start Angle is corresponding to the Min Value of content (0° denote as the Foreground Icon original orientation)
- End Angle is corresponding to the Max Value of content (180° denote as the Foreground Icon rotated by 180°)

Rotation Center

- Rotation Center Point could be Icon Center (auto select) or by manual
- X, Y Coordinate define the is the rotation reference point

Content

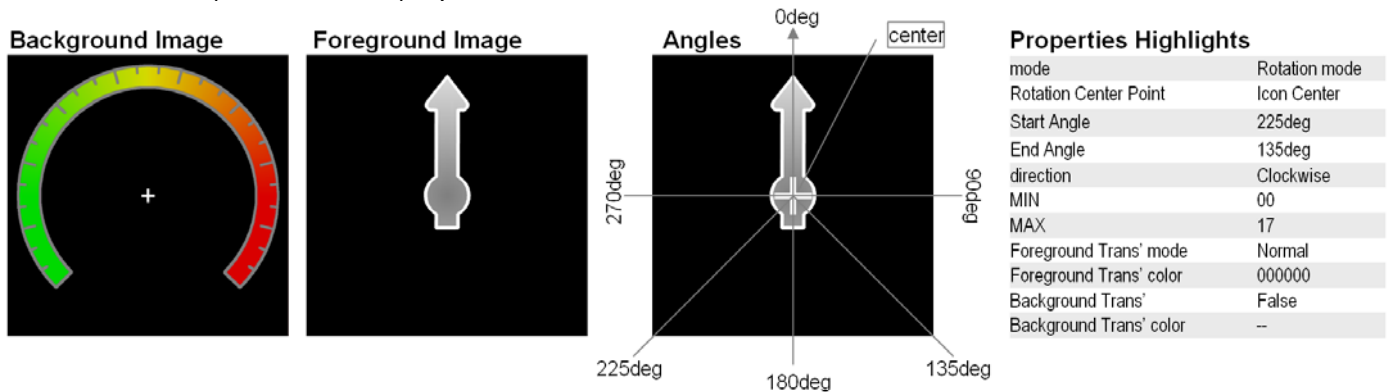
- VP Resource is the type of VP hold the content (VP_N16 by default)
- VP Address is the VP that hold the value
- Min Value is the minimum tachometer value to be show (value smaller than this value will hide the tachometer)
- Max Value is the maximum tachometer value to be show (value bigger than this value will hide the tachometer)

Parameter modification (advance feature)

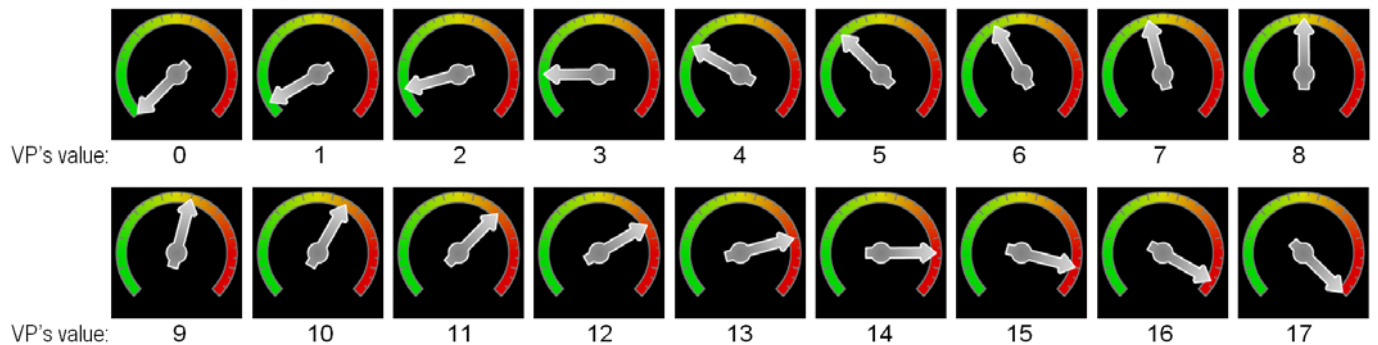
- Enable VP is none by default (element operate normally)
Select a VP for real time element control.
VP value will set to 1 as enable at power-on.
It can be disabled (hidden) by a zero value

note:

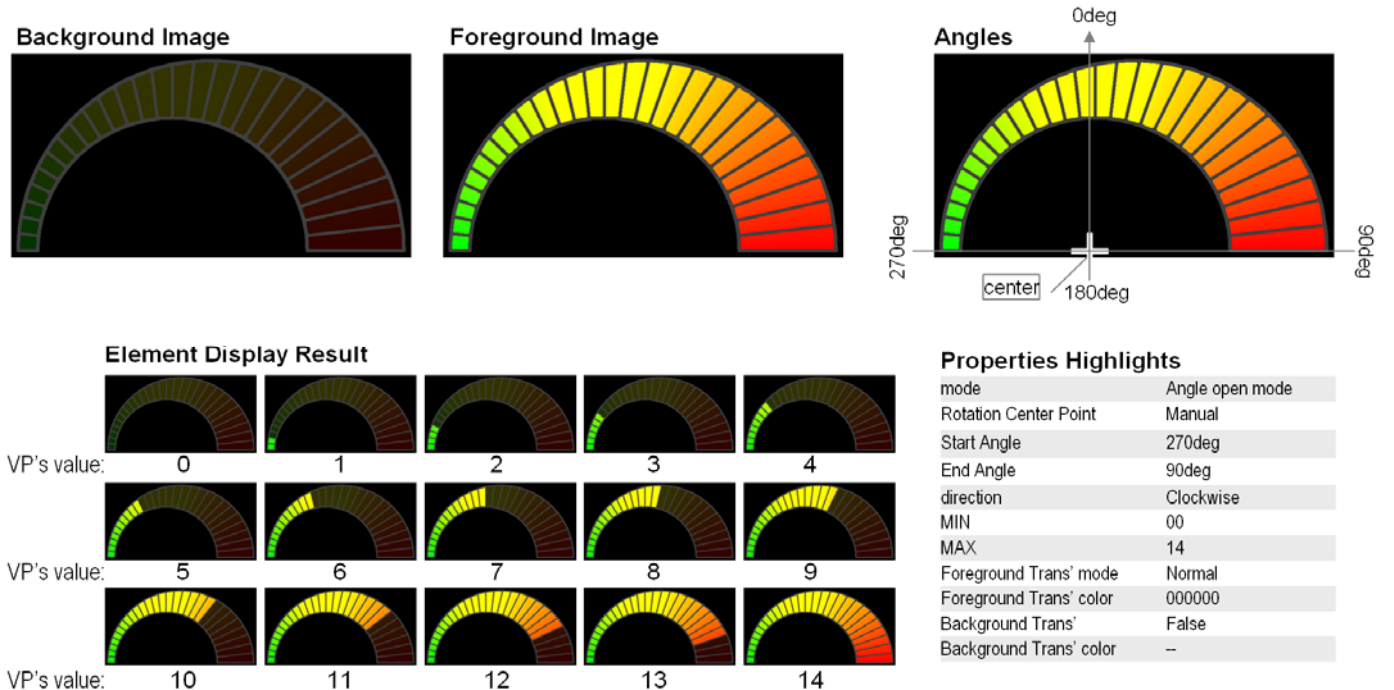
*1. Tachometer (rotation mode) Operation Idea



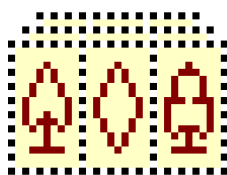
Element Display Result



*2. Tachometer (angle opening mode) Operation Idea



4.4.13 Decimal Icon (I16, I32)



Properties	
General	
Name	Decimal Icon
ID	0
X	383
Y	211
Width	60
Height	60
Style	
Transparent	False
Transparent Color	<input type="text" value="ffffff"/>
Format	
Align	Left
Integer Digits	Auto
Decimal Digits	0
Content	
VP Resource	VP_N16
VP Address	None
First Icon	None
Advance	
Compile As	Auto
Location Pointer	None
Real Time Properties Control	
Enable VP	None
ICON VP	None
Transparent Color VP	None
Transparent VP	None

General

- Show the basic information of the element
- X and Y is the location of element on screen (top-left corner of the screen = 0,0)
element top-left corner is the reference point
- Width and Height are in pixels

Style

- Transparent = False by default
True: enable transparent effect to show the IMG_ICO
- Transparent Color one color in the IMG_ICO could show as transparent.
(it is suggested to use pure color for transparent effect)
(e.g. pure-magenta pure-black, pure-white, etc...)

Format

- Align could be: Left / Right / Center
- Integer Digits is the no. of digit on left side of the dot (Auto by default)
- Decimal Digits is the no of digit on right side of the dot

Content

- VP Resource is the type of VP hold the content (VP_N16 by default)
- VP Address is the VP that hold the value
- First Icon is the first IMG_ICO of a series of icons
to be use as "number font"

Advance

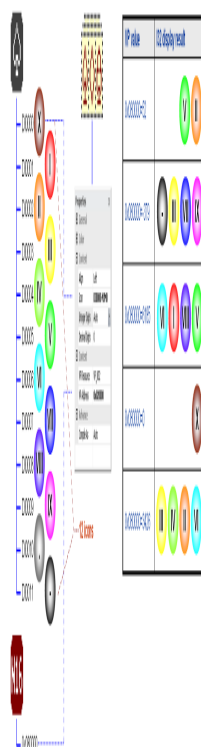
- Compile As can force VP variable work different than default (Auto)
VP_N16 (two successive VP_N16) work as VP_N32
VP_N32 (first two byte of VP_N32) work as VP_N16
VP_N64 (first two byte of VP_N64) work as VP_N16
VP_N64 (first four byte of VP_N64) work as VP_N32
- Location pointer is reserved, keep none for normal operation

Parameter modification (advance feature)

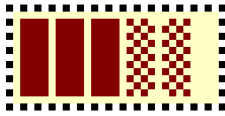
- Enable VP is none by default (element operate normally)
Select a VP for real time element control.
VP value will set to 1 as enable at power-on.
It can be disabled (hidden) by a zero value
- ICON VP is none by default (using the above selected font)
Select a VP for real time change the first IMG_ICO ID
of a series of icons to be use
VP value will copy the above ID at power-on.
- Transparent color VP is none by default (using the above selected color)
Select a VP for real time transparent color change
VP value will copy the above selected color at power-on.
- Transparent VP is none by default (using the above option)
Select a VP for real time transparent background control
VP value will copy the above transparent config at power-on.
It can be enable by a non-zero value or disable by a zero

note:

*1. Decimal Icon Operation Idea



4.4.14 Progress Bar (B16)



Properties	
General	
Name	Progress Bar
ID	0
X	12
Y	307
Width	60
Height	60
Style	
Mode	Color Mode
Direction	L->R
Forecolor1	00ff00
Forecolor2	00ff00
Transparent	False
Gap Transparent C...	ffffff
Gap Width	5
Step Width	10
Content	
VP Resource	VP_N16
VP Address	None
Icon	None
Min Value	1
Max Value	100
Real Time Properties Control	
Enable VP	None
Forecolor1 VP	None
Forecolor2 VP	None
ICON VP	None
Transparent Color VP	None
Transparent VP	None

General

- Show the basic information of the element
- X and Y is the location of element on screen (top-left corner of the screen = 0,0)
element top-left corner is the reference point
- Width and Height are in pixels

Style

- **Mode**
 - Color Mode: longer color bar could be show with a no. close to max (bar color could also be a mix of Forecolor1 & 2)
 - Icon Mode: Icon will move to the far end with a no. close to max
 - Fill Icon Mode: bar color could be replaced by an IMG_ICO
- **Direction**: L→R / R→L / Up→Dn / Dn→Up
- **Forecolor1** Progress start color
- **Forecolor2** Progress end color
- **Transparent** = False by default
True: enable transparent effect to show the IMG_ICO
- **Gap Transparent Color** function depends on mode selected
In Fill Color Mode, the selected color is gap color in the Progress Bar
In Icon Mode with Transparent Enabled, the selected color in the IMG_ICO will show as transparent. (it is suggested to use pure color for transparent effect) (e.g. pure-magenta pure-black, pure-white, etc...)
- **Gap Width** is in pixel for gap in the bar
- **Step Width** is in pixel for the distance between gap

Content

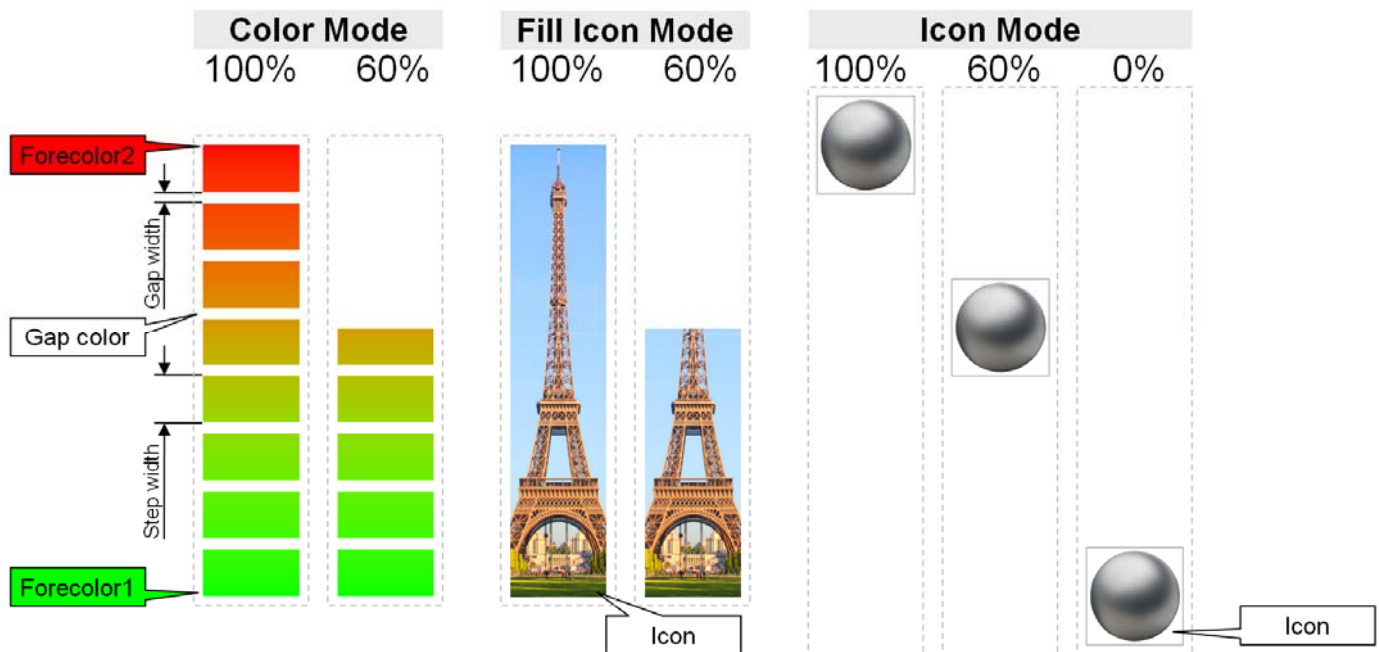
- **VP Resource** is the type of VP hold the content (VP_N16 by default)
- **VP Address** is the VP that hold the value
- **Icon** for Move Icon mode or Fill Icon mode
- **Min Value** is the minimum Progress Bar value
- **Max Value** is the maximum Progress Bar value

Parameter modification (advance feature)

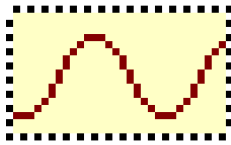
- **Enable VP** is none by default (element operate normally)
Select a VP for real time element control.
VP value will set to 1 as enable at power-on.
It can be disabled (hidden) by a zero value
- **Foreground color1 VP** is none by default (using the above selected color)
Select a VP for real time color change
VP value will copy the above selected color at power-on.
- **Foreground color2 VP** is none by default (using the above selected color)
Select a VP for real time color change
VP value will copy the above selected color at power-on.
- **Transparent color VP** is none by default (using the above selected color)
Select a VP for real time transparent color change
VP value will copy the above selected color at power-on.
- **Transparent VP** is none by default (using the above option)
Select a VP for real time transparent background control
VP value will copy the above transparent config at power-on.
It can be enable by a non-zero value or disable by a zero
- **ICON VP** is none by default (using the above selected font)
Select a VP for real time change the IMG_ICO ID
VP value will copy the above ID at power-on.

note:

*1. Progress Bar Operating Mode (Dn→Up) Idea



4.4.15 Graph Element (G16)



Properties	
General	
Name	Graph
ID	0
X	86
Y	307
Width	60
Height	60
Style	
Foreground Color	000000
Format	
Dot Width	1 Pixel
Dot Height	1 Pixel
Plot Type	Dot
Content	
VP Graph	None
Min Value	0
Max Value	32767
Real Time Properties Control	
Enable VP	None
Foreground Color VP	None
Ymin VP	None
Ymax VP	None
Ruler1 VP	None
Ruler2 VP	None

General

- Show the basic information of the element
- X and Y is the location of element on screen (top-left corner of the screen = 0,0)
element top-left corner is the reference point
- Width and Height are in pixels

Style

- Foreground Color is the color for of graph

Format

- Dot Width could also affect the X direction scale
1-4pixel could be selected
- Dot Height could also affect the X direction scale
1-4pixel could be selected
- Plot Type could be
Dot, only plot the dot on the screen
Line, joint the plotted Dot with lines
Area, fill the area between 0 and the plotted dot
Top Fill, fill the area between Max Value and the plotted dot
Bottom Fill, fill the area between Min Value and the plotted dot

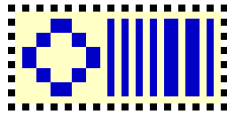
Content

- VP Graph select a VP_G16 array to be port on screen
- Min Value is minimal graph potting value
- Max Value is maximum graph potting value

Parameter modification (advance feature)

- Enable VP is none by default (element operate normally)
Select a VP for real time element control.
VP value will set to 1 as enable at power-on.
It can be disabled (hidden) by a zero value
- Foreground color VP is none by default (using the above selected color)
Select a VP for real time color change
VP value will copy the above selected color at power-on.
- Ymin VP is none by default (using the above selected color)
Select a VP for real time graph Min Value for Y scale zoom or pan
VP value will copy the above selected color at power-on.
- Ymax VP is none by default (using the above selected color)
Select a VP for real time graph Max Value for Y scale zoom or pan
VP value will copy the above selected color at power-on.
- Rule1 VP is none by default
Select a VP for real time horizontal line to be show in graph
- Rule2 VP is none by default
Select a VP for another real time horizontal line to be show in graph

4.4.16 Bitmap Element (BP1)



Properties	
General	
Name	Bitmap
ID	0
X	160
Y	307
Width	60
Height	60
Style	
Foreground Color(1)	000000
Background Color(0)	ffffff
Show Type	Show All
Content	
VP Bitmap	None
Real Time Properties Control	
Enable VP	None
Foreground Color VP	None
Background Color VP	None
Show Type VP	None

General

- Show the basic information of the element
- X and Y is the location of element on screen (top-left corner of the screen = 0,0)
element top-left corner is the reference point
- Width and Height are in pixels

Style

- Foreground Color is the color for "1" bits
- Background Color is the color for "0" bits
- Show Type are having three options
Show all (default) show all Foreground and Background Color
Show Foreground only show the "1" bits' color
Show Background only show the "0" bits' color

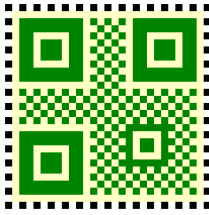
Content

- VP Bitmap select a VP_BP1 array for the 1bpp graph data

Parameter modification (advance feature)

- Enable VP is none by default (element operate normally)
Select a VP for real time element control.
VP value will set to 1 as enable at power-on.
It can be disabled (hidden) by a zero value
- Foreground color VP is none by default (using the above selected color)
Select a VP for real time color change
VP value will copy the above selected color at power-on.
- Background color VP is none by default (using the above selected color)
Select a VP for real time color change
VP value will copy the above selected color at power-on.
- Transparent VP is none by default (using the Show Type)
Select a VP for real time Show Type change
0: Show all (default) show all Foreground and Background Color
1: Show Foreground only show the "1" bits' color
2: Show Background only show the "0" bits' color
VP value will copy the above selected Show Type at power-on.

4.4.17 QR Code Element (QRC)



Properties	
General	
Name	QR Code
ID	0
X	234
Y	307
Width	82
Height	79
Style	
Scale	4
Format	
Size	45x45 (154 Byte)
Content	
VP Resource	VP_N16
VP Address	None
Real Time Properties Control	
Enable VP	None

General

- Show the basic information of the element
- X and Y is the location of element on screen (top-left corner of the screen = 0,0)
element top-left corner is the reference point
- Width and Height are in pixels

Style

- Scale could adjust the size of the QR code display on the screen

Format

- Size is the QR coding container size

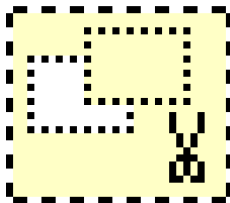
Content

- VP Resource is the type of VP hold the text content
VP_N16 by default, VP_STR is suggested for simple operation
- VP Address is the VP that hold the value

Parameter modification (advance feature)

- Enable VP is none by default (element operate normally)
Select a VP for real time element control.
VP value will set to 1 as enable at power-on.
It can be disabled (hidden) by a zero value

4.4.18 Draw Pad (DPD)



Properties	
General	
Name	Draw Pad
ID	0
X	383
Y	307
Width	60
Height	60
Content	
VP Resource	VP_N16
VP Address	None
Real Time Properties Control	
Enable VP	None

General

- Show the basic information of the element
- X and Y is the location of element on screen (top-left corner of the screen = 0,0)
element top-left corner is the reference point
- Width and Height are in pixels

Content

- VP Resource is the type of VP hold the text content
VP_N16 by default, VP_STR is suggested for simple operation
- VP Address is the start address of VPs that hold a series of draw command

Parameter modification (advance feature)

- Enable VP is none by default (element operate normally)
Select a VP for real time element control.
VP value will set to 1 as enable at power-on.
It can be disabled (hidden) by a zero value

Note.

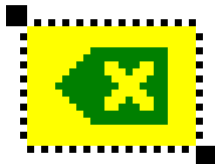
*1. see the Draw Pad in-memory command Functions section for details

4.4.19 Predefined Touch Key (ESC)



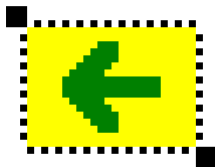
- PIP keyboard's Element
- For building PIP keyboard only
- Similar as Touch Key (TPK)
- Predefined as Call Key = "ESC"

4.4.20 Predefined Touch Key (DEL)



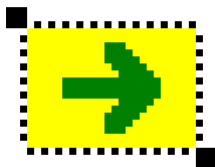
- PIP keyboard's Element
- For building PIP keyboard only
- Similar as Touch Key (TPK)
- Predefined as Call Key = "DelLastChar(VP)"

4.4.21 Predefined Touch Key (Left)



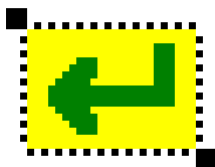
- PIP keyboard Element
- For building PIP keyboard only
- Similar as Touch Key (TPK)
- Predefined as Call Key = "Move Cursor Left"

4.4.22 Predefined Touch Key (Right)



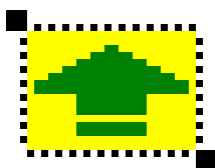
- PIP keyboard's Element
- For building PIP keyboard only
- Similar as Touch Key (TPK)
- Predefined as Call Key = "Move Cursor Right"

4.4.23 Predefined Touch Key (ESC)



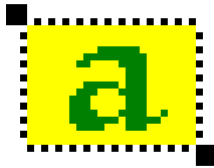
- PIP keyboard's Element
- For building PIP keyboard only
- Similar as Touch Key (TPK)
- Predefined as Call Key = "Enter"

4.4.24 Predefined Touch Key (CapLock)



- PIP keyboard's Element
- For building PIP keyboard only
- Similar as Touch Key (TPK)
- Predefined as Call Key = "CapLock"

4.4.25 Predefined Touch Key (char)

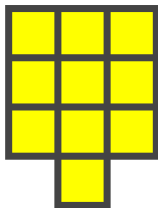


- PIP keyboard's Element
- For building PIP keyboard only
- Similar as Touch Key (TPK)
- Predefined as Call Key = "Con(Buff, Cap/Norm(Byte0/Byte1"
- "Value" should be assigned for the key input

Example:

Value	CapLock input	Normal input	Descriptions
0x4161	A	a	Value(Hi) = 0x41 ="A"; Value(Lo) = 0x61 ="a"
0x4262	B	b	Value(Hi) = 0x42 ="B"; Value(Lo) = 0x61 ="b"
0x4363	C	c	Value(Hi) = 0x43 ="C"; Value(Lo) = 0x61 ="c"
0x2131	!	1	Value(Hi) = 0x21 ="!"; Value(Lo) = 0x31 ="1"
0x4032	@	2	Value(Hi) = 0x40 ="@"; Value(Lo) = 0x32 ="2"
0x2333	#	3	Value(Hi) = 0x23 ="#"; Value(Lo) = 0x33 ="3"
"4"	Invalid	4	Value(Hi) = 0x00 = is not an valid value; Value(Lo) = 0x34 ="4" Suitable for the number keyboard without CapLock
"5"	Invalid	5	Value(Hi) = 0x00 = is not an valid value; Value(Lo) = 0x34 ="4" Suitable for the number keyboard without CapLock
"6"	Invalid	6	Value(Hi) = 0x00 = is not an valid value; Value(Lo) = 0x34 ="4" Suitable for the number keyboard without CapLock
"P"	Invalid	P	Value(Hi) = 0x00 = is not an valid value; Value(Lo) = 0x50 ="P" Suitable for the keyboard without CapLock
"Q"	Invalid	Q	Value(Hi) = 0x00 = is not an valid value; Value(Lo) = 0x51 ="Q" Suitable for the keyboard without CapLock
"R"	Invalid	R	Value(Hi) = 0x00 = is not an valid value; Value(Lo) = 0x52 ="R" Suitable for the keyboard without CapLock

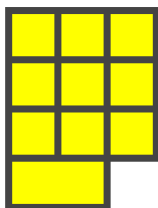
4.4.26 Predefined Touch Key - Set (number keyboard 1)



- PIP keyboard's Element
- For building PIP keyboard only
- Similar as Touch Key (TPK)
- generate a set of keys (telephone-like) for quick PIP keyboard generation
- totally generate 4x4 → 16keys
- user may design a background art work before overlaying the Key - Set

1	2	3	ESC
4	5	6	←
7	8	9	-
Clear	0	.	ENTER

4.4.27 Predefined Touch Key - Set (number keyboard 2)



- PIP keyboard's Element
- For building PIP keyboard only
- Similar as Touch Key (TPK)
- generate a set of keys (PC-KB-like) for quick PIP keyboard generation
- totally generate 4x4 → 15keys
- user may design a background art work before overlaying the Key - Set

7	8	9	ESC
4	5	6	←
1	2	3	-
0	.	ENTER	

4.4.28 Predefined Touch Key - Set (English keyboard)



- PIP keyboard's Element
- For building PIP keyboard only
- Similar as Touch Key (TPK)
- generate a set of keys (QWERTY) for quick PIP keyboard generation
- totally generate 54keys
- user may design a background art work before overlaying the Key - Set

~	-	+	"	{	}	←	→	Close	
!	@	#	\$	%	^	&	*	()
1	2	3	4	5	6	7	8	9	0
Q	W	E	R	T	Y	U	I	O	P
A	S	D	F	G	H	J	K	L	DEL
 \ 	Z	X	C	V	B	N	M	? /	
Cap	<	>	Space				:	OK	

Page Configurations

4.4.29 Page Properties



Properties	
General	
Name	PG0000
ID	PG0000
Width	800
Height	480
Style	
Color	<input type="text" value="ffffff"/>
Background I...	None
Auto Jump	
Target	None
Delay(sec)	0

General

- Show the basic information of the element
- X and Y is the location of element on screen (top-left corner of the screen = 0,0)
element top-left corner is the reference point
- Width and Height are in pixels

Style

- Color select a color for page background
- Background Image select a Background Image

Auto Jump

- Target select a target page to jump to
- Delay(sec) the delay before page jump

4.4.30 Page Functions

Page Functions	
Functions List	
ID	Function Name
0	Function0
1	Function1
2	Function2
3	Function3
4	Function4
5	Function5
6	Function6
7	Function7
<input type="button" value="Add"/> <input type="button" value="Delete"/>	
Function Properties	
Property	Value
ID	0
Call	VP := Value
VP Address	0x020004
Value	99
Input Length	59
Min Value	-2147483648
Max Value	2147483647
Compile As	Auto
<input type="button" value="Close"/>	

Page functions are a series of VP operations that could carry out at the moment a page going to be display.
e.g. assign a value to a VP

Every page have its own Page Functions,

Right click on the empty space of a Page and select Page functions could bring up the Page Functions window.

User may add several functions to any of the page.

Note.***1. Operation functions that could be called in Page Functions**

VP:= VP XOR Value
 BUFF:= VP
 VP:= BUFF
 VP:= Value
 VP:= VP + Value
 VP:= VP - Value
 VP:= VP * Value
 VP:= VP/Value

Byte0 (VP) := Byte0 (Value)
 Byte1 (VP) := Byte0 (Value)
 Byte2 (VP) := Byte0 (Value)
 Byte3 (VP) := Byte0 (Value)
 VP:=Concatenate (VP, Value)
 VP:= DellastChar (VP)

Bit0 (VP) := LSB (Value)
 Bit1 (VP) := LSB (Value)
 Bit2 (VP) := LSB (Value)
 Bit3 (VP) := LSB (Value)
 Bit4 (VP) := LSB (Value)
 Bit5 (VP) := LSB (Value)
 Bit6 (VP) := LSB (Value)
 Bit7 (VP) := LSB (Value)

4.5 Elements Sub Functions Details

4.5.1 Draw Pad (DPD) in-memory command Functions

- Draw Pad element is an element that execute a drawing function that store inside VP_N16 which point by its properties VP_address.
- Host could send the Function Pack into VP by using Multi_Write (0x82) or N16_Write (0x3D) command.
- it is possible to adjust the function or parameter by VPK, TPK or Page_Call.
- And it will refresh/update the Draw Pad content accordingly.
- Each Draw Pad Element could only execute ONE command multiple times.
- The coordinate of the drawing content is based on the page coordinate.
- The Draw Pad element area is the valid area that shows the content.

Draw Pad Element Function Pack Structure

Add	Content	Descriptions
VP	CMD	Command function
VP+2	Number_of_Parmeter_SET	Number of parameter set (*2)
VP+4	Parameter_set	Parameter sets for command (*1)

Draw Pad Element Functions

CMD	Name	Parameter Set			Descriptions
		Relative Add Seq	Size in byte	Definitions	
0x0001	Draw_dot	0x00	4	xh,xl,yh,yl	Dot coordinate (*1)
		0x04	2	Color	Dot color
0x0002	Draw_successive_line	0x00	2	Color	Line color (*3)
		0x02	4	x0h,x0l,y0h,y0l	Coordinate of the line starting point (*1)
		0x06	4	X1h,x1l,y1h,y1l	Line to this coordinate
		:	4	:	:
		0x02+4*n	4	xnh,xnl,ynh,ynl	Last coordinate for line
0x0003	Draw_line	0x00	4	xsh,xsl,ysh,ysl	Line start coordinate (*1)
		0x04	4	xeh,xel,yeh,yel	Line end coordinate
		0x08	2	Color	Line color
0x0004	Draw_rectangle	0x00	4	xsh,xsl,ysh,ysl	Top-left corner coordinate (*1)
		0x04	4	xeh,xel,yeh,yel	Bottom-right corner coordinate
		0x08	2	Color	Rectangle line color
0x0005	Fill_rectangle	0x00	4	xsh,xsl,ysh,ysl	Top-left corner coordinate (*1)
		0x04	4	xeh,xel,yeh,yel	Bottom-right corner coordinate
		0x08	2	Color	Rectangle fill color
0x0006	Show_crop_Page_BKG	0x00	2	PAGE_IDh,PAGE_IDl	PAGE's background image (NOT IMG_BKG) to be crop and show (*1)
		0x02	4	xsh,xsl,ysh,ysl	Top-left corner coordinate for crop
		0x06	4	xeh,xel,yeh,yel	Bottom-right corner coordinate for crop
		0x0a	4	xh,xl,yh,yl	Top-left corner coordinate for showing the cropped image on screen
0x0007	Show_IMG_ICO	0x00	4	xh,xl,yh,yl	Top-left corner coordinate for showing the cropped image on screen (*1)
		0x04	2	IMG_ICO_ID	IMG_ICON to be show
0x0008	Show_text	0x00	4	xh,xl,yh,yl	Top-left corner coordinate for showing the text on screen (*1)
		0x04	2	Color	Text color
		0x06(0x06H)	1	FONT_ID	Font ID
		0x07(0x06L)	1	Text_len	Text length in no. of byte
		0x08	n	Text_STRING	Text content

Note.

*1. First byte of this parameter could "control" the functionality

- 0xFF terminate the following draw function
- 0xFE terminate current parameter_set, jump to next parameter_set

*2. Draw_successive_line need n set for (n-1) lines

*3. Color is excluded from Number_of_Parmeter_SET

4.5.2 TPK, VPK Call Functions

Call functions are functions which could be called up by TPK or VPK.

4.5.2.1 Call Functions - Keyboards

Function	Descriptions
Keyboard Number	Number Keyboard for number value input and report the value to the host. (*1) The input value will be stored inside the predefined VP Variable
Keyboard Password	Password Keyboard for number password input. (display as *, 9char max.) and report the input value to the host. (*1) The input value will be stored inside the predefined VP Variable
Keyboard English	English Keyboard for English input and report the input value to the host. (*1) The input value will be stored inside the predefined VP Variable
Keyboard Chinese	Chinese Pinyin Keyboard for number value input and report the value to the host. (*1) The input value will be stored inside the predefined VP Variable
PIP Menu	Page-in-page Menu is a customizable menu. It pop-up a little window of another page as a menu selection input and report the input value to the host. (*1) The input value will be stored inside the predefined VP Variable
PIP Keyboard	Page-in-page Keyboard is a customizable keyboard. It could pop up a little window of another page as a keyboard keys and report the input value to the host. (*1) The input value will be stored inside the predefined VP Variable.
PIP Number Keyboard	Page-in-page Number Keyboard is a customizable keyboard. It pop up a little window of another page as a keyboard keys and report the input value to the host. (*1) The input value will be stored inside the predefined VP Variable.
PIP RTC	Page-in-page RTC is a customizable interface to adjust the Real Time Clock time. It could pop up a little window of another page as a keyboard keys.
Adjust Backlight	Backlight brightness level setting.
Adjust RTC	Real Time Clock time adjust interface.
Slider Single	Single slider input interface input is corresponding ratio of the predefined max. min. value and report the input value to the host. (*1) The input value will be stored inside the predefined VP Variable.
Slider Dual	Dual slider input interface function as Single Slider with two VP Variable input, Where, the two slider will limit each other for non-overlapped value input and report the input value to the host. (*1) The two input value will be stored inside two successive predefined VP Variable.
Listing	VP_STR listing window Display the VP_STR content with predefined VP_STR Address and no. of VP_STR to be display. Providing touch and scroll interface.
Listing with Slider	VP_STR listing window Display the VP_STR content with predefined VP_STR Address and no. of VP_STR to be display. Providing slider bar for scroll.

Note: *1. Entered value that report to host with command header 0x77

4.5.2.2 Call Functions - PIP Keyboard's keys

Function	Descriptions
Enter	Enter key for PIP keyboard for finishing and confirming the input.
Esc	Escape key for PIP keyboard/menu. It could cancel the input and close the PIP keyboard/menu.
CapsLock	CapsLock key function for PIP Keyboard
VP:= Value, Enter	Directly assigning the value into VP variable and follow with a Enter to confirm the input. Mainly for PIP menu items selection
Buff:=Con(Buff,Cap, Nom(Byte0/Byte1))	Generally key call for PIP keyboard with (Capslock feature) It will put the lower byte value into the end of Buff area. If the CapsLock toggled, the high byte will be insert into the end of the Buff area (note: its value in properties should be a 16bit formatted value)
VP:=Concatenate(VP,Value)	Put its properties value to the end of the content of the predefined VP value Mainly work with BUFF as a general char input E.g. VP(BUFF)="123", Properties' Value = 4, after call VP(BUFF)="1234"
VP:= DelLastChar(VP)	Delete the last byte of the the VP variable Mainly for VP(BUFF) as a backspace functon. E.g. VP(BUFF)="123", after call VP(BUFF)="12"

4.5.2.3 Call Functions - Operations

Function	Descriptions
VP:= VP XOR Value	Predefine VP variable XOR with a constant stay in Value, and store the result back into the VP variable E.g. VP_N32(0x020000)=0x0055, properties' Value= 0x00AA; result VP_N32 (0x020000)=0x00FF
BUFF:= VP	Copy the VP variable content to the BUFF(string format) location. E.g. VP_N32(0x020000)=123; result VP_STR(BUFF)="123" (note: number converted to string)
VP:= BUFF	Copy the BUFF content to a VP variable (integer or a string) E.g. VP(BUFF)="123", result VP_N32(0x020000)=123 (note: string converted to interger)
VP:= Value	Copy a constant stay in Value to a VP variable E.g. properties' Value= 123; result VP_N32(0x020000)=123
VP:= VP + Value	Predefine VP variable add with a constant stay in Value, and store the result back into the VP variable E.g. VP_N32(0x020000)=666, Value=2; result VP_N32 (0x020000)=668
VP:= VP - Value	Predefine VP variable minus by a constant stay in Value, and store the result back into the VP variable E.g. VP_N32(0x020000)=666, Value=10; result VP_N32 (0x020000)=656
VP:= VP + Value, loop	Predefine VP variable times by a constant stay in Value, and store the result back into the VP variable If the value is bigger than the predefined maximum, it will "loop" to the predefined minimum. E.g. VP_N32(0x020000)=666, Value=2; result VP_N32 (0x020000)=668
VP:= VP - Value, loop	Predefine VP variable minus by a constant stay in Value, and store the result back into the VP variable If the value is smaller than the predefined minimum, it will "loop" to the predefined maximum. E.g. VP_N32(0x020000)=666, Value=10; result VP_N32 (0x020000)=656
VP:= VP * Value	Predefine VP variable times by a constant stay in Value, and store the result back into the VP variable Eg: VP_N32(0x020000)=666, Value=10; result VP_N32 (0x020000)=6660
VP:= VP/Value	Predefine VP variable divided by a constant stay in Value, and store the result back into the VP variable E.g. VP_N32(0x020000)=666, Value=2; result VP_N32 (0x020000)=333
Byte0(VP):= Byte0(Value)	Copy the constant byte0(lowest byte) stay in Value to VP variable byte0(Lowest byte) location E.g. VP_N32(0x020000)=0x5555, Value=0x0A; result VP_N32 (0x020000)=0x550A
Byte1(VP):= Byte0(Value)	Copy the constant byte0(lowest byte) stay in Value to VP variable byte1 location E.g. VP_N32(0x020000)=0x5555, Value=0x0A; result VP_N32 (0x020000)=0x0A55
Byte2(VP):= Byte0(Value)	Copy the constant byte0(lowest byte) stay in Value to VP variable byte2 location E.g. VP_N32(0x020000)=0x5555, Value=0x0A; result VP_N32 (0x020000)=0x0A5555
Byte3(VP):= Byte0(Value)	Copy the constant byte0(lowest byte) stay in Value to VP variable byte3(Highest byte) location E.g. VP_N32(0x020000)=0x5555, Value=0x0A; result VP_N32 (0x020000)=0x0A005555
Bit0(VP):= LSB(Value)	Copy the constant LSB stay in Value to VP variable bit0(LSB) location E.g. VP_N32(0x020000)=0x0000, Value=0x01; result VP_N32 (0x020000)=0x0001
Bit1(VP):= LSB(Value)	Copy the constante LSB stay in Value to VP variable bit1 location Eg: VP_N32(0x020000)=0x0000, Value=0x01; result VP_N32 (0x020000)=0x0002
Bit2(VP):= LSB(Value)	Copy the constant LSB stay in Value to VP variable bit2 location E.g. VP_N32(0x020000)=0x0000, Value=0x01; result VP_N32 (0x020000)=0x0004
Bit3(VP):= LSB(Value)	Copy the constant LSB stay in Value to VP variable bit3 location E.g. VP_N32(0x020000)=0x0000, Value=0x01; result VP_N32 (0x020000)=0x0008
Bit4(VP):= LSB(Value)	Copy the constant LSB stay in Value to VP variable bit4 location E.g. VP_N32(0x020000)=0x0000, Value=0x01; result VP_N32 (0x020000)=0x0010
Bit5(VP):= LSB(Value)	Copy the constant LSB stay in Value to VP variable bit5 location E.g. VP_N32(0x020000)=0x0000, Value= 0x01; result VP_N32 (0x020000)=0x0020
Bit6(VP):= LSB(Value)	Copy the constant LSB stay in Value to VP variable bit6 location E.g. VP_N32(0x020000)=0x0000, Value= 0x01; result VP_N32 (0x020000)=0x0040
Bit7(VP):= LSB(Value)	Copy the constant LSB stay in Value to VP variable bit7 location E.g. VP_N32(0x020000)=0x0000, Value= 0x01; result VP_N32 (0x020000)=0x0080

5 Serial Communication

Smart TFT LCD Module serial command is for real-time control and access. Host machine get the data which input through the Smart TFT LCD Module interface or provide the data for display.

5.1.1 Hardware connection

Smart TFT LCD Modules serial UART interface are base on RS232-C standard by default config as 8N1 115200bps.

5.1.2 Communication Packet Structure

Commands and Response Packet should be format as follow (host→module):

Seq	Code	Code type	Description
1	0xAA	Packet header	1byte
2	Cmd-code	Command code	1byte
3	Par-data	Parameter or Data	(*1)
:	:	-	-
:	:	-	-
:	:	-	-
N-3 th	0xCC	Packet tail	4byte
N-2 th	0x33		
N-1 th	0xC3		
N th	0x3C		

Note.

*1. Unless otherwise specified,
all the multi-byte values, data, address' byte sequence are MSB first, LSB last.
e.g. 0x12345678 should send as following sequency, 0x12(first), 0x34, 0x56, 0x78(last)

5.1.3 Packet Acknowledgment

Packet Acknowledgment is two byte in ASCII (module → host):

Response	code	Description
Command (in packet) executed and wait for next Command	":>"	In ASCII (0x3a, 0x3e)
Command (in packet) error and wait for next Command	"!>"	In ASCII (0x21,0x3e)
Invalid Packet	null	No response

Note.

*1. Packet Acknowledgement response to a valid packet only.

*2. Acknowledgement could be disable in RGTools option (Manual → Tools → Project Setting → RS232 → Enable ACK)

5.1.4 Color Data Value Configuration

16 bit color value															
R4	R3	R2	R1	R0	G5	G4	G3	G2	G1	G0	B4	B3	B2	B1	B0
High byte (MSB)								Low byte (LSB)							
D7	D6	D5	D4	D3	D2	D1	D0	D7	D6	D5	D4	D3	D2	D1	D0

5.1.5 Data / Address / Page_ID / Location Values Configuration

64 bit number value							
D63...D56	D55...D48	D47...D40	D39...D32	D31...D24	D23...D16	D15...D8	D7...D0
Byte7 (MSB)				Byte0 (LSB)			
D7...D0	D7...D0	D7...D0	D7...D0	D7...D0	D7...D0	D7...D0	D7...D0

32 bit number value			
D31...D24	D23...D16	D15...D8	D7...D0
Byte3 (MSB)		Byte0 (LSB)	
D7...D0	D7...D0	D7...D0	D7...D0

16 bit number value	
D15...D8	D7...D0
High Byte (MSB)	
D7...D0	D7...D0

5.2 Command Summary

Functions	Name	Code	Description
Config/ Status Functions	hand_shake	0x30	Read a Hand Shake
	read_ver	0x31	Read firmware version
	read_pg_id	0x32	Read Current page ID
	touch_response	0x72/0x73/0x77 0x78/0x79	see also set_sys_config
	set_sys_config	0xE0	System parameter configuration and Baud Rate
	sel_project	0xE1	Specify operating project folder
	touch_calib	0xE4	Touch panel calibration
	screen_saver	0x5E	Screen saver (backlight dim down time out)
	backlight_ctrl	0x5F	backlight brightness control (64 levels)
	buzzer_touch_sound	0x79	buzzer enable time length (in 10ms step)
	Buzzer_ctrl	0x7A	Buzzer control
	Flash_write	0x90	Write data to the flash
	Flash_read	0x91	Read data from the flash
	RTC_set	0x9C	Set the RTC
	RTC_read	0x9B	Read the RTC values
	USR_bin_read	0x93	Read data from the USR_bin
	u_drv_format	0xE2	Format the u_drive
	u_drv_unlock	0xE3	Unlock the u_drive with pre-stored password
Display Control/ Draw Functions	disp_page	0x70	Display a pre-stored TML file (page)
	set_element_fg	0x7E	Set the foreground color of STR, N16, N32 or N64
	set_element_bg	0x7F	Set the background color of STR, N16, N32 or N64
	set_codepage	0xE7	Sets country character set and code-page character set
	suspend_vp_fresh	0xE8	Set the screen to pause the refresh and deactivate the touchkey or release the pause to refresh and enable the touchkey
VP Functions	Successive_write	0x82	Write successive value to VP_N16, VP_N32, VP_N64
	Successive_read	0x83	Read successive value from VP_N16, VP_N32, VP_N64
	STR_write	0x42	Write string to VP_STR
	STR_read	0x43	Read string from VP_STR
	STR_fill	0x46	Fill strings to the VP_STR
	N16_write	0x3D	Write 16bit (signed integer) value to VP_N16
	N16_read	0x3E	Read 16bit (signed integer) value from VP_N16
	N16_fill	0x3F	Fill numbers to the VP_N16
	N32_write	0x44	Write 32bit (signed integer) value to VP_N32
	N32_read	0x45	Read 32bit (signed integer) value from VP_N32
	N32_fill	0x47	Fill numbers to the VP_N32
	N64_write	0x48	Write 64bit (signed integer) value to VP_N64
	N64_read	0x49	Read 64bit (signed integer) value from VP_N64
	N64_fill	0x4A	Fill numbers to the VP_N64
	Successive_write	0x82	Write successive value to VP_N16, VP_N32, VP_N64
	Successive_read	0x83	Read successive value from VP_N16, VP_N32, VP_N64
	BP1_write	0x4B	Write bit-map (1bpp) data to VP_BP1
	BP1_write_comp	0x4C	Write compressed bit-map (1bpp) data to VP_BP1
	G16_write	0x4D	Write 16bit (signed integer) graphic array to VP_G16
	G16_write_rotate	0x4E	Rotate the VP_G16 array data inside the module and write a 16bit (signed integer) value into end-of-array
	System Register (Timer_Ctrl)Write	0x3B	Write System Register
	System Register (Timer_Ctrl)Read	0x3C	Read System Register

5.3 Command Details

5.3.1 Config / Status Functions

	Header	Command	Data:no_of_byte	Tail
hand_shake (0x30)	AA	30	--	CC 33 C3 3C
	e.g. [host]: AA 30 CC 33 C3 3C e.g. [LCM]: AA 30 54 6F 70 77 61 79 20 48 4D 54 20 52 65 61 64 79 00 CC 33 C3 3C (return "Topway HMT Ready" in ASCII)			
Read_ver (0x31)	AA	31	--	CC 33 C3 3C
	e.g. [host]: AA 31 CC 33 C3 3C e.g. [LCM]: AA 31 31 2E 31 32 CC 33 C3 3C (return "1.12" in ASCII)			
read_pg_id (0x32)	AA	32	Nil or ID:2	CC 33 C3 3C
	e.g. [host]: AA 32 CC 33 C3 3C e.g. [LCM]: AA 32 00 01 CC 33 C3 3C (current page's ID is 0x0001)			
touch_response (0x72/0x73)	AA	72/73	X:2 Y:2	CC 33 C3 3C
	e.g. [LCM]: AA 72 00 32 00 64 CC 33 C3 3C (touch release coordinate is(50,100)) e.g. [LCM]: AA 73 00 32 00 64 CC 33 C3 3C (touch down coordinate is(50,100))			
data_entered (0x77)	AA	77	Addr_ID:4 Data:2/4/8/n	CC 33 C3 3C
	e.g. [LCM]: AA 77 00 02 00 00 00 00 00 32 CC 33 C3 3C (Addr_ID 0x020000's 32bit data is 0x0000 0032) (please refer to RGTools description about Addr_ID and Data size)			
touch_response (0x78/79)	AA	78/79	Page_ID:2 TPK_ID:1	CC 33 C3 3C
	e.g. [LCM]: AA 78 00 02 01 CC 33 C3 3C (PAGE:0x0002 TPK:0x01 key up) e.g. [LCM]: AA 79 00 02 01 CC 33 C3 3C (PAGE:0x0002 TPK:0x01 key down)			
set_sys_config (0xE0)	AA	E0 55 AA 5A A5	BaudRate:1 sys_par1:1 sys_par2:1	CC 33 C3 3C
	e.g. [host]: AA E0 55 AA 5A A5 07 83 00 CC 33 C3 3C (config baud rate to 152000, and enable TPK response (the most common setting)) (Baud rate will back to project default setting after next power on)			
Set_project (0xE1)	AA	E1	Prj_ID:1	CC 33 C3 3C
	e.g. [host]: AA E1 01 CC 33 C3 3C (switch to project 0x01, (THMT01 folder))			
Touch_calib (0xE4)	AA	E4 55 AA 5A A5	--	CC 33 C3 3C
	e.g. [host]: AA E4 55 AA 5A A5 CC 33 C3 3C (start touch calibration)			
screen_saver (0x5E)	AA	5E	Time:2 Level:1	CC 33 C3 3C
	e.g. [host]: AA 5E 00 0A 00 CC 33 C3 3C (after 10sec of idel, backlight down to 0)			
Backlight_ctrl (0x5F)	AA	5F	Level:1	CC 33 C3 3C
	e.g. [host]: AA 5F 3F CC 33 C3 3C (adjust the backlight to highest level 0x3f)			
Buzzer_touch_sound (0x79)	AA	79	Time(in10ms):1	CC 33 C3 3C
	e.g. [host] : AA 79 01 CC 33 C3 3C (adjust the buzzer beeping duration to 0x01= 1x10ms)			
Buzzer_ctrl (0x7A)	AA	7A	Loops T1 T2 Freq1 Freq2	CC 33 C3 3C
	e.g. [host] : AA 7A 10 0A 08 05 32 CC 33 C3 3C (sound the buzzer beeping 16loops of 1sec 500Hz → 0.8sec 5kHz)			

	Header	Command	Data:(no. of byte)	Tail
Flash_write (0x90)	AA	90	ADDR:4 Len:2 Data:n	CC 33 C3 3C
	e.g. [host]: AA 90 00 00 00 00 00 02 30 31 CC 33 C3 3C (start form user flash area address 0x0000000 write 2byte of data 0x30 & 0x31) (note. it is a must to ensure the power supply during the write process)			
Flash_read (0x91)	AA	91	ADDR:4 Len:2 or Data:n	CC 33 C3 3C
	e.g. [host]: AA 91 00 00 00 00 00 02 CC 33 C3 3C (read 2byte start form address 0x00000000) e.g. [LCM]: AA 91 30 31 CC 33 C3 3C (return of two byte data 0x30 0x31)			
RTC_set (0x9C)	AA	9C	YYMMDDHHMMSS:6	CC 33 C3 3C
	e.g. [host]: AA 9C 0E 07 0A 17 3B 30 CC 33 C3 3C (set the real time clock to 2014 - 07- 10 23:59:48)			
RTC_read (0x9B)	AA	9B	-- or YYMMDDHHMMSS:6	CC 33 C3 3C
	e.g. [host]: AA 9B CC 33 C3 3C (ask for full data time) e.g. [LCM]: AA 9B 0E 07 0B 00 01 12 CC 33 C3 3C (return the day and time 2014-07-11 00:01:18)			
Usr_bin_read (0x93)	AA	93	ADDR:4 Len:2 or Data:n	CC 33 C3 3C
	e.g. [host]: AA 93 00 00 00 00 00 02 CC 33 C3 3C (request for 2 byte of data form the preloaded usr.bin file at address 0x0000 0000) e.g. [LCM]: AA 93 30 31 CC 33 C3 3C (return 2byte of requested data 0x30 0x31)			
U_drv_format (0xE2)	AA	E2	55 AA 5A A5	CC 33 C3 3C
	e.g. [host]: AA E2 55 AA 5A A5 CC 33 C3 3C (format the whole internal flash memory space)			
U_drv_unlock (0xE3)	AA	E3	PWD:n	CC 33 C3 3C
	e.g. [host]: AA E3 30 31 32 33 34 35 36 00 CC 33 C3 3C (send a password "0123456" to un-lock the USB interface for file access Note: password defined in the display project by using RGTools, where password string should end with "\0")			

note. Please refer to the Smart_LCD user manual for the details

5.3.2 Display Control / Draw Functions

	Header	Command	Data:no_of_byte	Tail
Disp_page (0x70)	AA	70	Page_id:2	CC 33 C3 3C
	e.g. [host]: AA 70 00 01 CC 33 C3 3C (show page 0x0001)			
Set_element_fg (0x7E)	AA	7E	Element_type:1 Page_ID:2 Element_ID:1 color:2 0x00	CC 33 C3 3C
	e.g. [host]: AA 7E 00 00 03 05 FF FF 00 CC 33 C3 3C (change the STR_ID=0x05's foreground to white in Page_ID=0x0003)			
Set_element_bg (0x7F)	AA	7F	Element:1 Page_ID:2 Element_ID:1 color:2 mode:1	CC 33 C3 3C
	e.g. [host]: AA 7F 00 00 03 05 00 00 00 CC 33 C3 3C (change the STR_ID=0x05's foreground to (non-transparent) black in Page_ID=0x0003)			
Set_codepage (0xE7)	AA	E7	Country Codepage	CC 33 C3 3C
	e.g. [host]: AA E5 03 07 CC 33 C3 3C switch the ASCII to German, Extended Codepage as 866 (OEM - Russian)			
Suspend_vp_refresh (0xE8)	AA	E8	55 AA 5A A5 mode	CC 33 C3 3C
	e.g. [host]: AA E8 55 AA 5A A5 01 CC 33 C3 3C (pause the refresh and deactivate the touch key; mode=00 for release the pause)			

note. please refer to the Smart_LCD user manual for the details

5.3.3 VP Functions

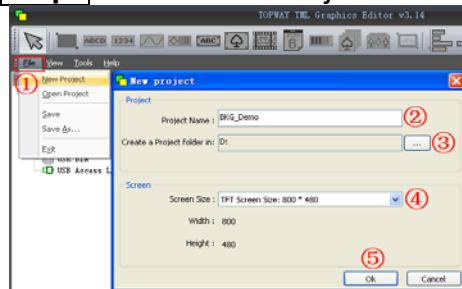
	Header	Command	Data:no_of_byte	Tail
Successive_write (0x82)	AA	82	Add:4 Len:1 Value:n	CC 33 C3 3C
	e.g. [host]: AA 82 00 08 00 00 03 00 00 11 11 22 22 CC 33 C3 3C Write 3 successive 16bit value to VP_N16 address 0x0008 0000			
	e.g. [host]: AA 82 00 02 00 00 02 55 55 55 55 66 66 66 66 CC 33 C3 3C Write 2 successive 32bit value to VP_N32 address 0x0002 0000			
Successive_read (0x83)	AA	83	Add:4 Len:1 or Value:n	CC 33 C3 3C
	e.g. [host]: AA 83 00 08 00 00 03 CC 33 C3 3C Read 3 successive 16bit value from VP_N16 address start at 0x0008 0000			
	e.g. [host]: AA 83 00 02 00 00 02 CC 33 C3 3C Read 2 successive 32bit value from VP_N32 address start at 0x0002 0000			
STR_write (0x42)	AA	42	Add:4 Text:n	CC 33 C3 3C
	e.g. [host]: AA 42 00 00 00 80 54 4F 50 57 41 59 00 CC 33 C3 3C (write string "TOPWAY" to VP_STR address 0x0000 0080 (string should end with "\0"))			
STR_read (0x43)	AA	43	Add:4 or Text:n	CC 33 C3 3C
	e.g. [host]: AA 43 00 00 00 80 CC 33 C3 3C (request for VP_STR address 0x0000 0080 content value)			
	e.g. [LCM]: AA 43 54 4F 50 57 41 59 00 CC 33 C3 3C (the feedback content value is "TOPWAY" (string end with "\0"))			
STR_fill (0x46)	AA	46	Add:4 Len:2 Text:n	CC 33 C3 3C
	e.g. [host]: AA 46 00 00 00 80 00 03 54 4F 50 57 41 59 00 CC 33 C3 3C (fill 0x0003 VP_STR as "TOPWAY" (string should end with "\0")) Start from the VP_STR address 0x00000080)			
N16_write (0x3D)	AA	3D	Add:4 Value:2	CC 33 C3 3C
	e.g. [host]: AA 3D 00 08 00 00 00 32 CC 33 C3 3C (write a 16bit value 0x0032 into VP_N16 address 0x0008 0000)			
N16_read (0x3E)	AA	3E	Add:4 or Value:2	CC 33 C3 3C
	e.g. [host]: AA 3E 00 08 00 00 CC 33 C3 3C (read VP_N16 address 0x0008 0000's 16bit data content)			
	e.g. [LCM]: AA 3E 00 32 CC 33 C3 3C (feed back the VP_N16 content value 0x0032)			
N16_fill (0x3F)	AA	3F	Add:4 Len:2 Value:2	CC 33 C3 3C
	e.g. [host]: AA 3F 00 08 00 00 00 03 00 32 CC 33 C3 3C (fill 0x0003 VP_N16 with 0x0032 start from VP_N16 address 0x0008 0000)			
N32_write (0x44)	AA	44	Add:4 Value:4	CC 33 C3 3C
	e.g. [host]: AA 44 00 02 00 00 00 00 00 32 CC 33 C3 3C (write a 32bit value 0x0000 0032 into VP_N32 address 0x0002 0000)			
N32_read (0x45)	AA	45	Add:4 or Value:4	CC 33 C3 3C
	e.g. [host]: AA 45 00 02 00 00 CC 33 C3 3C (read VP_N32 address 0x0002 0000's 32bit data content)			
	e.g. [LCM]: AA 45 00 00 00 32 CC 33 C3 3C (feed back the VP_N32 content value 0x0000 0032)			
N32_fill (0x47)	AA	47	Add:4 Len:2 Value:4	CC 33 C3 3C
	e.g. [host]: AA 47 00 02 00 00 00 03 00 00 00 32 CC 33 C3 3C (fill 0x0003 VP_N32 with 0x0000 0032 start from VP_N32 address 0x0002 0000)			

	Header	Command	Data:no_of_byte	Tail
N64_write (0x48)	AA	48	Add:4 Value:8	CC 33 C3 3C
	e.g. [host]: AA 48 00 03 00 00 00 00 00 00 00 00 32 CC 33 C3 3C (write a 64bit value 0x0000 0000 0000 0032 into VP_N64 address 0x0003 0000)			
N64_read (0x49)	AA	49	Add:4 or Value:8	CC 33 C3 3C
	e.g. [host]: AA 49 00 03 00 00 CC 33 C3 3C (read VP_N64 address 0x0003 0000's 64bit data content) e.g. [LCM]: AA 49 00 00 00 00 00 00 00 32 CC 33 C3 3C (feed back the VP_N64 content value 0x0000 0000 0000 0032)			
N64_fill (0x4A)	AA	4A	Add:4 Len:2 Value:8	CC 33 C3 3C
	e.g. [host]: AA 4A 00 03 00 00 00 03 00 00 00 00 00 00 32 CC 33 C3 3C (fill 0x0003 VP_N64 with 0x0000 0000 0000 0032 start form VP_N64 address 0x0003 0000)			
BP1_write (0x4B)	AA	4B	Add:4 Len:4	CC 33 C3 3C
	e.g. [host]: AA 4B 00 04 00 00 00 00 02 00 CC 33 C3 3C + 512byte data... (fill 512 byte of graphics data (1bpp) in to BP1 address 0x0004 0000)			
BP1_write_comp (0x4C)	AA	4C	Add:4 Len:4	CC 33 C3 3C
	e.g. [host]: AA 4C 00 04 00 00 00 00 02 00 CC 33 C3 3C +512byte data... (fill 512 byte of compressed graphics data (1bpp) in to BP1 address 0x0004 0000)			
G16_write (0x4D)	AA	4D	Add:4 Len:2 Value:2n	CC 33 C3 3C
	e.g. [host]: AA 4D 00 06 00 00 00 02 00 32 00 33 CC 33 C3 3C (fill 0x0002 16bit graph data(0x0032, 0x0033) into G16 address 0x0006 0000)			
G16_write_rotate (0x4E)	AA	4E	Add:4 Size:2 Data:2	CC 33 C3 3C
	e.g. [host]: AA 4E 00 06 00 00 00 32 00 33 CC 33 C3 3C (place rotate 0x0032 of G16 forward which address 0x0006 0000 and place 0x0033 at the last)			
System_register_write (timer_ctrl) (0x3B)	AA	3B	Add:4 Data:1	CC 33 C3 3C
	e.g. [host]: AA 3B 00 FF FF 00 01 CC 33 C3 3C (start the timer_ctrl0 as count down mode)			
System_register_read (timer_ctrl) (0x3C)	AA	3C	Add:4	CC 33 C3 3C
	e.g. [host]: AA 3C 00 FF FF 00 CC 33 C3 3C (read the timer_ctrl status value)			
note. Please also refer to the Smart LCD user manual for the details				

6 Basic Function Examples

6.1 Display a PAGE with background image (IMG_BKG)

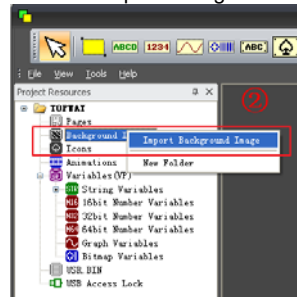
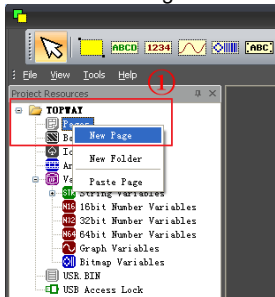
Step1 Start a New Project



- ① start TOPWAY RGTools, start a new project through manual bar File --> New Project
- ② Enter the Project name
Project Name: BKG_Demo (as an example)
- ③ Select a Project folder location
Create a Project Folder in: D:
- ④ Select the display resolution of the SmartLCD
Screen Size: 800x480 (for this example HMT050CC-C)
- ⑤ Click "OK"

Step2 Build a Page and import a picture as Background Image

- ① In Resource window, right click on Pages, select New Page
- ② In Resource window, right click on Background Images, select Import Background Image

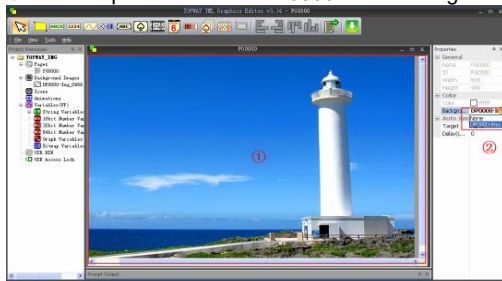


Note: RGTools supports BMP, JPG, PNG. In this case a 800x480 picture could be best fit to the display. or RGTools will resize it to fit the display



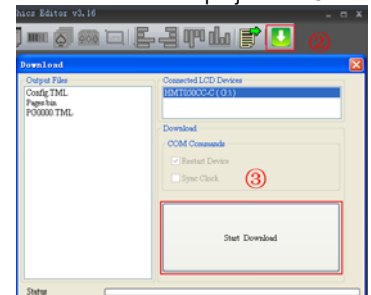
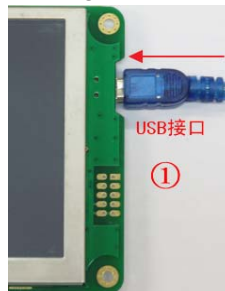
Step3 Link the page with image

- ① Click on the working area of the page, its properties will show on the right.
- ② In Properties window DP0000 in the Background Image.



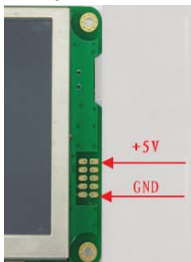
Step4 Compile and download

- ① Connect the Smart TFT LCD Module with PC by using an mini USB cable
- ② In tool bar, click on download(or F9)
- ③ click "Start Download" to download the project to LCM



Step5 Power on and display

- ① disconnect the mini USB cable
- ② power on the Smart TFT LCD Module with 5V
- ③ display



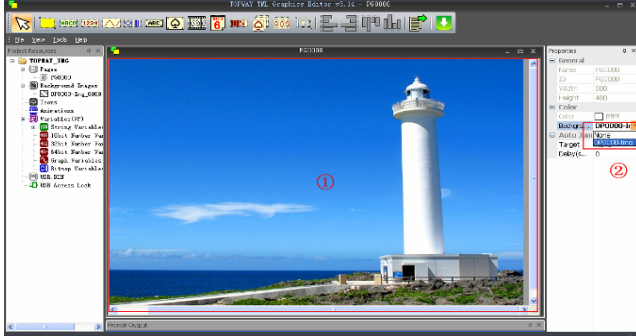
-- done --

6.2 Using Touch Key (TPK) to Jump to a Page

Step1 Start a New Project
(please refer to the previous examples)

Step3 Link the page (PG0000) with the image

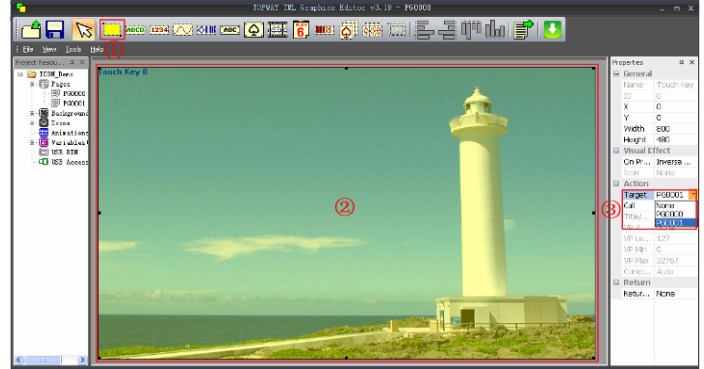
- ① Click on the working area of the page, its properties will show on the right.
- ② In Properties window, select DP0000 in the Background Image.



Step2 Built 2 pages, import 2 pictures as background
(please refer to the previous examples)

Step4 Built a Touch Key (TPK)

- ① In tools bar, select Touch Key
- ② create a rectangle touch key area on PG0000
- ③ In touch key properties, set Target as PG0001



Step5 Link the page (PG0001) with the image

- ① Click on the working area of PG0001, its properties will show on the right.
- ② In Properties window, select DP0001 in the Background Image.

Step6 Compile and download
(please refer to the previous examples)

Step7 Power on and display

- ① disconnect the mini USB cable
- ② power on the Smart TFT LCD Module with 5V
- ③ display
- ④ touch on the touch key area
- ⑤ when release, the display will jump to the target page



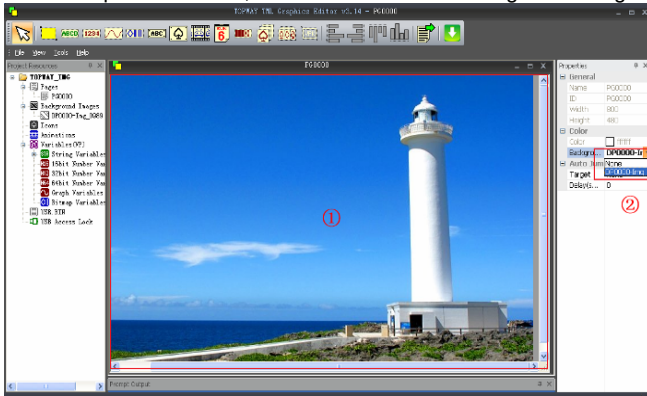
-- done --

6.3 Show a Static Icon (ICO)

Step1 Start a New Project
(please refer to the previous examples)

Step3 Link the page with the image

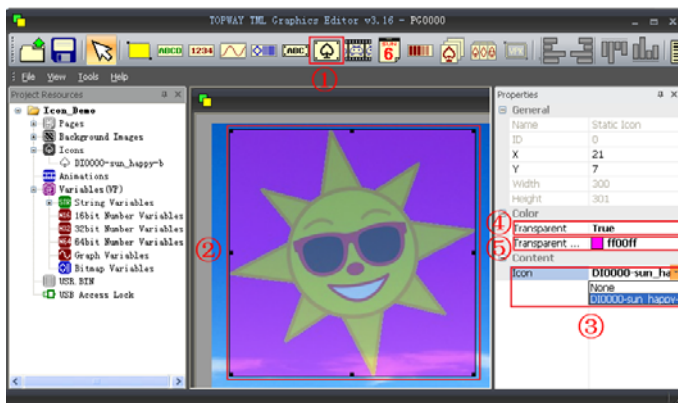
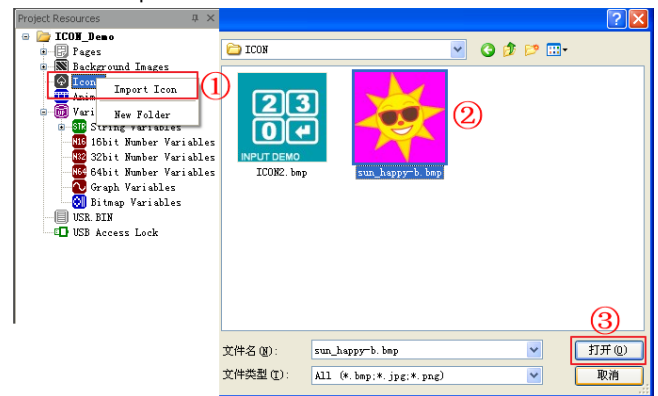
- ③ Click on the working area of the page, its properties will show on the right.
- ④ In Properties window, select DP0000 in the Background Image.



Step2 Built a page, import a pictures as background
(please refer to the previous examples)

Step4 Import an icon

- ① In Resources window, right click Icons select Import Icon.
- ② Select a icon file
- ③ select "Open"



Step5 Built a static icon on the page

- ① In tool bar, select static icon.
- ② Create a rectangle static icon area on page
- ③ In static icon properties, select the imported icon DI0000
- ④ In static icon properties, set Transparent as "True"
- ⑤ In static icon properties, set Transparent Color as "0xff00ff"

Step6 Compile and download
(please refer to the previous examples)

Step7 Power on and display
(please refer to the previous examples)



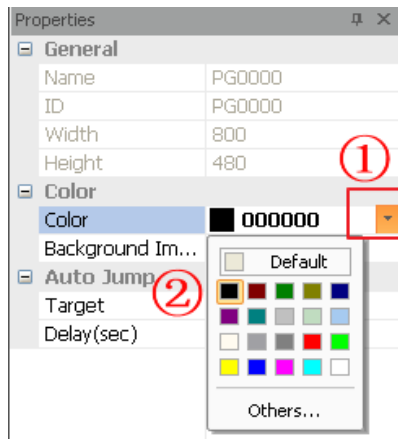
-- done --

6.4 Show a Static String (STS)

Step1 Start a new project and built a page
(please refer to the previous examples)

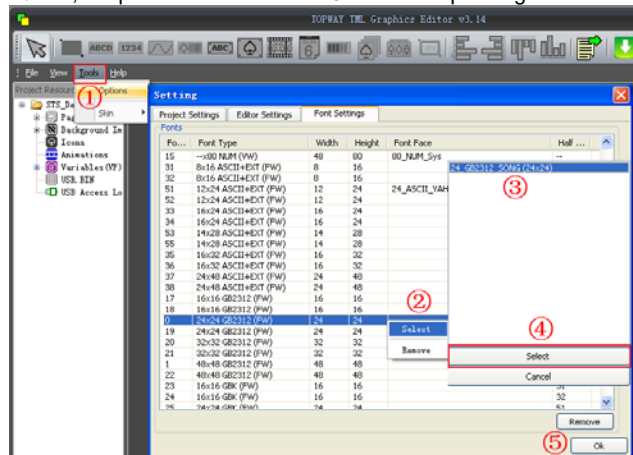
Step2 Select a background color for PAGE

- ① In PAGE properties, click Color pull down menu
- ② Select black (0x000000) as background color



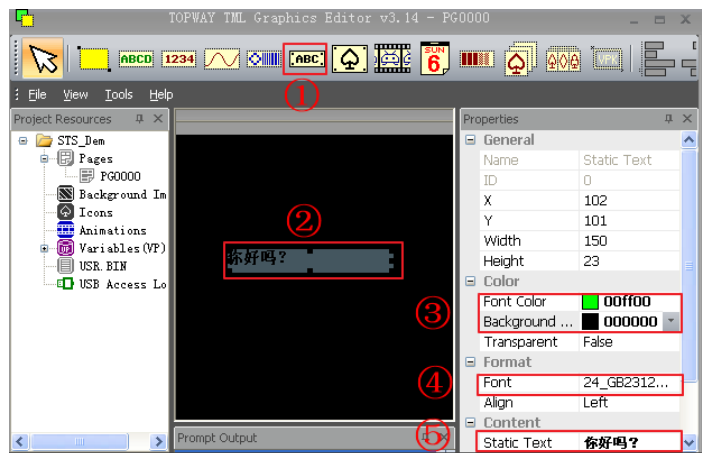
Step3 Import a Font Face

- ① In menu bar, Tools-->Options-->Font Setting.
 - ② Right click on font 0 then "Select"
 - ③ Select "24_GB2312_SONG(24X24)" (a full width char font)
 - ④ Click "Select"
 - ⑤ Click "OK"
- (note, repeat the same on font 51 for corresponding half width char font)



Step4 Built a Static String on page

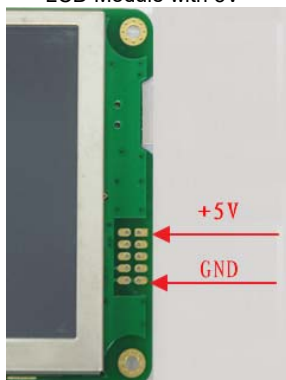
- ① In tools bar, select Static String
- ② Create a rectangle static string area on page
- ③ In static string properties, select font color
- ④ select Font as 24_GB2312_SONG(24X24)
- ⑤ Input Static Text: "你好吗?"



Step5 Compile and download
(please refer to the previous examples)

Step6 Power on and display
(please refer to the previous examples)

- ① disconnect the mini USB
- ② display cable
- ③ power on the Smart TFT LCD Module with 5V



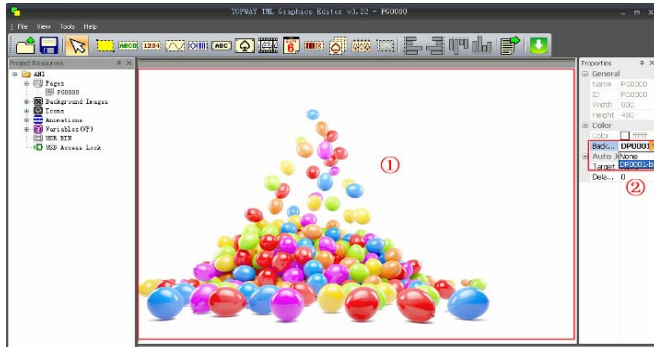
-- done --

6.5 Show an Animation Element (ANI)

Step1 Start a New Project
(please refer to the previous examples)

Step3 Link the page with the image

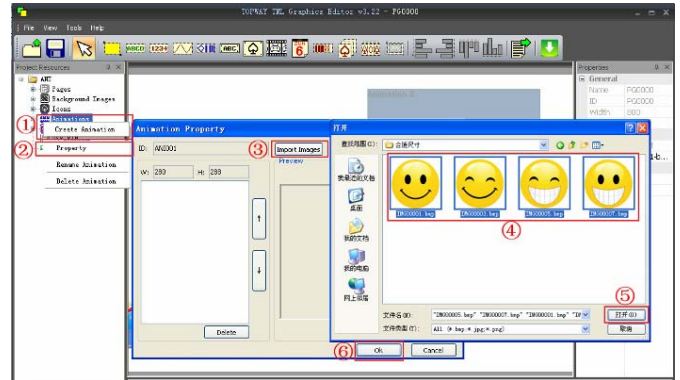
- ① Click on the working area of the page, its properties will show on the right.
- ② In Properties window, select DP0000 in the Background Image.



Step2 Built a page, import a pictures as background
(please refer to the previous examples)

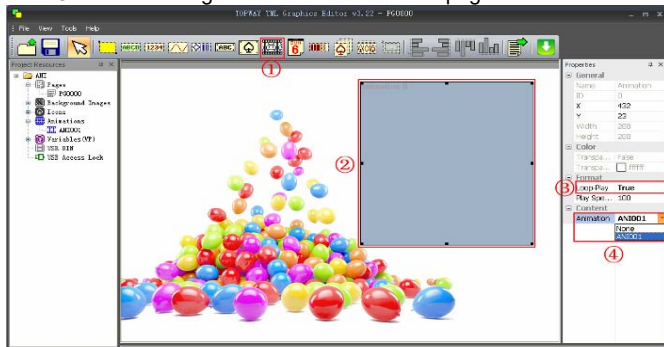
Step4 Import pictures as Animation

- ① In resources window, right click Animations, select "Create Animation"
- ② In resources window, right click on ANI001 select "Property"
- ③ In Animation Property's window click "Import Images"
- ④ select a series of animation-pictures (in same size)
- ⑤ click "OPEN"
- ⑥ click "OK"



Step5 Built a Animation Element

- ① In tool bar, select animation element
- ② Create a rectangle animation element on page



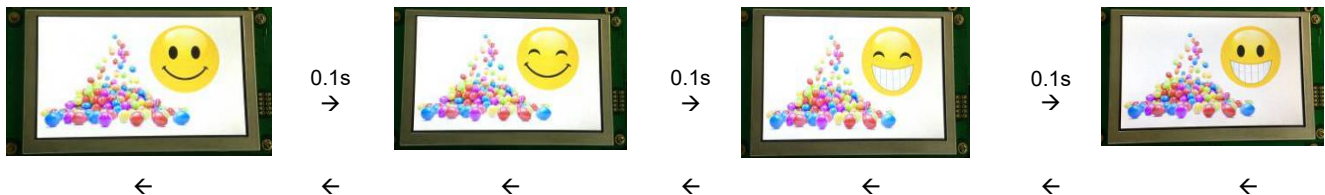
Step6 Link the Animation with the Element

- ③ In Animation Element properties, set Loop-Play as True
- ④ In Animation Element properties, select the Animation import in step4

Step7 Compile and download
(please refer to the previous examples)

Step8 Power on and display
(please refer to the previous examples)

- ① disconnect the mini USB cable
- ② power on the Smart TFT LCD Module with 5V
- ③ display



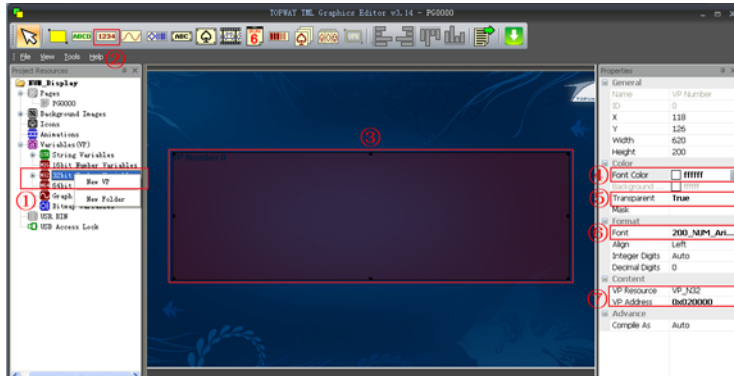
-- done --

7 VP_Variables and Keyboard Examples

7.1 Using Internal Number Keyboard to input a value to a VP_N32

Step1 Start a New Project
(please refer to the previous examples)

Step3 Link the page with the IMG_BKG



Step2 Built a page, import a pictures as IMG_BKG
(please refer to the previous examples)

Step4 Import a Font Face

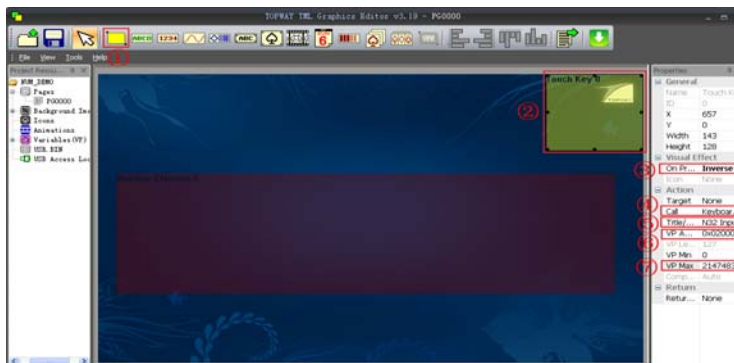
- ① In menu bar, Tools-->Options-->Font Setting.
- ② Right click on font 50 then "Select"
- ③ Select "120x200NUM" font face
- ④ Click "Select"
- ⑤ Click "OK"

Step5 Allocate a VP_N32 variable

- ① In Resources window, right click on "32bit Number Variables" select : "New VP"

Step6 Built a Number Element and link with VP

- ① In tools bar, select number element
- ② Create a rectangle Number element area on page
- ③ In Number Element properties, set "Font color" as "0xFFFFFFFF"
- ④ set "Transparent" as "True"
- ⑤ set "font" as "Font 50"
- ⑥ set "VP Address" link to "VP_N32 0x020000"



Step7 Built a Touch Key (TPK) to call out Number Keyboard

- ① In tools bar, select Touch Key
- ② Create a rectangle touch key area on the PAGE
- ③ In touch key properties, set "On Press Down" as "Inverse Color"
- ④ set CALL as "Keyboard Number"
- ⑤ set "Title/Value" as "N32 Input"
- ⑥ set "VP Address" as "0x020000"
- ⑦ set "VP Max" as "2147483647"

Step8 Compile and download
(please refer to the previous examples)

Step9 Power on and Display

- ① Disconnect the mini USB cable
- ② Power on the Smart TFT LCD Module with 5V
- ③ Touch the TPK, a number keyboard pop up.
- ④ Input a value and click "OK"



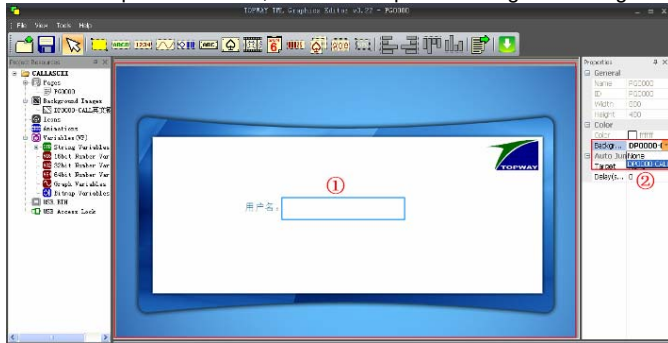
-- done --

7.2 Using Internal English Keyboard to input a string to a VP_STR

Step1 Start a New Project
(please refer to the previous examples)

Step3 Link the page with the IMG_BKG

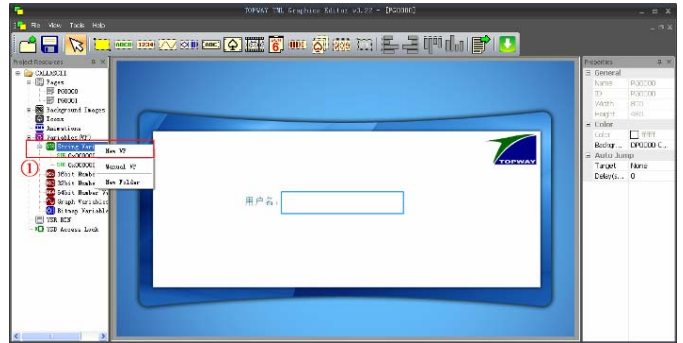
- ① Click on the working area of the PAGE, its properties will show on the right.
- ② In Properties window, select the imported Background Image.



Step2 Built a page, import a pictures as IMG_BKG
(please refer to the previous examples)

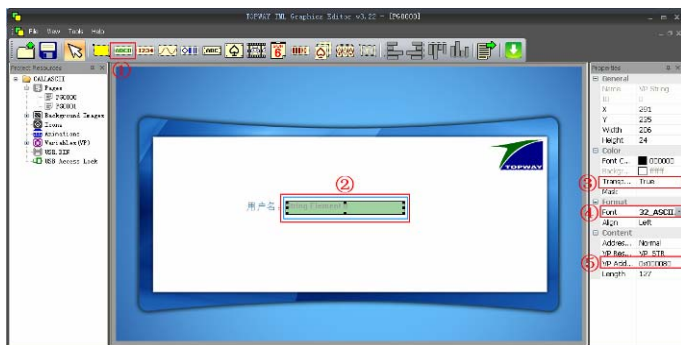
Step4 Allocate a VP_STR

- ① In Resources window, right click on "String Variables" select "New VP"



Step5 Built a String Element and linke with VP

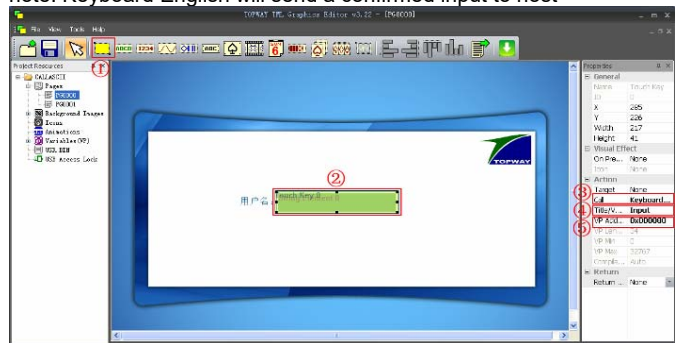
- ① In tools bar, select String Element
- ② Create a rectangle String Element area on PAGE
- ③ In String Element properties set Transparent as "True"
- ④ In String Element properties, set Font as: "32_ASCII_SysBold"
- ⑤ In String Element properties, set VP Address link to VP_STR 0x000080



Step6 Built a Touch Key (TPK) to call out English Keyboard

- ① In tools bar, select Touch Key
- ② Create rectangle touch key area on the PAGE
- ③ In touch key properties, set "CALL" as "Keyboard English"
- ④ set "Title/Value" as "Input"
- ⑤ set "VP Address" as 0x000080

note: Keyboard English will send a confirmed input to host

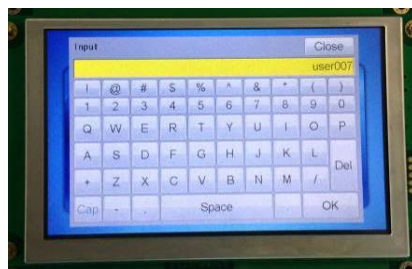
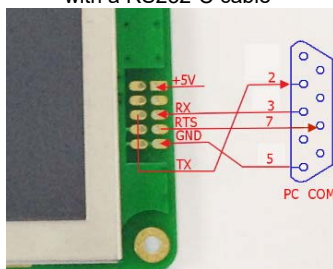


Step7 Compile and download (please refer to the previous examples)

Step8 Power on and Display

- ① Disconnect the mini USB cable
- ② Power the Smart TFT LCD Module with 5V
- ③ Connect to PC (115200,8,n,1) with a RS232-C cable
- ④ touch the TPK area
- ⑤ type in "user007"
- ⑥ click "OK"

- ⑦ Host PC could also see "user007" in packet with any serial terminal program.

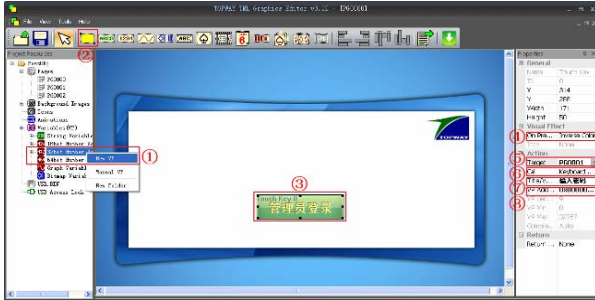


-- done --

7.3 Using Internal Password Keyboard to input a number pin

Step1 Start a New Project
(please refer to the previous examples)

Step2 Built 3 pages, import 3 pictures as IMG_BKG
(please refer to the previous examples)



Step6 Built a Virtual Touch Key (VPK) to trigger a page jump

- ① In tool bar, select Virtual Touch Key
- ② Create rectangle VPK area on the PAGE
- ③ In Touch Key properties, set "Monitor-VP" as "0x020000"
- ④ set "Monitor-Value" as "12345"(*1)
- ⑤ set "Target" as "PG0002"
- ⑥ Built a retry TPK on DP0001 (login failed) page for jump back to PG0000 (login page)

*1. If 0x020000 value equal to "12345", it jump to PG0002



Step3 Link the pages with the IMG_BKGs

- ① Click on the working area of each Page, the properties will show on the right.
- ② In Properties window, select the imported Background image where DP0000 for login, DP0001 for login failed, DP0002 for successfully login

Step4 Allocate a VP_N32

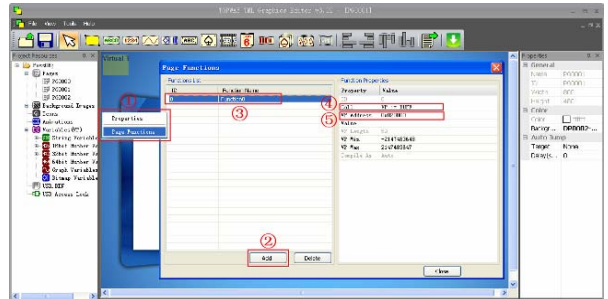
- ① In Resources Window, right click on 32bit Number Variables select "New VP"

Step5 Built a Touch Key (TPK) to call out Password Keyboard

- ① In tools bar, select Touch Key
- ② Create rectangle touch key area on the PAGE
- ③ In Touch Key properties, set "On Press Down" as "Inverse Color"
- ④ set "Target" as "PG0001"
- ⑤ set "Call" as "Keyboard password"
- ⑥ set "Title/Value" as "Input Password"
- ⑦ set "VP Address" as "0x000000 (BUFF)"

Step7 Page Function that action at Page loading (copy the BUFF value that input by password keyboard to the Monitor-VP)

- ① Right click on the empty area of DP0001 select "Page Functions"
- ② Click on "Add"
- ③ select the Functions List ID:0
- ④ set "Call" as "VP:=BUFF"
- ⑤ set "VP Address" "0x020000"
- ⑥ click "Close" to finish



Step8 Compile and download (please refer to the previous examples)

Step9 Power on and Display

- ① Disconnect the mini USB cable
- ② Power the Smart TFT LCD Module with 5V
- ③ touch the login key
- ④ input "12345"
- ⑤ click "ok"
- ⑥ for wrong password, click retry



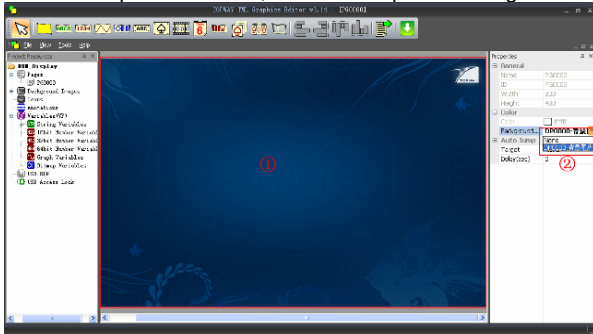
-- done --

7.4 Using Internal Real-Time-Clock Keyboard to set the Real-Time-Clock(RTC) time

Step1 Start a New Project
(please refer to the previous examples)

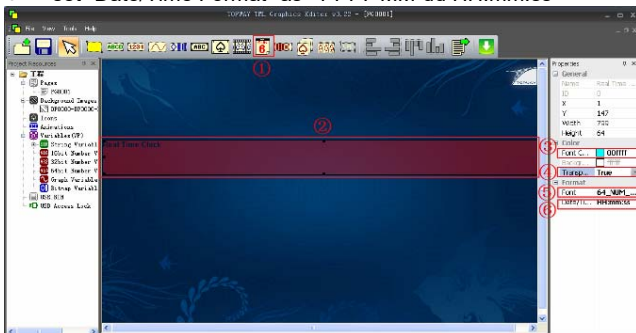
Step3 Link the page with the IMG_BKG

- Click on the working area of the PAGE, its properties will show on the right.
- In Properties window, select the imported Background Image.



Step5 Built a RTC on screen RTC

- In tools bar, select RTC
- Create a rectangle area on page
- In Touch Key properties, set "Font Color" as "0x00FFFF" (CYAN)
- set "Transparent" as "True"
- set "Font" as "64_NUM_SevenSegment(40x64)"
- set "Date/Time Format" as "YYYY-MM-dd HH:mm:ss"



Step7 Compile and download (please refer to the previous examples)

Step8 Power on and Display

- Disconnect the mini USB cable
- Power on the Smart TFT LCD Module with 5V
- Touch on the RTC area and pop up a RTC keyboard for time adjustment.
- Click "OK" then update the time



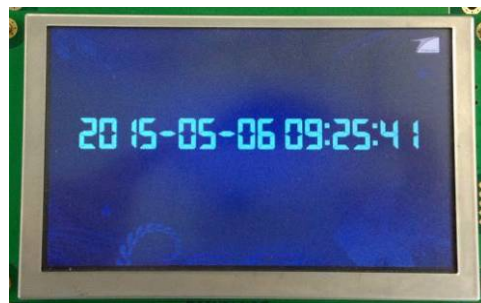
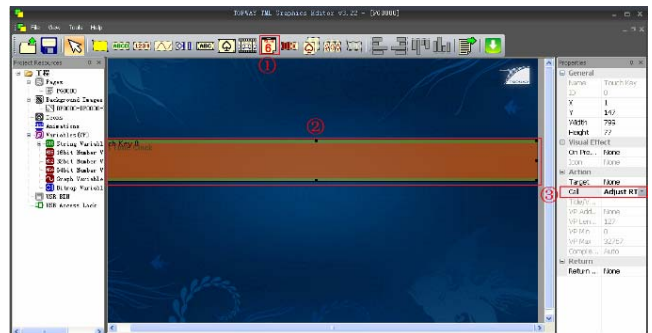
Step2 Built a page, import a pictures as IMG_BKG
(please refer to the previous examples)

Step4 Import a Font Face
(please refer to the previous examples)

- In menu bar, Tools-->Options-->Font Setting.
- Right click on font 44 then "Select"
- Select "64_NUM_SevenSegment(40x64)"
- Click "Select"
- Click "OK"

Step6 Built a Touch Key (TPK) to call RTC Keyboard

- In tool bar, select Touch Key
- Create a rectangle touch key area on the PAGE
- In touch key properties, set "Call" as "Adjust RTC"



-- done --

7.5 Using PIP Menu to input a String Value

Step1 Start a New Project

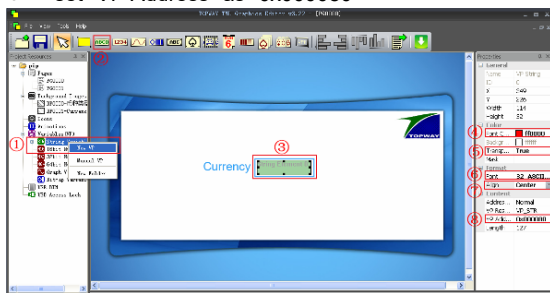
(please refer to the previous examples)

Step3 Link the page with the IMG_BKG

(please refer to the previous examples)

Step4 Allocate a VP_STR and show on page

- ① In Resources Window, right click on String Variables, select "New VP"
- ② In tools bar, select String Element
- ③ Create rectangle string element area on the PAGE
- ④ In String Element properties, set "Font Color" as "0xFF0000"
- ⑤ set "Transparent" as "True"
- ⑥ set "Font" as "32_ASCII_SysBold"
- ⑦ set "Align" as "Center"
- ⑧ set "VP Address" as "0x000080"

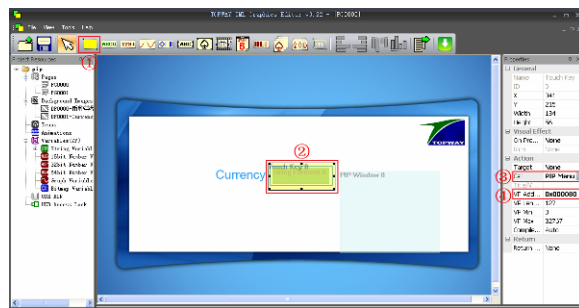


Step2 Built two pages, import a pictures as IMG_BKG (please refer to the previous examples)

- ① Click on the working area of the PAGE, its properties will show on the right.
- ② In Properties window, select the imported Background Image.
note: DP0000 is for main page; DP0001 for PIP menu page

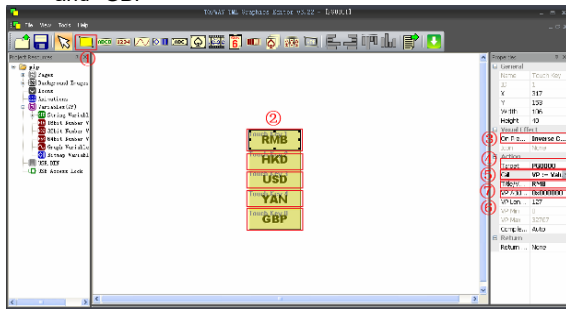
Step5 Built a Touch Key (TPK) to call PIP Menu

- ① In tools bar, select Touch Key
- ② Create rectangle touch key area on the PAGE
- ③ In Touch Key properties, set "Call" as "PIP Menu"
- ④ set "VP Address" as "0x000080"



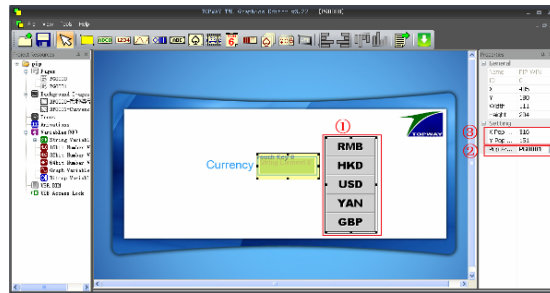
Step6 Built 5keys in PIP menu page (PG0001)

- ① In tools bar, select Touch Key
- ② Create 5 rectangle touch keys area on the PAGE
- ③ In Touch Key properties
set all 5 TPKs' "On Press Down" as "Inverse Color"
- ④ set all 5 TPKs' "Target" as "PG0000"
- ⑤ set all 5 TPKs' "Call" as "VP:=value"
- ⑥ set all 5 TPKs' "VP Address" as "0x000080"
- ⑦ set each TPK "Title/Value" as "RMB", "HKD", "USD", "YAN" and "GBP"



Step7 PIP Menu window configuration

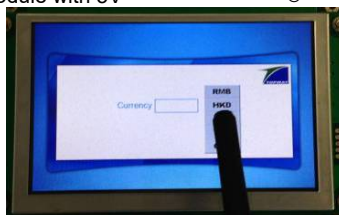
- ① click on PIP Menu window
- ② In PIP WIN properties,
set "Pop Area of Page" as "PG0001"(the menu page)
- ③ set "PIP WIN X/Y" pop up position
(hold the Ctrl key with mouse to shift the PIP window content area)
(After selected the TPK by mouse click, PIP window will show on page)



Step8 Compile and download (please refer to the previous examples)

Step9 Power on and Display

- ① Disconnect the mini USB cable
- ② Power on the Smart TFT LCD Module with 5V
- ③ Click on the input box, and select the PIP MENU
- ④ The corresponding string show screen



-- done --

7.6 Using PIP (Number) Keyboard to input a value

Step1 Start a New Project

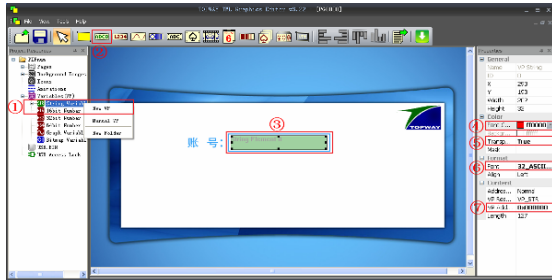
(please refer to the previous examples)

Step3 Link the page with the IMG_BKG

(please refer to the previous examples)

Step4 Allocate a String VP and show on page

- ① In Resources Window, right click on String Variable, select "New VP"
- ② In tools bar, select String Element
- ③ Create rectangle string element area on the PAGE
- ④ In String Element properties, set "Font Color" as "0xFF0000"
- ⑤ set "Transparent" as "True"
- ⑥ set "Font" as "32_ASCII_SysBold"
- ⑦ set "VP Address" as "0x000080"



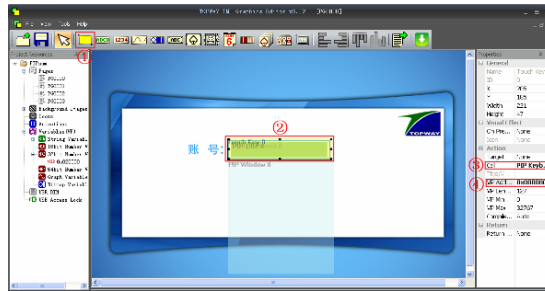
Step2 Built two pages, import a pictures as

IMG_BKG (please refer to the previous examples)

- ① Click on the working area of the PAGE, its properties will show on the right.
 - ② In Properties window, select the imported Background Image.
- note: DP0000 is for main page; DP0001 for PIP(number) Keyboard page, where DP0002 for press down effect only.

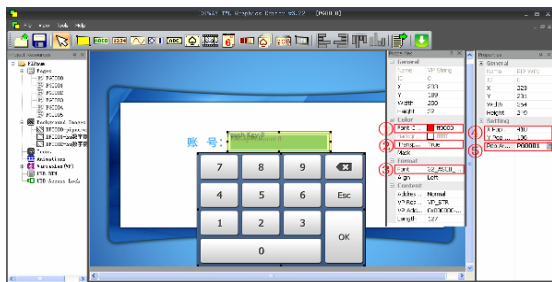
Step5 Built a Touch Key (TPK) to call PIP (Number) Keyboard

- ① In tools bar, select Touch Key
- ② Create rectangle touch key area on the PAGE
- ③ In Touch Key properties, set "Call" as "PIP Number Keyboard"
- ④ set "VP Address" as "0x000080"



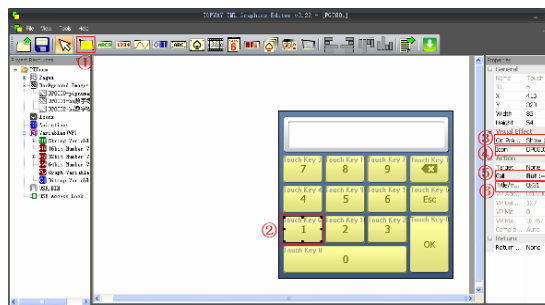
Step6 PIP Keyboard configuration

- ① In POP TITLE properties, set "Font Color" as "0xFF0000"
- ② set "Transparent" as "True"
- ③ set "Font" as "32_ASCII_SysBold"
- ④ In POP WIN properties, set "Pop Area of Page" as "PG0001" (keyboard page)
- ⑤ set "PIP WIN X/Y" pop up position (hold the Ctrl key with mouse to shift the POP window content area)



Step7 Built PIP keyboard's keys and its setting

- ① In tools bar, select Touch Key
- ② Create 13 rectangle touch keys area on the PAGE
- ③ In Touch Key properties, set all 13 TPKs' "On Press Down" as "Cropped BgImg"
- ④ set all 13 TPKs' "Icon" as DP0002
- ⑤ set the 10 numeric TPKs' "Call as "Buf:=Con(Buff,Cap/Nom(Byte0/Byte1))"
- ⑥ set "Title/Value" as its own value(*1)
- ⑦ set the "X" TPK's "Call" as "vp:=DelLastchar(vp)"
- ⑧ set the "ESC" TPK's "Call" as "Call=Esc"
- ⑨ set the "OK" TPK's "Call" as "Call=Enter"



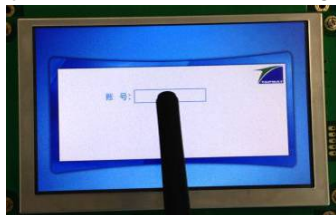
*1. Key and code value

Key	Code Value
0	0x30
1	0x31
2	0x32
:	:
8	0x38
9	0x39

Step8 Compile and download (please refer to the previous examples)

Step9 Power on and Display

- ① Disconnect the mini USB cable
- ② Power on the Smart TFT LCD Module with 5V
- ③ Click on the input box
- ④ enter "768986"click "OK"



-- done --

7.7 Using PIP Keyboard to input a value

Step1 Start a New Project

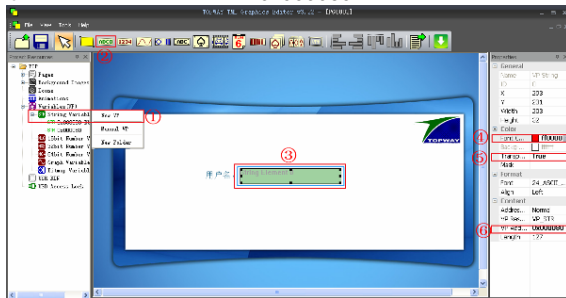
(please refer to the previous examples)

Step3 Link the page with the IMG_BKG

(please refer to the previous examples)

Step4 Allocate a String VP and show on page

- 1 In Resources Window, right click on String Variable, select "New VP"
- 2 In tools bar, select String Element
- 3 Create rectangle string element area on the PAGE
- 4 In String Element properties, set "Font Color" as "0xFF0000"
- 5 set "Transparent" as "True"
- 6 set "Font" as "32_ASCII_SysBold"
- 7 set "VP Address" as "0x000080"

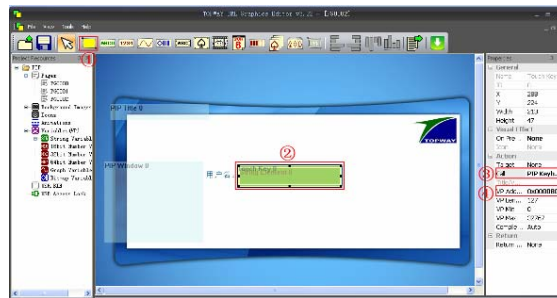


Step2 Built two pages, import a pictures as IMG_BKG (please refer to the previous examples)

- 1 Click on the working area of the PAGE, its properties will show on the right.
 - 2 In Properties window, select the imported Background Image.
- note: DP0000 is for main page; DP0001 for PIP (English) Keyboard page,

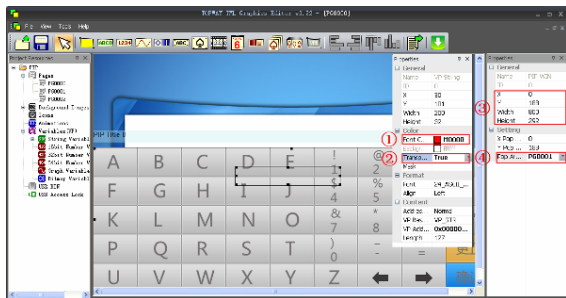
Step5 Built a Touch Key (TPK) to call PIP Keyboard

- 1 In tools bar, select Touch Key
- 2 Create rectangle touch key area on the PAGE
- 3 In Touch Key properties, set "Call" as "PIP Keyboard"
- 4 set "VP Address" as "0x000080"



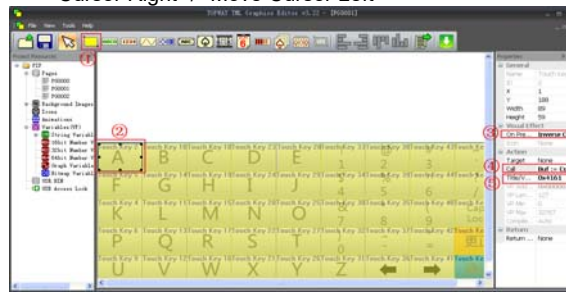
Step6 PIP Keyboard configuration

- 1 In POP TITLE properties set "Font Color" as "0xFF0000"
 - 2 set "Transparent" as "True"
 - 3 set "Font" as "32_ASCII_SysBold"
 - 4 In POP WIN properties, set "Pop Area of Page" as "PG0001" (keyboard page)
- set "PIP WIN X/Y" pop up position (hold the Ctrl key with mouse to shift the POP window content area)



Step7 Built PIP Keyboard's keys and its setting

- 1 In tools bar, select Touch Key
- 2 Create 45 rectangle touch keys area on the PAGE
- 3 In Touch Key properties set all 45 TPKs' "On Press Down" as "Inverse Color"
- 4 set the 40 TPKs' "Call" as "Buf:=Con(Buff,Cap/Nom(Byte0/Byte1))" set "Title/Value" as its own value(*1)
- 5 set the other 5 TPKs "Call" as special functions as "CapLock" / "vp:=DelLastchar(vp)" / "Enter" / "Move Cursor Right" / "Move Cursor Left"



*1 TPK Title/Value

Key Code	Upper	Lower
0x4161	A	a
0x4262	B	b
0x4363	C	c
0x4464	D	d
0x4565	E	e
0x4666	F	f
0x4767	G	g
0x4868	H	h
0x4969	I	i
0x4A6A	J	j
0x4B6B	K	k
0x4C6C	L	l
0x4D6D	M	m
0x4E6E	N	n
0x4F6F	O	o
0x5070	P	p
0x5171	Q	q
0x5272	R	r
0x5373	S	s
0x5474	T	t
0x5575	U	u
0x5676	V	v
0x5777	W	w
0x5878	X	x
0x5979	Y	y
0x5A7A	Z	z
0x7E60	~	`
0x2131	!	1
0x4032	@	2
0x2333	\$	3
0x2423	#	4
0x2535	%	5
0x2636	^	6
0x2637	&	7
0x2A38	*	8
0x2839	(9
0x2930)	0
0x5F2D	-	=
0x2B3D	+	=
0x3F2F	?	/

Step8 Compile and download (please refer to the previous examples)

Step9 Power on and Display

- 1 Disconnect the mini USB cable
- 2 Power on the Smart TFT LCD Module with 5V
- 3 Click on the input box
- 4 enter "user006" click "OK"



-- done --

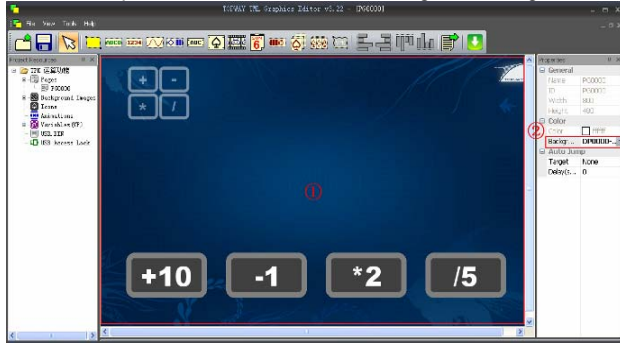
7.8 Using TPK to operate VP value

Step1 Start a New Project

(please refer to the previous examples)

Step3 Link the page with the IMG_BKG

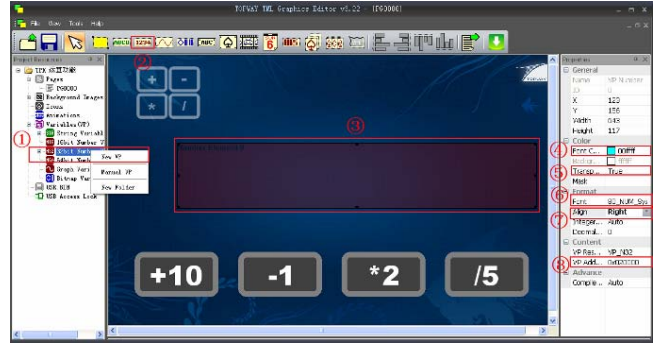
- Click on the working area of the page, its properties will show on the right.
- In Properties window, select the Background Image.



Step2 Built two pages, import a pictures as IMG_BKG (please refer to the previous examples)

Step4 Allocate a 32bit Number Variable (N32)

- In Resources window, right click on "32bit Number Variables" select : "New VP"

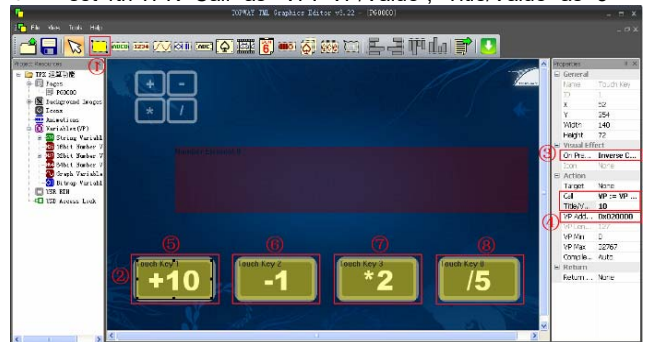


Step6 Built a Number Element and link with VP

- In tools bar, select number element
- Create a rectangle Number element area on page
- In Number Element properties, set "Font Color" as "0x00FFFF"
- set "Transparent" as "True"
- set Font" as "80_NUM_Sys"
- set "Align" as "Right"
- set "VP Address" as "0x020000"

Step6 Built 4 Touch-Key (TPK) on screen

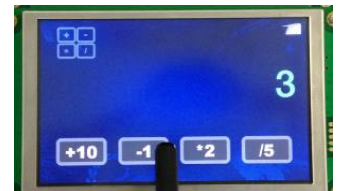
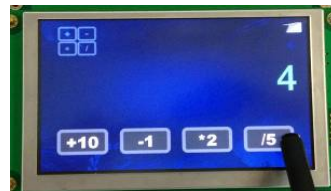
- In tools bar, select Touch-Key element
- Create 4 rectangle Touch-Key element on page
- In Touch-Key element properties set "On Press Down" as "Inverse Color"
- set "VP Address" as "0x020000"
- set 1st TPK "Call" as "VP:=VP+Value", "Title/Value" as "10"
- set 2nd TPK "Call" as "VP:=VP-Value", "Title/Value" as "1"
- set 3rd TPK "Call" as "VP:=VP*Value", "Title/Value" as "2"
- set 4th TPK "Call" as "VP:=VP/Value", "Title/Value" as "5"



Step8 Compile and download (please refer to the previous examples)

Step9 Power on and Display

- Disconnect the mini USB cable
- Power on the Smart TFT LCD Module with 5V
- Touch the TPKs, to get VP value update accordingly



-- done --

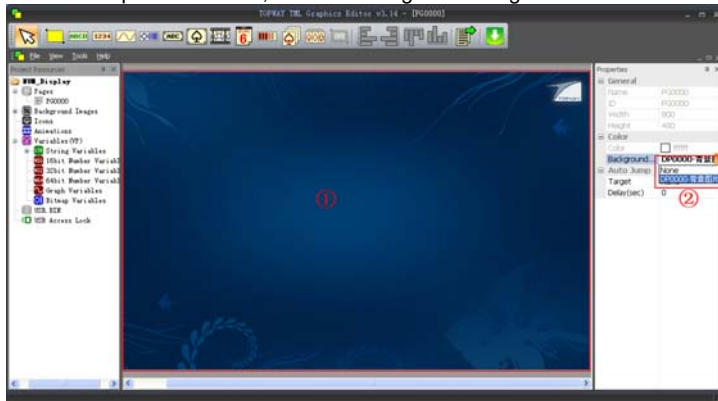
8 VP and COM communication Examples

8.1 Using Serial Command to update a Number variable

Step1 Start a New Project
(please refer to the previous examples)

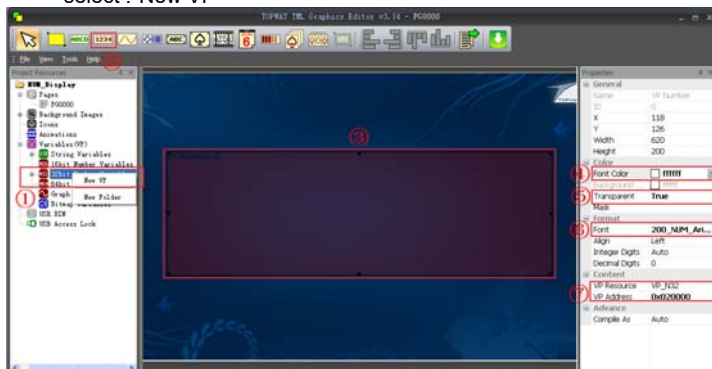
Step3 Link the page with the IMG_BKG

- ① Click on the working area of the page, its properties will show on the right.
- ② In Properties window, select a Background Image.



Step5 Allocate a VP_N32 variable

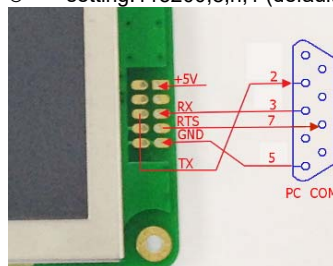
- ① In Resources window, right click on "32bit Number Variables" select : "New VP"



Step7 Compile and download (please refer to the previous examples)

Step8 Power on and Display

- ① Disconnect the mini USB cable
- ② Power on the Smart TFT LCD Module with 5V
- ③ connect the serial port
- ④ setting: 115200, 8, n, 1 (default)
- ⑤ send a command:
AA 44 00 02 00 00 00 02 90 FA CC 33 C3 3C
(write a value 0x000290FA(168186)
to VP address 0x00020000)



- ⑥ send a command:
AA 44 00 02 00 00 00 08 AD 6D CC 33 C3 3C
(write a value 0x0008AD6D(568685)
to VP address 0x00020000)



-- done --

9 Advance element Examples

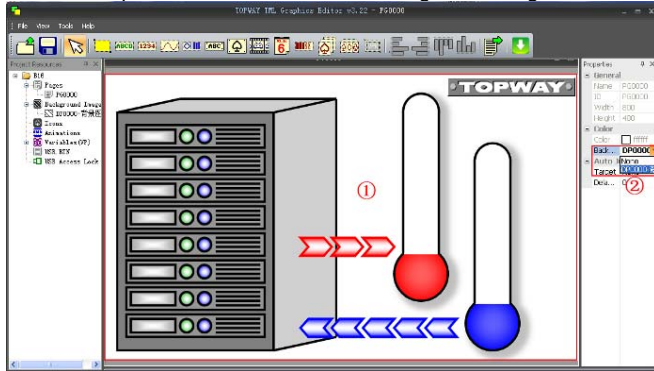
9.1 Show a Progress Bar

Step1 Start a New Project

(please refer to the previous examples)

Step3 Link the page with the IMG_BKG

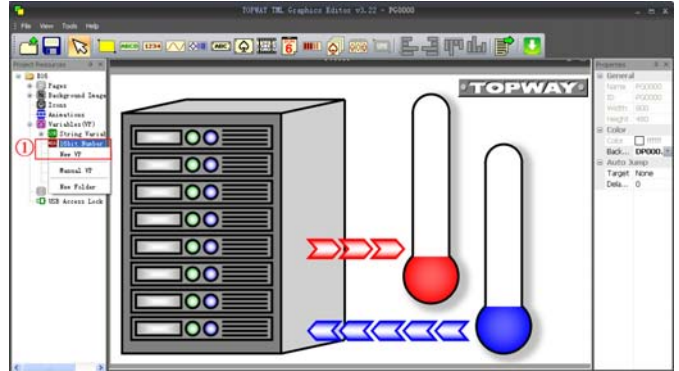
- ① Click on the working area of the page, its properties will show on the right.
- ② In Properties window, select the Background Image.



Step2 Built two pages, import a pictures as IMG_BKG (please refer to the previous examples)

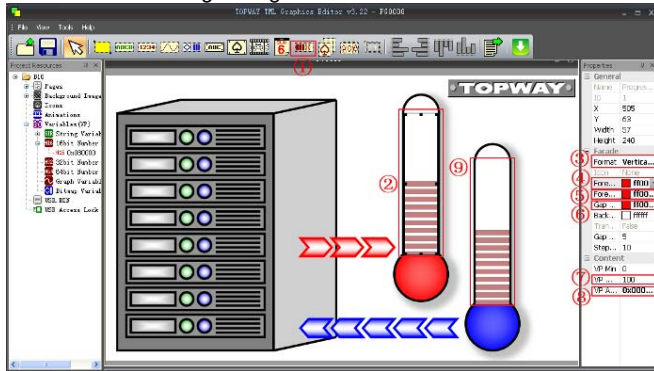
Step4 Allocate a 16bit Number Variable (N16)

- ① In Resources window, right click on "16bit Number Variables" select : "New VP"



Step5 Built a Progress Bar on screen

- ① In tools bar, select Progress Bar element
- ② Create 2 rectangle Progress Bar area on screen



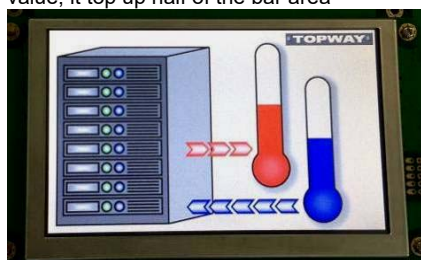
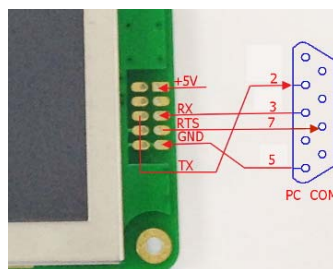
Step6 Config the Progress Bar and link to VP

- ③ In Progress Bar element properties set "Format" as "Vertical:Down to Up"
- ④ set "Forecolor 1" as "0xFF0000"
- ⑤ set "Forecolor 2" as "0xFF0000"
- ⑥ set "Gap Color" as "0xFF0000"
- ⑦ set "VP Max" as "100"
- ⑧ set "VP Address" as "VP 0x080000"
- ⑨ repeat the above to the second Progress Bar with another color

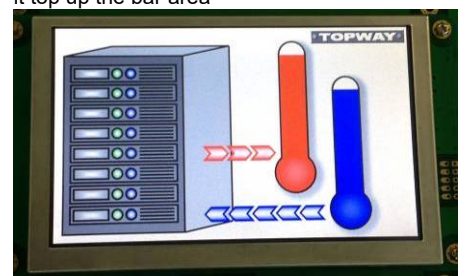
Step8 Compile and download (please refer to the previous examples)

Step9 Power on and Display

- ① Disconnect the mini USB cable
- ② Power on the Smart TFT LCD Module with 5V
- ③ connect the serial port
- ④ setting: 115200, 8, n, 1 (default)
- ⑤ send a command:
AA 3D 00 08 00 00 00 32 CC 33 C3 3C
(write a value 0x0032 (50)
to VP address 0x00080000)
Note: the value 50 is a half between MAX/MIN value, it top up half of the bar area



- ⑥ send a command:
AA 3D 00 08 00 00 00 64 CC 33 C3 3C
(write a value 0x0032 (100)
to VP address 0x00080000)
Note: the value 100 is same as MAX value,
it top up the bar area



-- done --

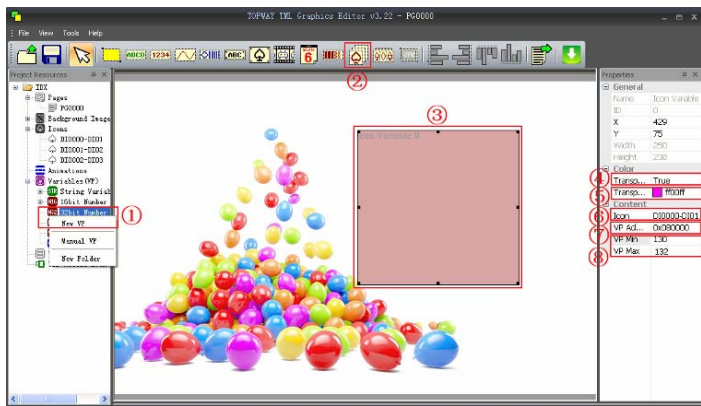
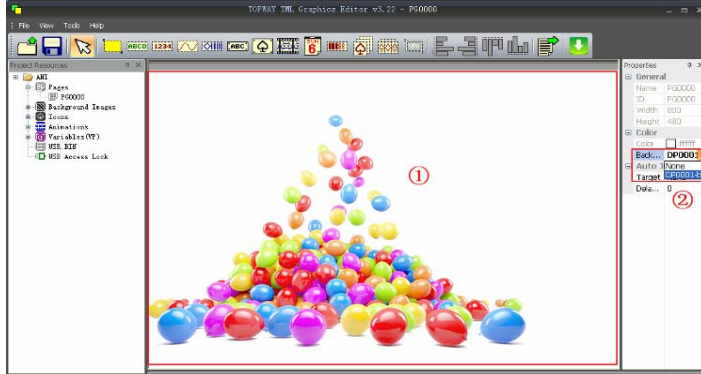
9.2 Using Index Icon

Step1 Start a New Project

(please refer to the previous examples)

Step3 Link the page with the IMG_BKG

- ① Click on the working area of the page, its properties will show on the right.
- ② In Properties window, select the Background Image.



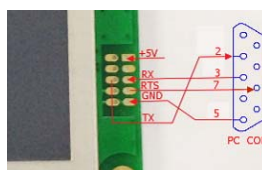
Step8 Power on and Display

- ① Disconnect the mini USB cable
- ② Power on the Smart TFT LCD Module with 5V
- ③ connect the serial port
- ④ setting: 115200, 8, n, 1 (default)

- ⑤ send a command:
AA 3D 00 08 00 00 00 82 CC 33 C3 3C
(write a value 0x0083 (131)
to VP address 0x00080000)

Note:

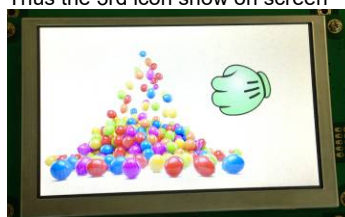
value 0x0082 is same as the (MIN value + 1),
thus, the 2nd icon show on screen



- ⑥ send a command:
AA 3D 00 08 00 00 00 84 CC 33 C3 3C
(write a value 0x0084 (132)
to VP address 0x00080000)

Note:

value 0x0084 is same as the (MIN value + 2),
Thus the 3rd icon show on screen



Step2 Built a page, import a pictures as IMG_BKG (please refer to the previous examples)

Step4 Import Icons

- ① In Resources window, right click Icons select "Import Icon"
- ② select the icon files
- ③ click "Open" to finish



Step5 Allocate a 16bit Number Variable (N16)

- ① In Resources window, right click on "16bit Number Variables" select : "New VP"

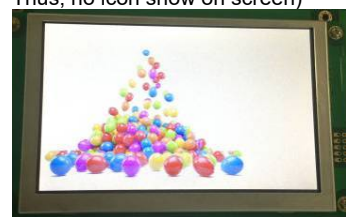
Step6 Built Indexed Icon and link to VP

- ① In tools bar, select Indexed Icon element
- ② Create a rectangle Indexed Icon area on screen
- ③ In Indexed Icon properties, set "Transparent" as "True"
- ④ set "Transparent Color" as "0xFF00FF" (icons' background)
- ⑤ set "ICON" as "DI0000" (the first imported icons)
- ⑥ set "VP Address" as "VP 0x080000"
- ⑦ set "VP Min" as "130" and "VP Max" as "132"

Step7 Compile and download

(please refer to the previous examples)

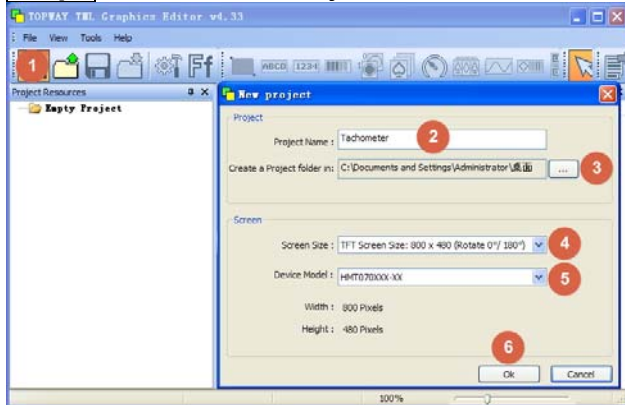
- ⑦ send a command:
AA 3D 00 08 00 00 00 85 CC 33 C3 3C
(write a value 0x0085 (133)
to VP address 0x00080000)
Note:
value is outside the range of MAX and MIN,
Thus, no icon show on screen



-- done --

9.3 Using Tachometer Element (Angle Open Mode) Application Example

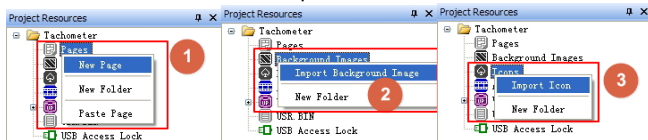
Step1 Start a New Project



- ⑥ start TOPWAY RGtools, start a new project through manual bar
File --> New Project
- ⑦ Enter the Project name
Project Name: Tachometer (as an example)
- ⑧ Select a Project folder location
Create a Project Folder in: desktop\Tachometer (as an example)
- ⑨ Select the display resolution of the SmartLCD
Screen Size: 800x480 (for this example HMT070CQ-C)
- ⑩ Click "OK"

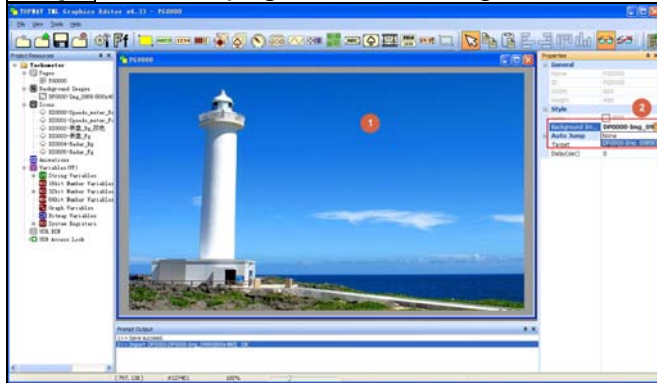
Step2 Built a Page, import Background Image and icons

- ① In Resource window, right click on Pages, select New Page
- ② In Resource window, right click on Background Images, select Import Background Image
- ③ Select images for background, select "Open"
- ④ In Resources window, right click Icons select Import Icon.
- ⑤ Select icon files, select "Open"



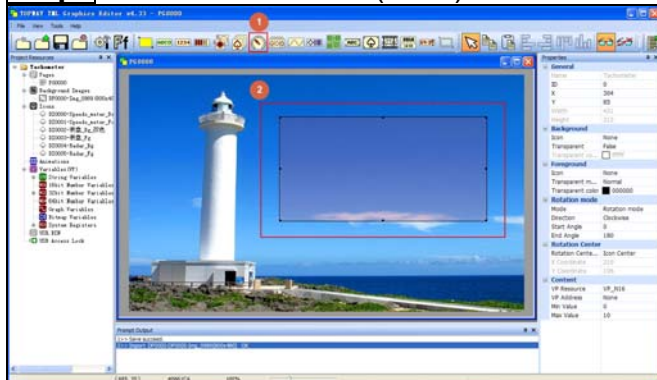
Note: RGTools supports BMP, JPG, PNG. In this case a 800x480 picture could be best fit to the display. or RGTools will resize it to fit the display

Step3 Link the page with the image



- ⑤ Click on the working area of the page, its properties will show on the right.
- ⑥ In Properties window, select DP0000 in the Background Image.

Step4 Built a Tachometer(TCM) Element on the page

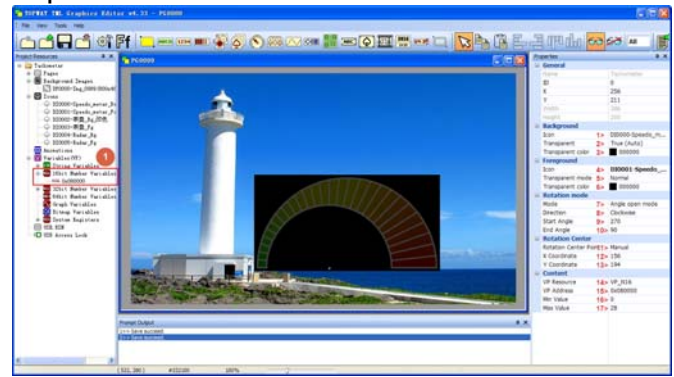


- ① In tool bar, select Tachometer
- ② Create a rectangle static icon area on page

Step5 Config the Tachometer Element as "Angle Open" mode

- ① In Resources window, right click on "16bit Number Variables" select : "New VP" to create a VP 0x080000.
 ② Config the TCM properties

No.	items	configuration
1	Bg. Icon	Select DI0000 (*1)
2	Bg. Transparent	Select True (Auto)
3	Bg. Transparent color	Black (Auto)
4	Fg. Icon	Select DI0001 (*2)
5	Fg. Transparent mode	Select Normal
6	Fg. Transparent color	Black
7	Mode	Select " Angle Open" mode
8	Direction	Select Clockwise
9	Start Angle	Select 270° (*3)
10	End Angle	Select 90° (*3)
11	Rotation Center Point	Manual
12	X Coordinate	Manual adjust to its center (*4)
13	Y Coordinate	Manual adjust to its center (*4)
14	VP Resource	Link to VP_N16
15	VP Address	Link with VP 0x080000
16	Min Value	0 as no light on
17	Max Value	28 as all light on



*1. DI0000



*3. Start/End Angle



*2. DI0001



*4. Rotation Center

Step6 Compile and download to the Smart TFT LCD Module (please refer to the previous examples)**Step7** Power on and Display

- ⑦ Disconnect the mini USB cable
 ⑧ Power on the Smart TFT LCD Module
 ⑨ connect the serial port
 ⑩ setting: 115200, 8, n, 1 (default)
- ⑪ send a values 0 to VP_N16(0x080000) by the following command
 AA 3D 00 08 00 00 0000 CC 33 C3 3C
- ⑫ send a values 23 to VP_N16(0x080000) by the following command
 AA 3D 00 08 00 00 0017 CC 33 C3 3C



No light on (0/28)



Some light on (23/28)

note. value outside 0~28 will hide the TCM form the screen

-- done --

9.4 Using Tachometer Element (Rotation Mode) Application Example (Meter)

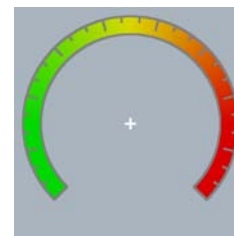
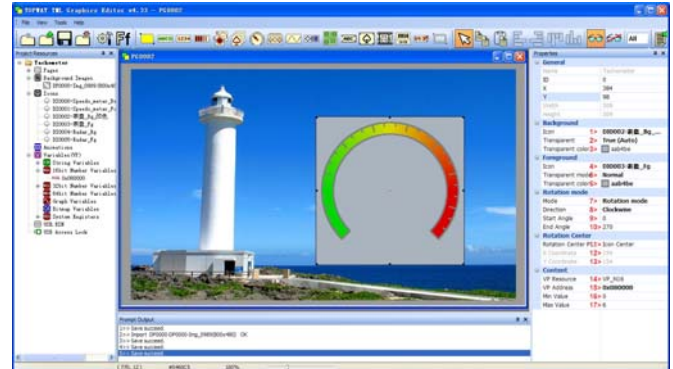
Step1 ~ **Step4** (please refer to the last examples)

Step5 Config the Tachometer(TCM) Element as "Rotation Mode" mode

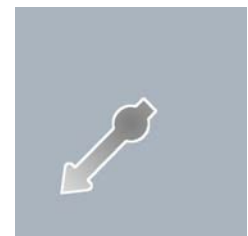
① In Resources window, right click on "16bit Number Variables" select : "New VP" to create a VP 0x080000.

② Config the TCM properties

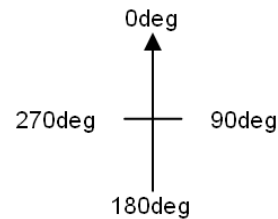
No.	items	configuration
1	Bg. Icon	Select DI0002 (*1)
2	Bg. Transparent	Select True (Auto)
3	Bg. Transparent color	0xaab4be (Auto)
4	Fg. Icon	Select DI0003 (*2)
5	Fg. Transparent mode	Select Normal
6	Fg. Transparent color	0xaab4be
7	Mode	Select "Rotation" mode
8	Direction	Select Clockwise
9	Start Angle	Select 0° (*3)
10	End Angle	Select 270°(*3)
11	Rotation Center Point	Manual
12	X Coordinate	Manual adjust to its center (*4)
13	Y Coordinate	Manual adjust to its center (*4)
14	VP Resource	Link to VP_N16
15	VP Address	Link with VP 0x080000
16	Min Value	0
17	Max Value	6



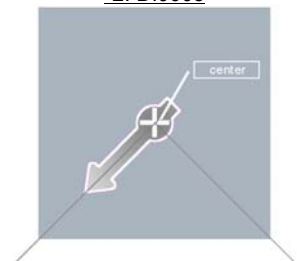
*1. DI0002



*2. DI0003



*3. Start/End Angle



*4. Rotation Center

Step6 Compile and download to the Smart TFT LCD Module (please refer to the previous examples)

Step7 Power on and Display

- ① Disconnect the mini USB cable
- ② Power on the Smart TFT LCD Module
- ③ connect the serial port
- ④ setting: 115200, 8, n, 1 (default)

- ⑤ send a values 0 to VP_N16(0x080000) by the following command
AA 3D 00 08 00 00 0000 CC 33 C3 3C



Radar point at 359deg (0/6)

- ⑥ send a values 3 to VP_N16(0x080000) by the following command
AA 3D 00 08 00 00 0003 CC 33 C3 3C



Radar rotate... (3/6)

note. value outside 0~6 will hide the TCM from the screen

-- done --

9.5 Using Tachometer Element (Rotation Mode) Application Example (Radar)

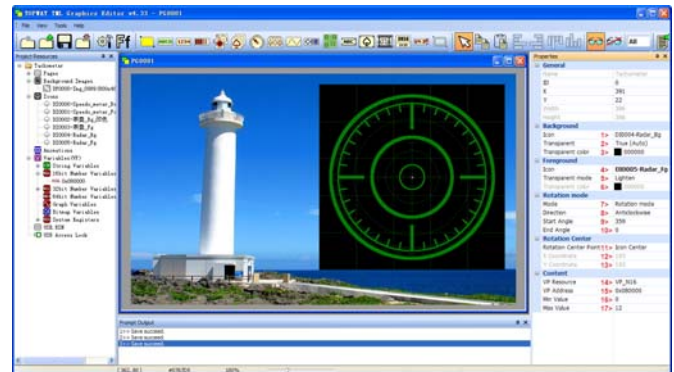
Step1 ~ **Step4** (please refer to the last examples)

Step5 Config the Tachometer(TCM) Element as "Rotation Mode" mode

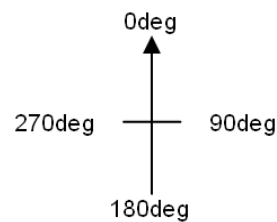
① In Resources window, right click on "16bit Number Variables" select : "New VP" to create a VP 0x080000.

② Config the TCM properties

No.	items	configuration
1	Bg. Icon	Select DI0004 (*1)
2	Bg. Transparent	Select True (Auto)
3	Bg. Transparent color	Black (Auto)
4	Fg. Icon	Select DI0005 (*2)
5	Fg. Transparent mode	Select Lighten
6	Fg. Transparent color	Black (Auto)
7	Mode	Select "Rotation" mode
8	Direction	Select Anti Clockwise
9	Start Angle	Select 0° (*3)
10	End Angle	Select 330°(*3)
11	Rotation Center Point	Manual
12	X Coordinate	Manual adjust to its center (*4)
13	Y Coordinate	Manual adjust to its center (*4)
14	VP Resource	Link to VP_N16
15	VP Address	Link with VP 0x080000
16	Min Value	0
17	Max Value	11



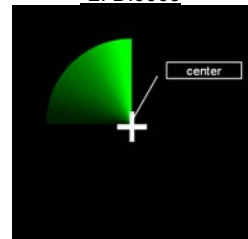
*1. DI0004



*3. Start/End Angle



*2. DI0005



*4. Rotation Center

Step6 Compile and download to the Smart TFT LCD Module (please refer to the previous examples)

Step7 Power on and Display

- ⑦ Disconnect the mini USB cable
- ⑧ Power on the Smart TFT LCD Module
- ⑨ connect the serial port
- ⑩ setting: 115200, 8, n, 1 (default)
- ⑪ send a values 0 to VP_N16(0x080000) by the following command
AA 3D 00 08 00 00 0000 CC 33 C3 3C
- ⑫ send a values 4 to VP_N16(0x080000) by the following command
AA 3D 00 08 00 00 0004 CC 33 C3 3C



Radar point at 0deg (0/11)



Radar rotate... (4/11)

note. value outside 0~12 will hide the TCM form the screen

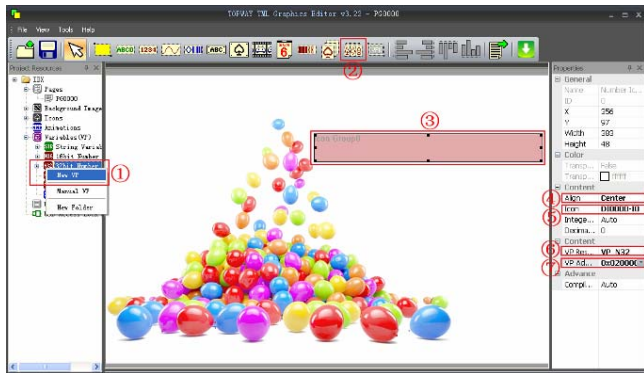
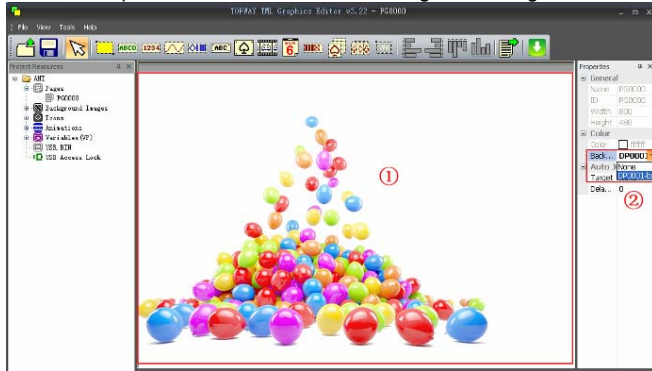
-- done --

9.6 Using Decimal Icon

Step1 Start a New Project
(please refer to the previous examples)

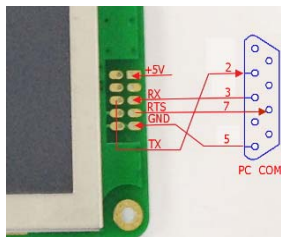
Step3 Link the page with the IMG_BKG

- ③ Click on the working area of the page, its properties will show on the right.
- ① In Properties window, select the Background Image.



Step8 Power on and Display

- ① Disconnect the mini USB cable
- ② Power on the Smart TFT LCD Module with 5V
- ③ connect the serial port
- ④ setting: 115200, 8, n, 1 (default)
- ⑤ send a command:
AA 44 00 02 00 00 00 1C B6 59 CC 33 C3 3C
(write a value 0x001CB659 (1881689)
to VP address: 0x00020000)
Note:
the value in decimal 1881689 are showing on
screen as indexed icons form the group of icons



Step2 Built a page, import a pictures as IMG_BKG (please refer to the previous examples)

Step4 Import Icons

- ④ In Resources window, right click Icons select "Import Icon"
- ③ select the icon files
- ① click "Open" to finish



Note:

1st icon	2nd icon	...	10th icon	11th icon	12th icon
0	1	...	9	0	-

Step5 Allocate a 32bit Number Variable (N32)

- ② In Resources window, right click on "32bit Number Variables" select : "New VP"

Step6 Built Indexed Icon and link to VP

- ① In tools bar, select Decimal Icon element
- ② Create a rectangle Decimal Icon element on screen
- ③ In Decimal Icon properties, set "Align" as "Center"
- ④ set "ICON" as "DI0000"
- ⑤ set "VP Resource" as "VP_N32"
- ⑥ set "VP Address" as "VP: 0x020000"

Step7 Compile and download

(please refer to the previous examples)

- ⑥ send a command:
AA 44 00 02 00 00 00 66 14 49 CC 33 C3 3C
(write a value 0x00661449 (6689865)
To VP address 0x00020000)
Note:
The value in decimal 6689865 are showing on screen
as indexed icons form the group of icons



-- done --

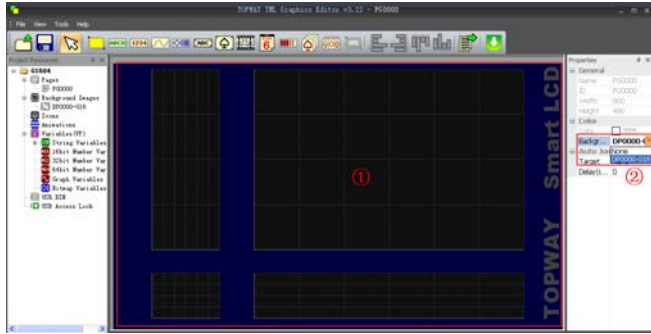
9.7 Using Graph Element

Step1 Start a New Project

(please refer to the previous examples)

Step3 Link the page with the IMG_BKG

- ① Click on the working area of the page, its properties will show on the right.
- ② In Properties window, select the Background Image.

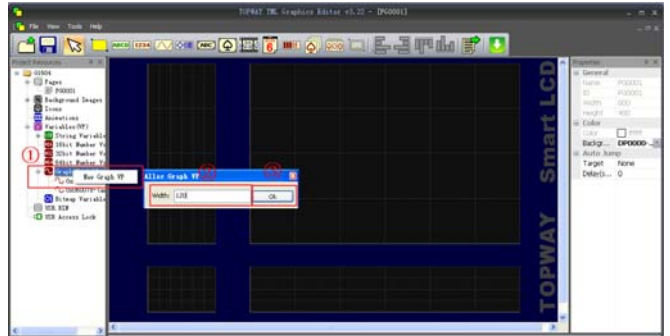


Step2 Built a page, import a pictures as

IMG_BKG (please refer to the previous examples)

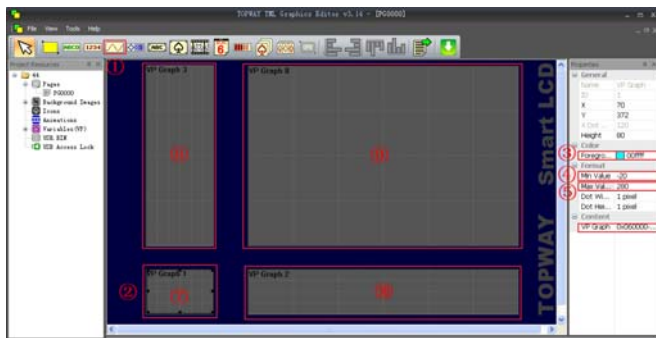
Step4 Allocate a Graph Variables

- ① In Resources window, right click on "Graph Variables" select "New Graph VP"
- ② Enter 120 into the Width setting
- ③ click "OK"



Step5 Built 4 Graph Element and link to VP

- ① In tools bar, select Graph Element
- ② Create 4 rectangle Graph area on screen
- ③ In Graph Properties, set all 4 "Foreground Color" as "0x00FFFF"
- ④ set all 4 "Min Value" as -20
- ⑤ set all 4 "Max Value" as 280
- ⑥ set all 4 "VP Graph" as "0x060000"
- ⑦ set one of the graphs "Dot Width" & "Dot Height" as 1 & 1
- ⑧ set one of the graphs "Dot Width" & "Dot Height" as 1 & 4
- ⑨ set one of the graphs "Dot Width" & "Dot Height" as 4 & 4
- ⑩ set one of the graphs "Dot Width" & "Dot Height" as 4 & 1



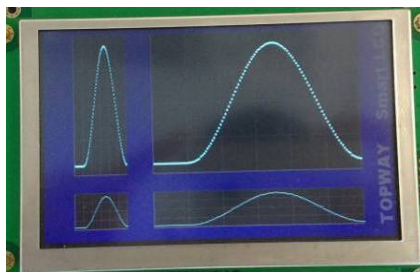
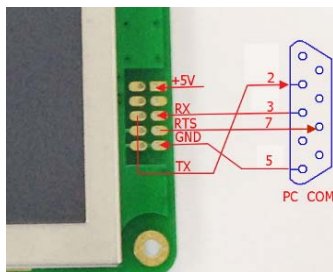
Step6 Compile and download

(please refer to the previous examples)

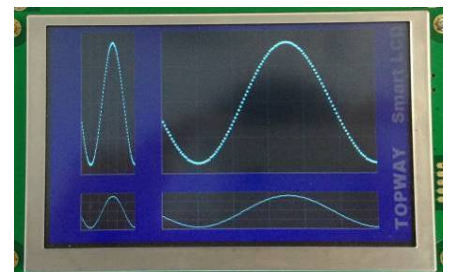
Step7 Power on and Display

- ① Disconnect the mini USB cable
- ② Power on the Smart TFT LCD Module with 5V
- ③ connect the serial port
- ④ setting: 115200,8,n,1 (default)
- ⑤ using "G16_write_rotate" commands (0x4E) to "insert" the data to the end of the graph array and shift all the data and show a "moving" graph.

Packet head	command	G16 Add	Insert at (*2)	Value (*3)	Packet tail	Note:
AA	4E	00 06 00 00	02 25	00 00	CC 33 C3 3C	*1. data insert command
AA	4E	00 06 00 00	02 25	00 00	CC 33 C3 3C	
AA	4E	00 06 00 00	02 25	00 01	CC 33 C3 3C	*2. insert at the end of the graph data array
AA	4E	00 06 00 00	02 25	00 02	CC 33 C3 3C	*3. SIN value sequence
:	:	:	:	:	:	



Inserted 100 data



Inserted 200 data

-- done --

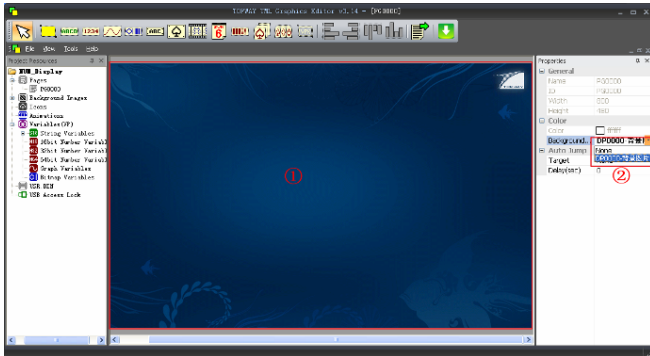
9.8 Show a Bitmap Element

Step1 Start a New Project

(please refer to the previous examples)

Step3 Link the page with the IMG_BKG

- ① Click on the working area of the page, its properties will show on the right.
- ② In Properties window, select the Background Image.

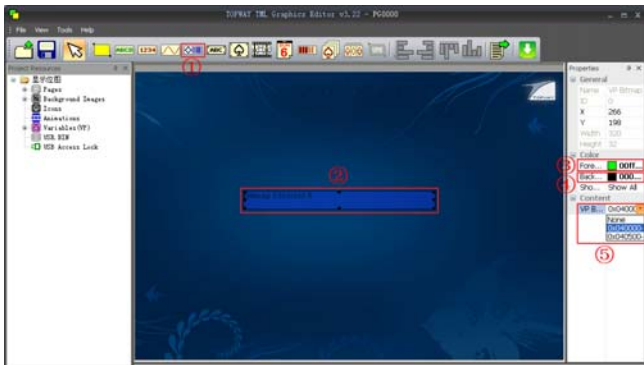
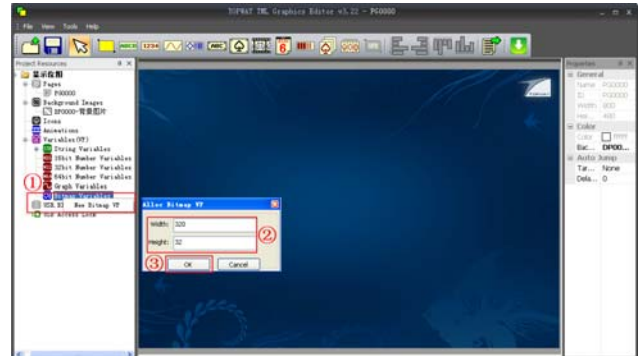


Step2 Built a page, import a pictures as IMG_BKG

(please refer to the previous examples)

Step4 Allocate a Bitmap variable

- ① In Resources window, right click on "Bitmap Variables" select "New Bitmap VP"
- ② Enter 320 to Width and 32 to Height
- ③ click "OK"



Step5 Built a Bitmap Element and link to VP

- ① In tool bar, select Bitmap Element
- ② Create a rectangle Bitmap area on screen
- ③ In Bitmap Properties, set "Foreground Color" as "0x00FF00"
- ④ set "Background Color" as "0x000000"
- ⑤ set "VP Bitmap" as "0x040000"

Note:

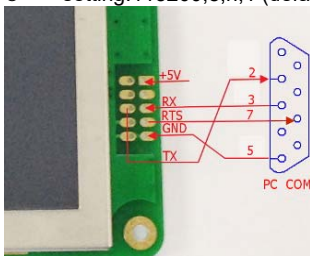
Bitmap data 1 for Foreground Color, 0 for Background Color

Step6 Compile and download

(please refer to the previous examples)

Step7 Power on and Display

- ① Power on the Smart TFT LCD Module with 5V
- ② connect the serial port
- ③ setting: 115200, 8, n, 1 (default)
- ④ send a command
AA 4B 00 04 00 00 00 00 05 00 CC 33 C3 3C
(write 0x00000500 ((320*32)/8=1280) byte data to address address 0x040000)



- ⑤ follow with 1280 byte of data
0XF9,0XCF,0X22,0X22,0X20,0X00,0X00,0X00,
0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,
0X00,0X00,0X00,0X00,0X00,.... ..

Data Example:







Black dot as 1; White dot as 0
Start from top left
1st row left 8 dots : 11111001(0xF9) (as 1st byte)
2nd row left 8 dots : 11001111(0xCF) (as 2nd byte)
3rd row left 8 dots : 00100010(0x22) (as 3rd byte)
:
:

-- done --







10 Appendix

10.1 RGTools Shortcut Keys



File Tools

icon	Name	Shortcut Key
	New Project	Ctrl + N
	Open Project	Ctrl + O
	Save Project	Ctrl + S
	Close Project	--
	Generate Files	F7
	Download to Module	F9

Alignment Tools

icon	Name	Shortcut Key
	Align Left	--
	Align Right	--
	Align Top	--
	Align Bottom	--
	Horizontal Distribute	--
	Vertical Distribute	--

Action Element Tools

icon	Mnemonic	Name	Shortcut Key
	TPK	Touch Key	Ctrl + K
	VPK	Virtual Key	--

String Element Tools

















icon	Mnemonic	Name	Shortcut Key
	STS	Static String	Ctrl + Shift + T
	STR	String Element	Ctrl + T
	N16 N32 N64	Number Element	Ctrl + M
	TMR	Timer Display	--
	RTC	Real Time Clock	Ctrl + R

Image Element Tools













icon	Mnemonic	Name	Shortcut Key
	ICO	Static Icon	Ctrl + Shift + I
	ANI	Animation Element	Ctrl + Shift + A
	IDX_BIT	Bit Icon	--
	IDX	Indexed Icon	--
	TCM	Tachometer	--
	I16 I32	Decimal Icon	--

Graphics Element Tools

icon	Mnemonic	Name	Shortcut Key
	B16	Progress Bar	Ctrl + P
	G16	Graph Element	Ctrl + G
	BP1	Bitmap Element	Ctrl + B
	QRC	QR Code Element	--
	DPD	Draw Pad	--

10.2 Project Limitation

10.2.1 PAGEs, Image-Resources and VP-Variables Limitation

icon	Mnemonic	Name	Linked VP.	Linked IMG	Limit	Memory Limit	Address range
	PAGE	Page	--	IMG_BKG	≤1000/project	256M byte (*4)	PG0000 ~ PG0999
	IMG_BKG	Background Image	--	--	≤10000/project (*1)		DP0000 ~ DP9999
	IMG_ICO	Icon	--	--	≤10000/project (*2)		DI0000 ~ DI9999
	IMG_ANI	Animation	--	--	≤1000/project (*2)(*3)		ANI000 ~ ANI999
	VP_STR	String Variable	--	--	≤1024 / project	1024(MAX) x (127+1)byte	0x000000 ~ 0x01FF80
	VP_N16	16bit Integer Variable	--	--	≤32512 / project	32512(MAX) x (2)byte	0x080000 ~ 0x08FDFF
	VP_N32	32bit Integer Variable	--	--	≤16128 / project	16128(MAX) x (4)byte	0x020000 ~ 0x02FBFF
	VP_N64	64bit Integer Variable	--	--	≤7936 / project	7936(MAX) x (8)byte	0x030000 ~ 0x03F7FF
	VP_G16	16bit Graph Variable	--	--	≤16384 / project	16384(MAX) x (8)byte (dynamic array allocation)	0x060000 ~ 0x07FFFF
	VP_BP1	Bitmap Variable	--	--	≤2048 / project	2048(MAX) x (64)byte (dynamic array allocation)	0x060000 ~ 0x05FFBF
	VP_REG	Register Variable	--	--	--	--	0xFFFF00 ~ 0xFFFFFFFF
	VP_TMR	Timer Variable	--	--	≤8 / project	--	0x02FFE0 ~ 0x02FFFC

Note:

*1. IMG_BKG must >1/2 of Full-Screen (RGTools auto resize to fit screen)












*2. IMG_ICON and IMG_ANI Size Limit

Project resolution	Size Limit
320x240	320x240 max. (Full-Screen)
480x272	480x272 max. (Full-Screen)
640x480	131072pixels (42% of Full Screen)
800x480	131072pixels (34% of Full Screen)
800x600	131072pixels (27% of Full Screen)
1366x480	131072pixels (20% of Full Screen)

*3. 1~64frame max.

*4. Memory Size may vary by model.

10.2.2 Page's Elements Limitation

icon	Mnemonic	Name	Linked VP.	Linked IMG	Limit	Memory Limit	ID range
	TPK	Touch Key	--	1x IMG_ICO or 1x IMG_BKG (*1)	≤256 / PAGE	--	0 ~ 255 (each page)
	VPK	Virtual Key	VP_N16 (*2) VP_N32 (*2)	--	≤64 / PAGE	--	0 ~ 63 (each page)
	STS	Static String	--	--	≤128 / PAGE	--	0 ~ 127 (each page)
	STR	String Element	VP_STR	--	≤128 / PAGE	--	0 ~ 127 (each page)
	N16 N32 N64	Number Element	VP_N16 VP_N32 VP_V64	--	≤120 / PAGE	-- -- --	0 ~ 119 (each page)
	TMR	Timer Display	VP_N32 (timer only)	--	≤8 / PAGE	--	0 ~ 7 (each page)
	RTC	Real Time Clock	--	--	≤8 / PAGE	--	0 ~ 7 (each page)
	ICO	Static Icon	--	1x IMG_ICO	≤128 / PAGE	--	0 ~ 127 (each page)
	ANI	Animation Element	--	1x IMG_ANI	≤8 / PAGE	--	0 ~ 7 (each page)
	IDX_BIT	Bit Icon	VP_N16 VP_N32	2(MAX) x IMG_ICON	≤64 / PAGE	--	0 ~ 63 (each page)
	IDX	Indexed Icon	VP_N16	10000(MAX) x IMG_ICON	≤64 / PAGE	--	0 ~ 63 (each page)
	TCM	Tachometer	VP_N16	181(MAX) x IMG_ICON		--	
	I16 I32	Decimal Icon	VP_N16 VP_N32	12x IMG_ICON (0~9, ".", ",", "-")	≤32 / PAGE	--	0 ~ 31 (each page)
	B16	Progress Bar	VP_N16	1x IMG_ICON	≤32 / PAGE	--	0 ~ 31 (each page)
	G16	Graph Element	VP_G16	--	≤8 / PAGE	--	0 ~ 7 (each page)
	BP1	Bitmap Element	VP_BP1	--	≤32 / PAGE	--	0 ~ 31 (each page)
	QRC	QR Code Element	VP_STR VP_N16 (in seq.)	--	≤4 / PAGE	--	0 ~ 3 (each page)
	DPD	Draw Pad	VP_N16 (in seq.)	--	≤4 / PAGE	--	0 ~ 3 (each page)

Note:

*1. IMG_ICO for key-down display, IMG_BKG will be cropped to key size for key-down display

*2. Monitoring values

*3. Element display Size: 8x8 (MIN), except Bitmap Element

10.2.3 System Registers

Address	System Register	Descriptions	Note
0xFFFF00 : 0xFFFF07	Timer_Ctrl0 : Timer_Ctrl7	D[0]=0 : stop the timer D[0]=1 : start the timer D[1]=0 : count down, time down to 0x00000000 D[1]=1 : count up, time up to 0x7fffffff D[7:2]=000000 : reserved	(*1)
0xFFFF10	RTC_Year	Value=0~99	0xffff10~0xffff16 should be written at the same time
0xFFFF11	RTC_Month	Value=1~12	
0xFFFF12	RTC_Day	Value=1~31	
0xFFFF13	RTC_Hour	Value=0~23	
0xFFFF14	RTC_Minute	Value=0~59	
0xFFFF15	RTC_Second	Value=0~59	
0xFFFF16	RTC_Set	Value=1 : sync the time	
0xFFFF20	Buzzer	Value=0~63 buzzer beeping duration (*1)	See Project Setting for details
0xFFFF21	Backlight	Value=0~63 Real time backlight brightness setting (*1)	
0xFFFF22	ScreenSaverBacklight	Value=0~63 Screen Saver Backlight Brightness (*1)	see Screen Saver section for details
0xFFFF23	ScreenSaverPage_H	Value=0~999	
0xFFFF24	ScreenSaverPage_L	Screen Saver display Page (*2)	
0xFFFF25	ScreenSaver Timer1_H	Value=0~65535	
0xFFFF26	ScreenSaver Timer1_L	Screen saver delay time 1 (*2)	
0xFFFF27	ScreenSaver Timer2_H	Value=0~65535	
0xFFFF28	ScreenSaver Timer2_L	Screen saver delay time 2 (*2)	
0xFFFF30	Font_CodePage	Font Code Page index value (*1) (*3)	
0xFFFF31	Font_Country	Font Country index value (*1) (*4)	

Note:

*1. Value can be update by command System register write (0xb3) to access the register

*2. Value should be written at the same time by command `Sucesstive write (0x82)`

*3. Code Page index value

index	Code Page
1	437(OEM United States)
2	737(OEM Greek437G)
3	852(OEM Latin II)
4	860(OEM Portuguese)
5	863(OEM Canadian French)
6	865(OEM Nordic)
7	866(OEM Russian)
8	874(ANSI/OEM Thai)
9	932(ANSI/OEM Japanese Shift JIS)
10	1250(ANSI Central Europe)
11	1251(ANSI Cyrillic)

index	Code Page
12	1252(ANSI Latin I)
13	1253(ANSI Greek)
14	1254(ANSI Turkish)
15	1255(ANSI Hebrew)
16	1256(ANSI Arabic)
17	1257(ANSI Baltic)
18	1258(ANSI/OEM Viet Nam)
19	GB2312 (Simplified Chinese)
20	GBK (Simplified Chinese)
21	ECU-KR (Korea)
22	BIG5 (Traditional Chinese)

*4. Country index value

index	Code Page
1	USA
2	France
3	Germany
4	UK
5	Denmark I
6	Denmark II

index	Code Page
7	Denmark II
8	Sweden
9	Italy
10	Spain
11	Japan

11 Precautions of using LCD Modules

Mounting

- For mounting use the holes arranged in the four corners of the LCD Module.
- Make sure to provide an even force on to LCD module. Uneven force (ex. twisted stress) should not be applied to the module. The casing on which a module is mounted should have sufficient strength to absorb any external force, so the force can't be transmitted directly to the module.
- It is suggested to attach a transparent protective plate to the surface of the module in order to protect the polarizer. It should have sufficient strength to resist external force.
- The housing should provide sufficient thermal radiation to satisfy the temperature specification.
- Acetic Acid type and Chlorine-type materials for the cover case are not desirable because the former generates corrosive gases, which may attack the polarizer at high temperatures which may cause circuit break by electro-chemical reactions.
- Do not touch, push or rub the exposed polarizer with glass, tweezers or anything harder than HB pencil lead. Never treat the polarizer with chemical agents. Do not touch the surface of polarizer with bare hand or greasy cloth. Otherwise it may result in some cosmetics deterioration of the polarizer.
- When the surface becomes dusty, please wipe gently with absorbent cotton or another soft material. Do not use acetone, toluene and alcohol because they cause chemical damage to the polarizer.
- Wipe off saliva or water drops as soon as possible. Their long time contact with polarizer may affect the appearance.

Operating

- Spike noise may cause deterioration of the circuitry. Noise should be within a range of $\pm 200\text{mV}$. (Over and under voltage)
- The LCD response time is depends on the temperature. (At lower temperatures, it becomes slower)
- Brightness depends on the temperature as well. (At lower temperatures, it becomes less bright and it takes more time until the brightness is stable after power on).
- Try to avoid sudden temperature change, because they may cause condensation. Condensation may damage the polarizer or the circuitry. After fading condensation a smear or spot may occur.
- When fixed patterns are displayed for a long time, remnant images are likely to occur.
- The LCD module incorporates high frequency circuitry. Sufficient suppression of electromagnetic interference shall be done by system manufacturers. Grounding and shielding methods may be important to minimized interferences.

Electrostatic Discharge Control

- Since a module is composed of electronic circuits, it may be affected by electrostatic discharge.
- Make sure that the worker, who is assembling the module into equipment, is connected to ground through a ESD wrist band or insure other ESD protection.
- Avoid touching any electrical contact of the module without proper ESD protection.

Strong Light Exposure

- Strong light exposure causes degradation of polarizer and color filter.

Storage

- When storing modules as spares for a long time, precautions are necessary.
- Store the LCD modules in a dark place.
- Do not expose the module to sunlight or fluorescent light.
- Keep the temperature between 5°C and 35°C at normal humidity.
- The polarizer surface should not come in contact with any other object. It is recommended that they be stored in the container in which they were shipped.

Protection Film

- When the protection film is peeled off, static electricity is generated between the film and polarizer. The film should be peeled off slowly and carefully by people who are electrically grounded. It is suggested to do that process while using an ion air blower or other suitable ESD equipment.
- The protection film is attached to the polarizer with a small amount of glue. If some stress is applied e.g. by rubbing the protection film against the polarizer during the time it is peeled off, some protection film may remain on top of the polarizer. Please carefully peel off the remaining protection film without rubbing it against the polarizer.
- When the module, with protection film attached, is stored for a long time, some very small amount of glue may remain still on the polarizer after the protection film is peeled off. You can remove the glue easily. In such a case please wipe them off with absorbent cotton or another soft material.

Transportation

- The LCD modules should not be exposed to drop, shock, excessive pressure, water or sunshine during transportation.

12 Revisions

Rev.	Descriptions	by	Release Date
0.01	- Draft Release	K.C.	2016-09-12
1.00	- Preliminary New Release	K.C.	2017-02-18
1.01	- Typo corrections - Update 3.1 RGTools Layout diagram - Add 4.2 commands descriptions 0x79, 0x7A, 0xE7, 0xE8	K.C.	2017-08-01
1.02	- Typo corrections - Add VP_REG & VP_TMR descriptions - Update 4.1 RGTools Layout diagram - Update 4.3.9 Build Font <2> diagram - Refine 4.4 elements properties list and descriptions - Refine 4.5 page properties list and descriptions - Add 5.1.3 note and 10.21 note	K.C.	2017-09-01
1.03	- Typo corrections - Add Command 0x82, 0x83 - Update Command 0xE2 - Add Distribute Tools	K.C.	2017-12-14
1.04	- Typo corrections - Add 10.2.3 System Register - Add 4.4.19 ~ 4.4.27 Predefined Touch Key	K.C.	2019-03-13
1.05	- Update 4.1 RGTools Layout diagram - Add 4.4.28 Predefined Touch Key - Set (English keyboard) - Typo corrections	K.C.	2020-03-25
1.06	- Typo corrections	K. C.	2020-06-03