MET-TP 2nd Course

A Guide to MONITOUCH

"A book is a gift you can open again and again."

ALMAWARED
ENGINEERING
& TRADING SAE

1st edition 11/2011

MET-TP 2nd Course

A Guide to MONITOUCH

"A book is a gift you can open again and again."

ALMAWARED
ENGINEERING
& TRADING SAE

Table of Contents

1. Introduction

Introduction

Welcome to the second course in the **MET-TP** series. In order to excel in this course and get the best advantage of it, you should be aware of Programmable Logic Controllers (PLCs) and how they work, so we highly recommend taking the first course in this training series.

Al-**M**awared **E**ngineering and **T**rading **T**raining **P**rogram offers a training service so that you can plan over the time the growth of the device knowledge, from the frequency inverters to the soft starters for asynchronous motors up to the PLC "Programmable Logic Controller"; Touch screens and SCADA system. Courses are targeted to engineers, technicians, users and installers and to the service personal as well. MET suggests courses that are mainly oriented to the use of the drives and of the automation system.

Our highly trained engineers will guide you step by step through each training course, allowing you to perform each step by yourself from small examples to large applications to help you practice everything you learn during the training course.

This course covers the **Hakko Touch Screens.** Upon completion of this course you will be able to:

- Connect Hakko Touch Screen to any type of PLC.
- Open existing projects in Hakko programming tool (V-SFTV5)
- Develop new software or modify an old one.
- Download and upload any software.
- Implement all the basic functions and shapes needed in software to build a fully functioned application.
- Know all the essential data needed to put you on the first step of learning advanced HMI systems like Indusoft SCADA.

2. Company Overview

Experience and high responsiveness

Year 2001 is not only a date for ALMAWARED ENGINEERING AND TRADING S.A.E (MET). It represents the starting of a company that is today specialized in the industrial automation, mechanical and electrical power transmission field, thanks to the entrepreneurial capabilities from foundation members Mr. Abdel Aziz Aboul Atta, Ms. Bahia Khairy, Eng. Mohamed Abdel Aziz, Eng. Khalid Abdel Aziz and Eng. Khalid Ateya, expertise and firm commitment of the promoting partners.

MET partners achieved the quality certificates, with the aim to grant a good quality system for the different market needs to satisfy most of their customers' needs by providing complete solutions or individual tailor-made solutions. Only a long experience allows a company to reach in a flexible and wide way to market demands, with a complete range of products and services.

Flexibility and rapidity of product range

Our structure is composed of real problem solvers who study the customer requirements and orient him towards simple and innovative solutions thanks to the structure of MET product range that gives our highly professional technical engineers a wide area of solutions. MET product range comprises a whole variety of automation, electrical and mechanical equipment such as PLCs, touch screens, SCADA systems, flow meters, density meters, BMS components, inverters, soft starters, AC/DC motors, Servo systems, gearboxes, clutches and brakes. Distributed Control System (DCS) is the latest product that MET provides to our customers.

A strong presence in the market

Our experienced sales and marketing team is highly technical and customer-oriented. They combine premium customer support with years of industry experience to develop and sustain long-term relationships.

Our team ensures each customer is treated with professionalism and integrity as we work hard to understand our customers' needs and to provide solutions to meet those needs, so we serve a wide variety of market fields such as soup and fats, sugar industries, petroleum and refineries, cement factories, iron and steel mills, pharmaceutics and cosmetics, textiles and dyeing, paints and chemicals, plastics and petrochemicals, food & beverages industries, paper and printing, packing and wrapping, pump & water treatment plants, sewage & water treatment plants and commercial HVAC.

3. Hardware Overview

Why do we use Touch Screens?

Programmable Logic Controllers (PLCs) were invented to replace the enormous number of timers, counters and other classic control components which cost a lot in most projects away from the fact that these components can malfunction at any time leaving customers in the headache of replacing them looking for the same component that does the same exact function with the same ratings. So PLCs are used to replace most of the inner panel components using software which varies from each PLC to the other.

Similar to PLCs, Touch Screens are used to replace the outer panel components such as switches, lamps, numerical displays and even alarm sirens in some cases.

Touch Screens also give some extra functions which were not available before like keeping history of the alarms of a system, trend graphs, and entering settings through a keypad on screen…etc.

About Hakko, MONITOUCH:

Hakko, MONITOUCH is a world leading manufacturer of graphical operator interface products ranging from 5.7" up to 15". Its collective expertise as an HMI manufacturer spans over 20 years with current shipments of over 8,000 HMIs per month. For most of its history, its focus has been on the Asian & European markets where it has emerged as an HMI technology leader. Now, its focus extends to meet the complete HMI needs of the North American market by offering high quality, full featured HMI solutions at a remarkable price.

MET understands that providing optimal support is important for your success! MET meets this need with services such as local inventory, technical support, application development, comprehensive training, repair services and more, all designed to make your life easier. No other company offers such a complete package at a better price!

Advantages of using Hakko, MONITOUCH:

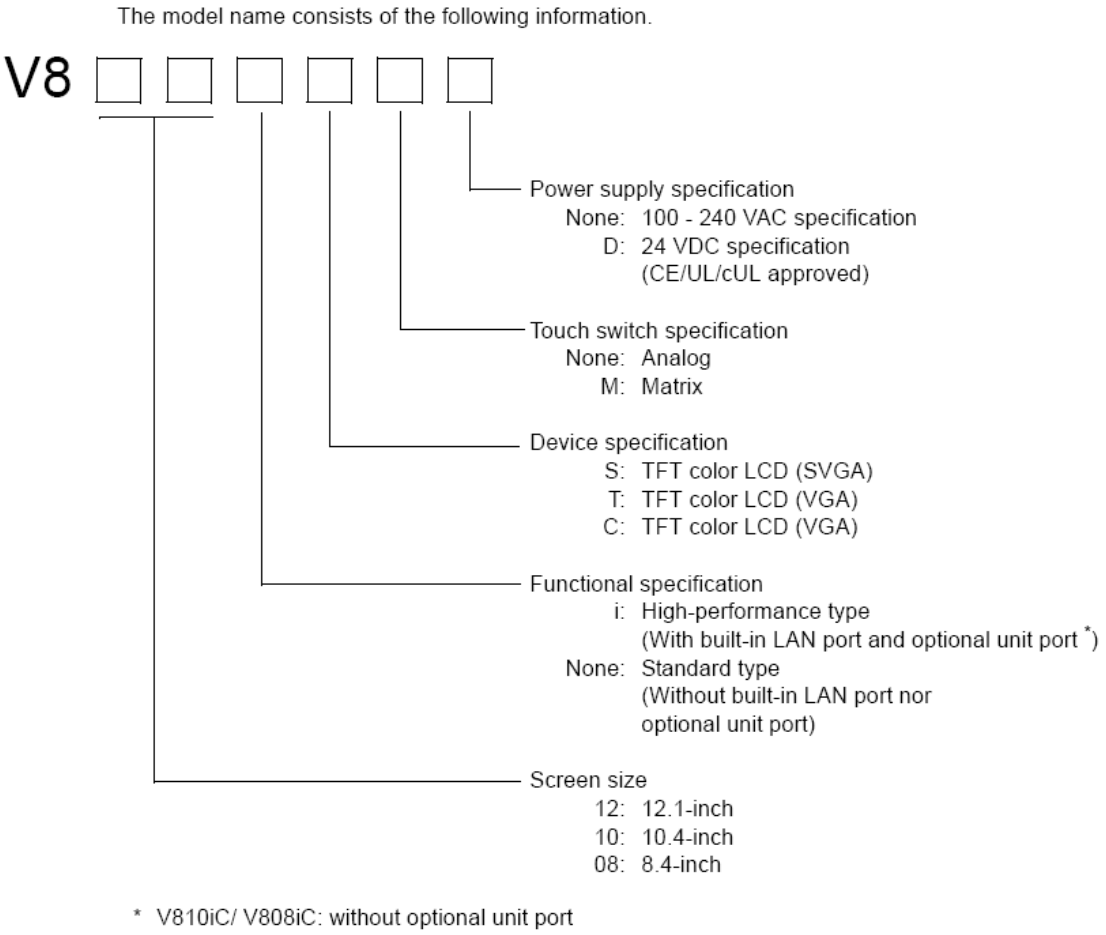
- High Quality & Reliability
- Connectivity to more than 1,000 Devices
- Powerful Features & Graphics
- Full Remote Monitor, Control and Data Acquisition
- Flexible choices

Product Range

Hakko, MONITOUCH touch screens range from 5.7” up to 15” in size.

V8 is the latest Hakko touch screen series with many new features that will be covered in this chapter. Also S8, an economical touch screen, is available in market now with both monochrome and full colors.

A simple way to determine the important features of a Hakko touch screen is through its model number as follows:



For example: V806iCD would be a V8 Series, almost 6 inches (5.7”), High-performance (with built-in LAN port), TFT color LCD (VGA) and 24VDC supply.

Features

V8 series has amazing features that make it superior to all previous series.

1. TFT Color LCD
2. Resolution up to 800 X 600
3. Up to 65,536 colors allowing the use of JPEG and BMP images, thus giving opportunities of creating a realistic attractive design for the screen.
4. Higher touch sensitivity and response
5. Connectivity to video, RGB and sound units with high speed displaying of 30 frames per second (FPS) for videos.
6. Connectivity to up to 8 different devices at the same time through serial and Ethernet communication ports. Also, up to 31 similar devices can be connected through each serial port.

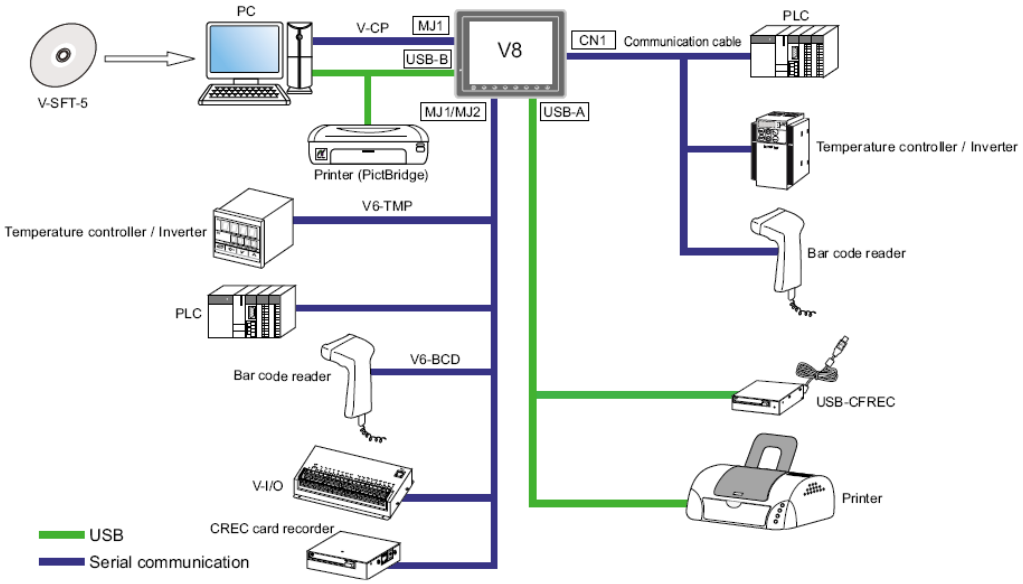
System Configuration:

All V8 models have two USB ports.

USB-A (Master) can be connected to a printer, keyboard, mouse or a card reader/writer.

USB-B (Slave) is used for transferring the software. Plus it can be connected to a PictBridge printer.

V8 Serial ports are used to connect PCs, PLCs, inverters, temperature controllers and barcode readers.



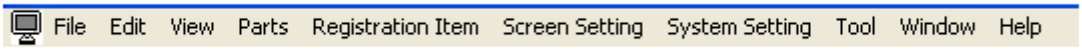
Drivers List:

Hakko touch screens support a huge number of PLC drivers, temperature controllers and inverters such as TOSHIBA, Siemens, and Allen Bradley …etc. For the complete drivers list, check V-SFT5 (touch screen programming tool) or the CD attached with this training document in a PDF file named “Connection Compatibility List”.

4. Software Interface Overview

Main Toolbar

In this section, we'll be covering the most important menu items in the main toolbar of V-SFT5 that are often used while developing software for touch screens. Other menu items are left for the user to discover along his way of learning the software as they will not be much important for us at the moment.
These are the pull down menus that appear in the main toolbar while a project is open and the often used items in each menu:



[File]:

- **New:** for opening a new project
- **Open:** for opening an existing project
- **Save:** for saving projects
- **Save as:** for saving a project under a different name or in another directory
- **Transfer:** this is the window we'll use to transfer data to and from the touch screen
- **Print:** here you'll find the option to print the current screen or all screens in your project
- **Start Emulator:** this is a very important tool to test your project and see how it really works before you transfer it to the screen.

[Edit]:

Items in this menu can be found also when you right click an object in a screen. It includes all the tools you may use to work with objects like cut, copy, paste, group, order (to put an object behind or in front of another object), align (to align two or more objects together from any side), put all objects in the same width or length or both… etc.

[View]:

- **Toolbar:** the most used toolbars would be Standard, Draw, Layout and Parts. So we'll suggest you check those toolbars if they're not checked.

- **View:** here you'll find (*Project View*) which appears by default on the left. This includes all the configuration of the current project. Also you'll find (*Catalog View*) which appears by default on the right. This includes all parts or objects you could use in your application or project. So if these two are by mistake unchecked, please check them.

- **Jump:** to jump to a certain screen in the project

- **Preview:** to go to the previous screen

- **Next:** to go to the next screen

- **Screen List:** it lists all the screens used in your project

[Parts]:

Items in this menu are exactly the same as in [*Catalog View*] area. It includes, as we said before, all parts or objects you could use in your application or project. We'll cover most of these items as we move forward with the training.

[Registration Item]:

The most used items in this menu are:

- **Macro Block:** this is the window where you write a script that does extra functions that are not available with switches, lamps or parts in general. Like reading the values of three variables and calculating their average and moving it to a fourth variable.

- **Overlap Library:** this is where you design your overlap screens.

[Screen Setting]:

The most important feature you can use in this menu is to change the background color of the current screen or all screens. You can access that though: Screen Setting /Main/ Back Color.

[System Setting]:

The most used item in this menu would be [Edit Model Selection]. From here you can change the model of the touch screen that your current project is using. This might be useful in cases where you need to replace an old or small touch screen by a new or larger touch screen. So you upload software from the old screen, change its model and then download software to the new touch screen.

[Tool]:

- **Error Check:** use this to check for errors in your software before you download it.
- **List of Memory Use:** here you can see how much is every screen, overlap, message…etc. consuming from the touch screen memory.
- **Memory Address Use:** this is used to search for registers in your project. Just type the memory address and a window will appear showing in which screens it is used.

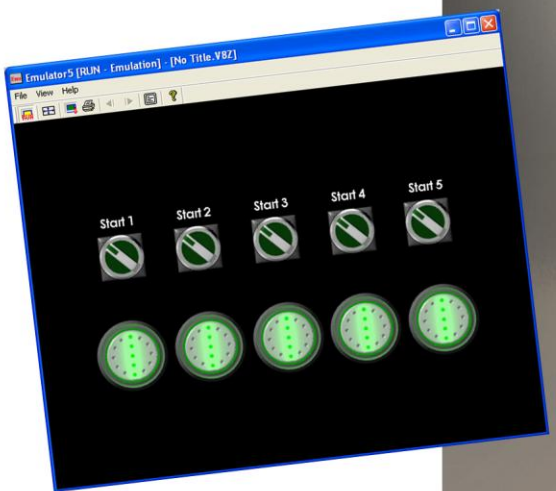
[Window]:

This menu is mainly to arrange screen windows when you have more than one open.

[Help]:

This includes all the manuals whether hardware or software. Also you can download the latest manuals from this menu as well.

5. Getting Started

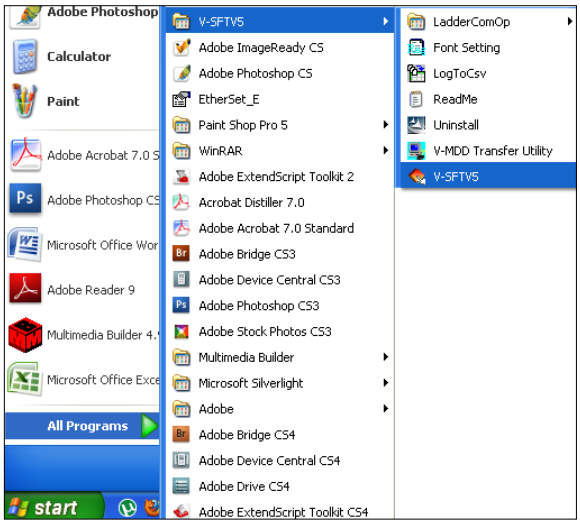


Starting Up the software

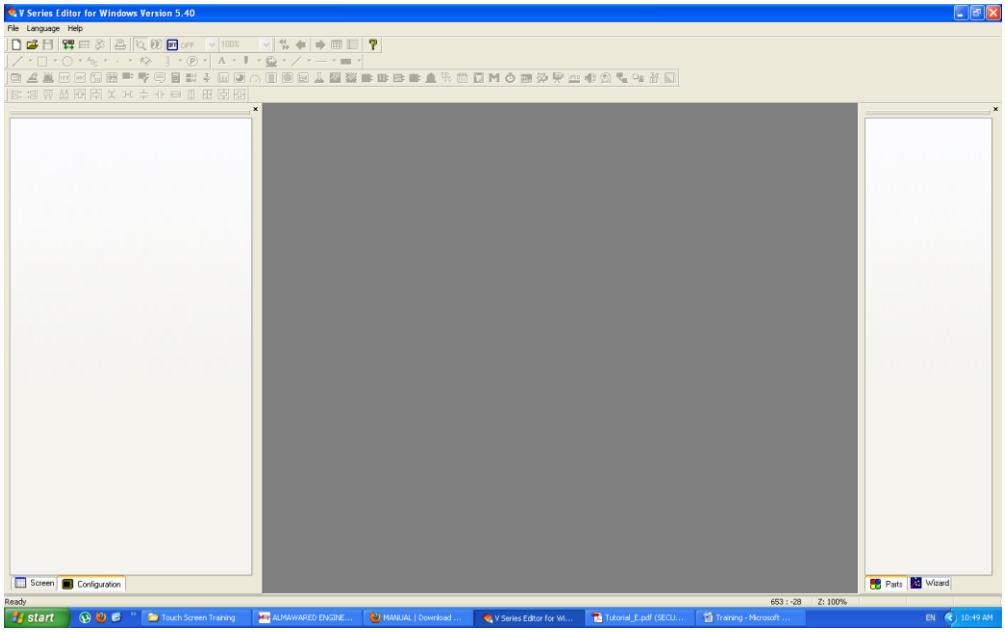
In this section, we'll be covering the most important procedures to make a new project and the parts you can place in your project and how to assign different functions and memory address to each part you use.

5.1. Opening and Closing a File

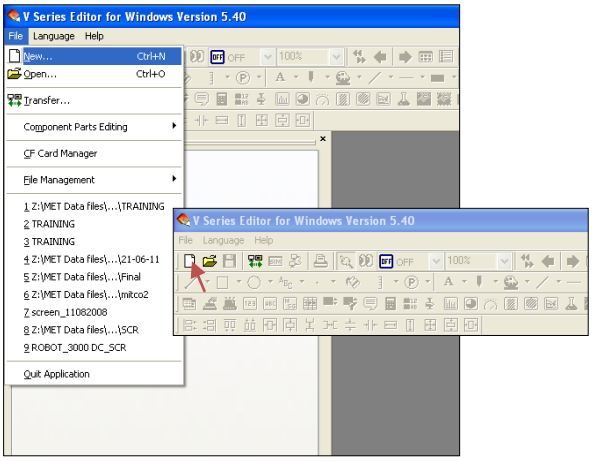
- As a start, let's begin by opening the V-SFTV5 software.
- Double-click the program icon. Or, click [Start] of Windows, and click [Programs], [V-SFTV5] then [V-SFTV5].



- The following V-SFT initial screen is displayed:

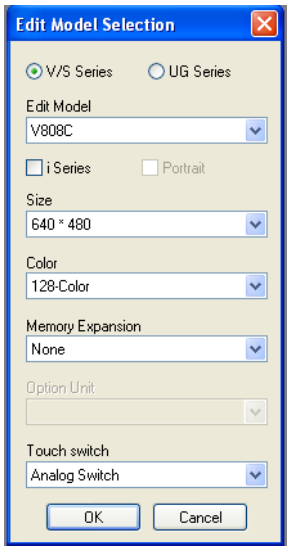


- Click the [New] icon in the tool bar, or select [New] from the [File] menu.

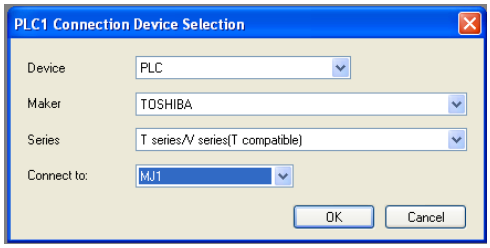


- The [Edit Model Selection] dialog is displayed.
- Under [Edit Model], choose the type of touch screen you would like to use in your project.
- For this tutorial we'll be using [V808C]
- From [Color] pull down menu, you can choose number of colors you need for you project and whether you need the blink function or not.

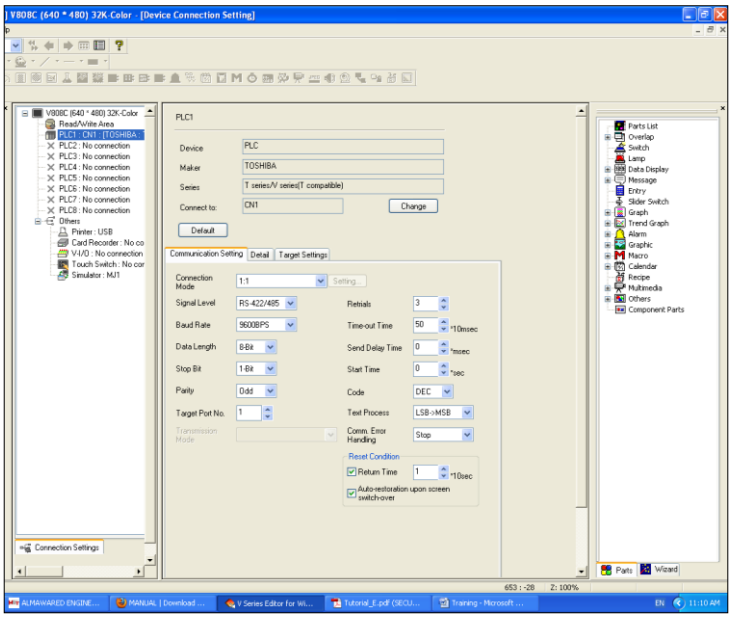
  Note: Blink is used sometimes in alarm messages to indicate importance.



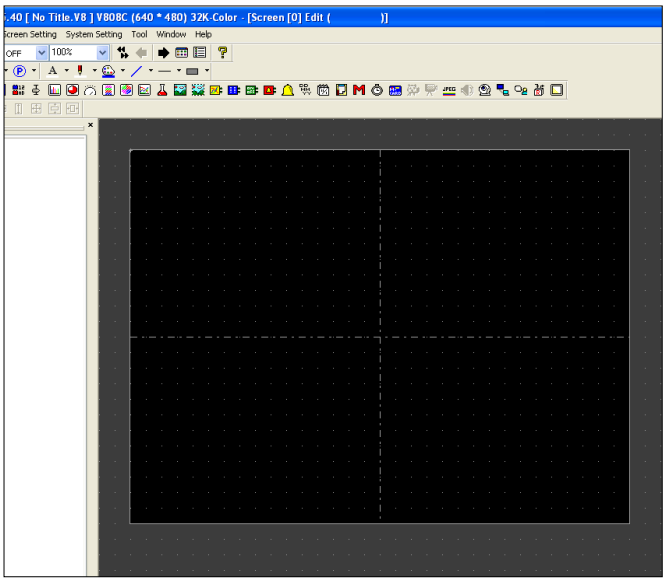
- Click [OK], you'll get another window where you should choose whether you'll connect the touch screen to a temperature controller/inverter or PLC. This will be under [Device].



- After choosing the device, choose [Maker] and [Series] of your device.
- Then choose the touch screen port you'll use to connect the device.
- For this tutorial we'll use TOSHIBA PLC as an example. So choose it then click [OK].
- You'll get the following window. This is the [Device Connecting Setting] that we'll discuss later.

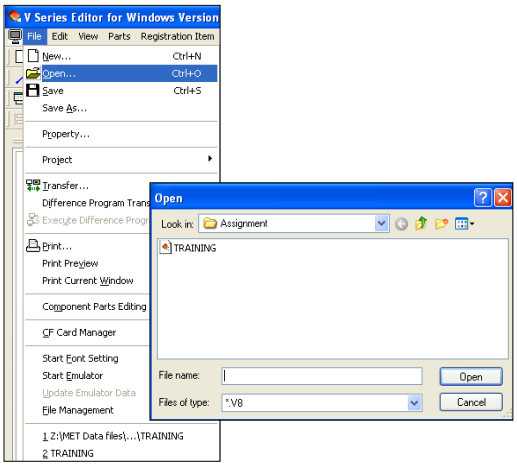


- You can close this window from the upper right [X] but take care not to close the whole program.
- You'll get a screen called [Screen [0] Edit]. This is the first screen in your application.



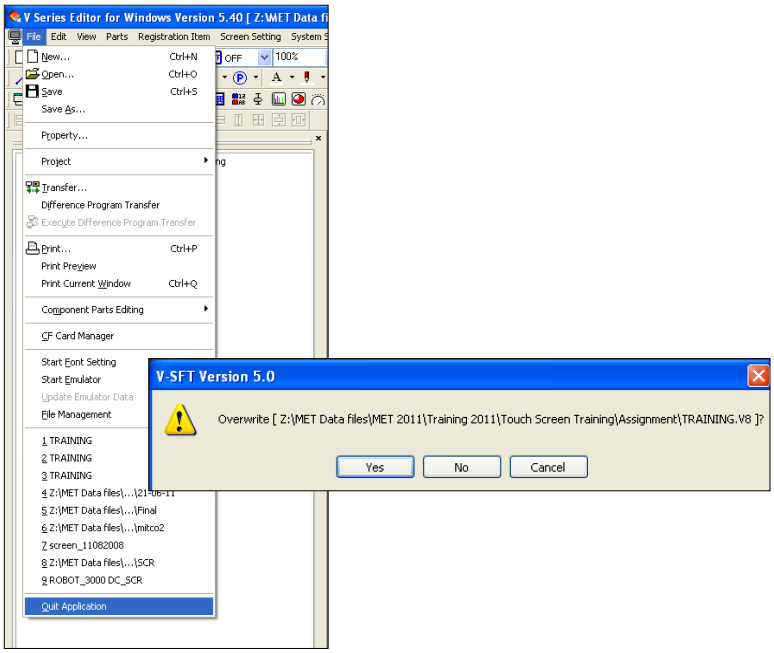
Opening a file:

- Click [Open] icon for [Open] in [File] menu.
- A dialog to select the file will be displayed. Select a screen data file to be edited then click [Open].



Closing a file:

- Click either [(Quit Application)] in [File] menu or [Close] button. The screen is then closed.
- If the modified data is not saved, the dialog appears to confirm whether the data will be saved or not. Click either [Yes] or [No] to close the program.



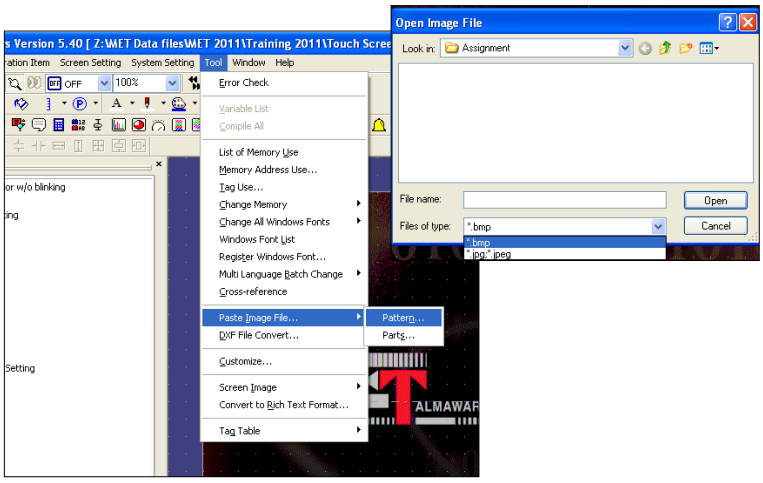
5.2. Creating a Screen Consisting of Characters and images

- You will create the following screen. This is the initial screen to be displayed at start-up.

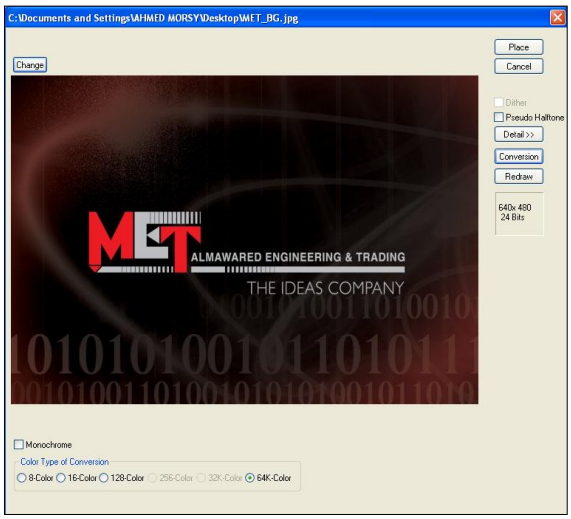
- The sentence “Welcome to Hakko Touch Screen …” is a text. Everything else is just one image (or bitmap) that we’ll use as a background.

Adding Images:

- Let's begin by adding the background image.
- Make sure that any image you choose doesn't exceed the size (resolution) of your screen which is (640 x 480). This depends on the screen model you selected.
- From [Tool] menu, select [Paste Image File]. Then click on [Pattern].
- You'll get the [Open Image File] Dialog.



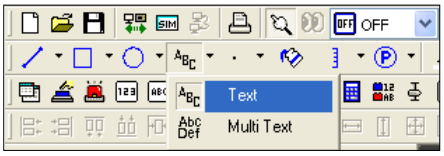
- Choose any image of types (bmp, jpg) with a resolution of (640X480). You'll find the image used in this tutorial with the name (MET_BG) in the CD attached with this tutorial.
- After selecting your photo, you'll get this window:



- Press [Place], the window will disappear. And you'll get a transparent rectangle showing where you want to place your image. Click wherever you want to place your image.

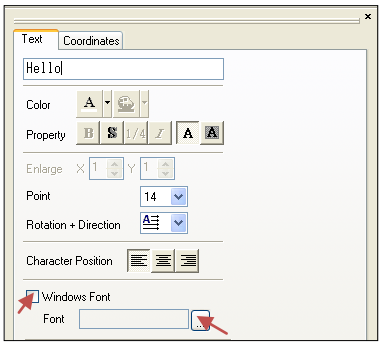
Adding Text:

- Now we want to place some text in our screen.



- Go to [Draw] toolbar. If you can't see that toolbar, go to [View]/Toolbar/Draw and check it.

- Click on [ABC] button to write a word. To write a paragraph, click on the small arrow beside it and choose [Abc, Def]. This will give you a multi-line text.



- Click where you want to add text in the screen and start typing.

- Once done, you can click on the text you just wrote to adjust it.

- You can easily modify text from the text box.

- You can also change font color and background color.

- Beside [Property], click on [**B**] to make text **bold**, or [S] to make a shadow for the text you wrote or click on [A] to make the background transparent in case you want to place the text on an image like our case here.

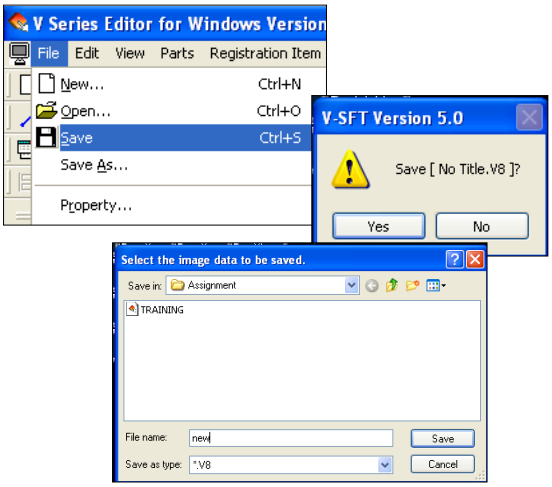
- You can change text size by adjusting the number in [Point].

- You can also change the text direction and alignment.

- In case you want to change font to one of the windows fonts, check [Windows Font] and choose your font.

Saving the screen data:

- Click the [Save] icon in the tool bar, or select [Save] from the [File] menu.
- You'll get a dialog confirming whether you want to save or not, select [Yes].



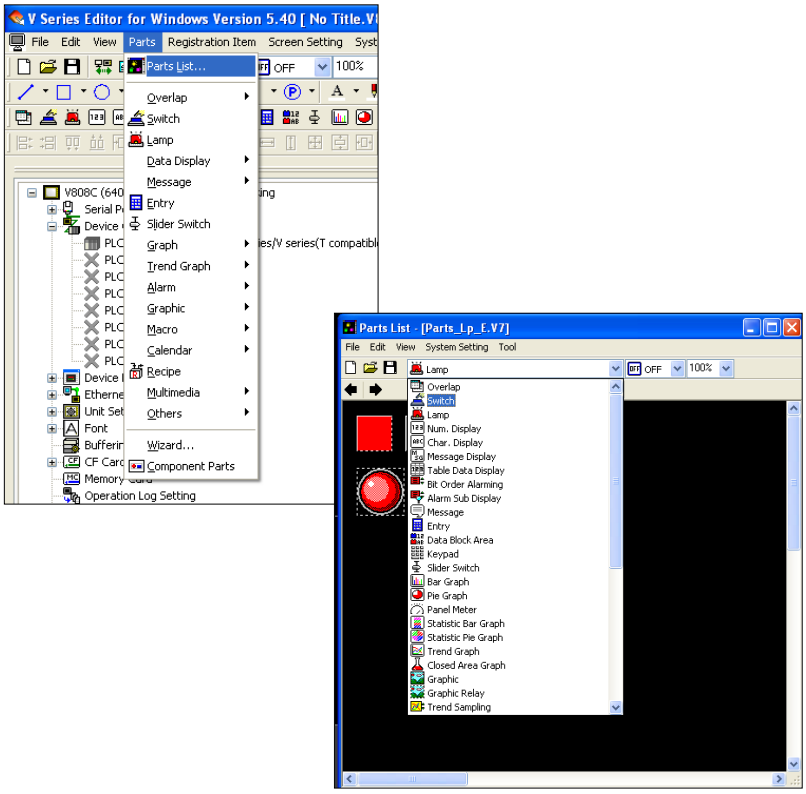
- You'll get a window titled [Select the image data to be saved].
- Write a name for you application in [File name] field, then click [Save].

5.3. Creating Switches

A switch has one of the following functions:

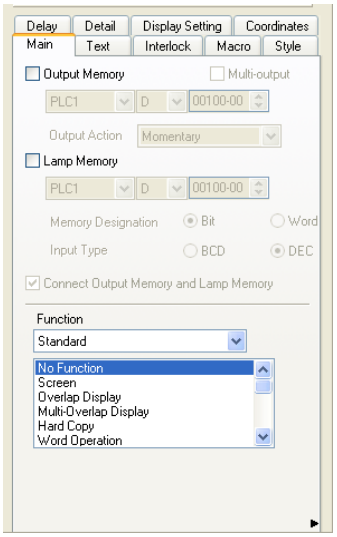
1. Changing the screen display (go from screen to another)
2. Showing an overlap display
3. Performing operations, and writing results into a PLC memory
4. Printing the image displayed
5. Turning a memory bit ON and showing an overlap display at the same time
6. Doing any extra function using script (Macro)

- For placing a switch in your screen, go to [*Catalog View*] or [Parts] pull down menu and choose [Parts List].



- Here in [Parts List], you'll find all kinds of parts you can place in your project.
- Choose [Switch] and select any type of switches you'd like. 2D and 3D switches are available depending on the screen model you're working on.
- Click on any switch and drag it to your current screen.
- You can scale the switch by clicking and holding one of the corners and dragging. Hold [Shift] to maintain the aspect ratio between length and height of the switch.

- Click on the switch to view its details window.
- **Main:** Check [Output Memory] if you want this switch to turn a bit ON or OFF in PLC, then write this Memory address.



- If you write R00005 for example: this means that when the switch is pressed, bit number 5 in the word RW000 will be ON.

Bit.	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
Value	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0

- You can choose the type of output action that controls how the switch will behave:

Set: When the switch is pressed, the specified bit is set (ON).
Reset: When the switch is pressed, the specified bit is reset (OFF).
Momentary (Momentary W): When the switch is held down, the specified bit is set (ON), and when it is released, it is reset (OFF).
Alternate: Each time the switch is pressed; the specified bit is alternately set (ON) and reset (OFF).

- You can also assign a lamp Memory so that when this bit in ON, the switch changes color.
- Other functions are available for switches under [Function],

like [Screen] to go to another screen or [Multi-Overlap Display] to open an overlap display once the switch is pressed.

- **Text:** is for adding text in ON state or OFF state of the switch. You can easily change font color, size, type and direction as mentioned before.

- **Macro:** you can write a macro script here that will be executed when switch is pressed ON or OFF.

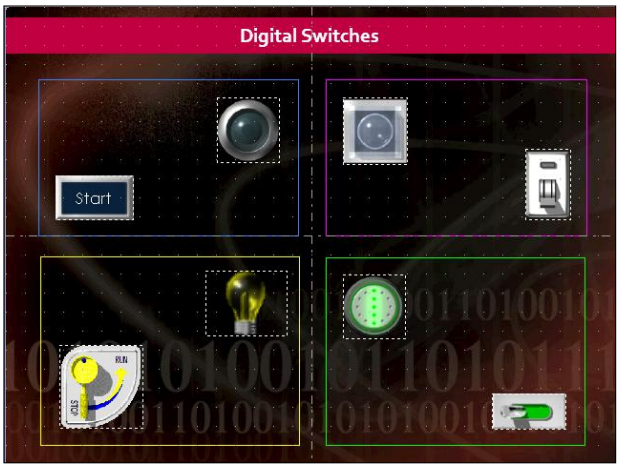
- **Style:** here you can change switch color, add more patterns other than OFF and ON or even change the switch itself.

- **Delay:** under this tab, you can specify an ON delay time. Now when the switch is pressed and kept pressed for the specified time, bit will become ON. Same goes for OFF delay time.

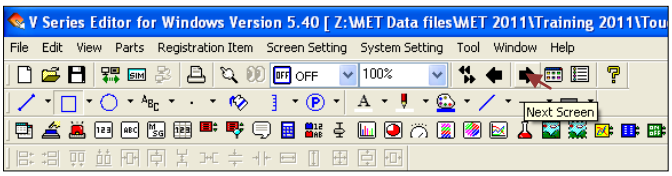
5.4. Creating Lamps

- Lamps are mainly used to indicate whether a bit inside a word in PLC is ON or OFF.

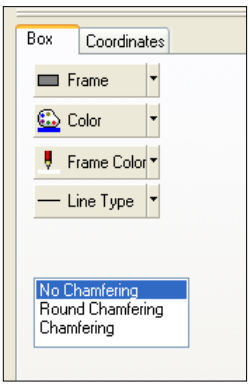
- In the end of this part, you should be able to create something similar to this screen:



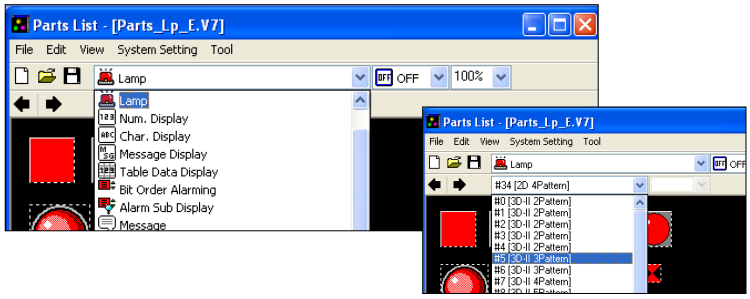
- If you're in Screen [0], Press [→] to move to Screen [1].



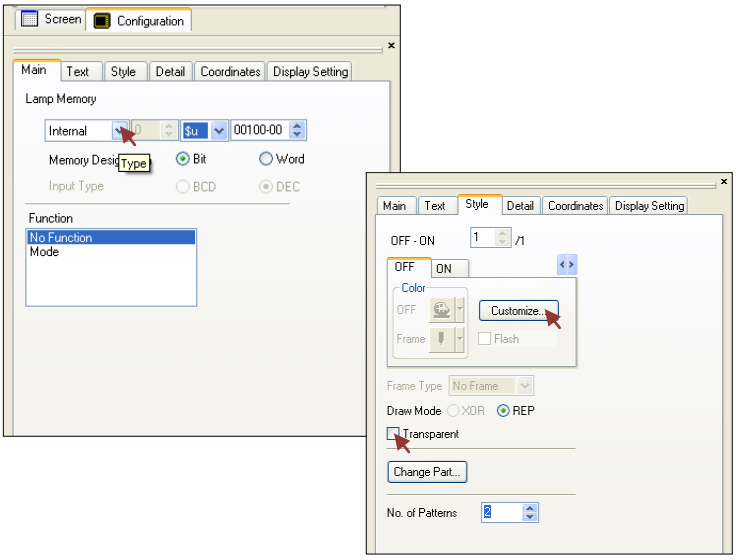
- Begin by adding a background to your screen as you learned before. This time we'll use the image (MET_BG_PLAIN) found in the training CD. It's the same as the one we used before, only without the logo and text.
- From the drawing toolbar, select [Rectangle].
- Click near the top of your screen, hold and drag to draw the red rectangle (page title bar) in the top.
- In the [details] window of the rectangle, you can choose whether you want to display a frame to the rectangle or no frame at all.
- You can adjust the rectangle color, frame color and Line type.



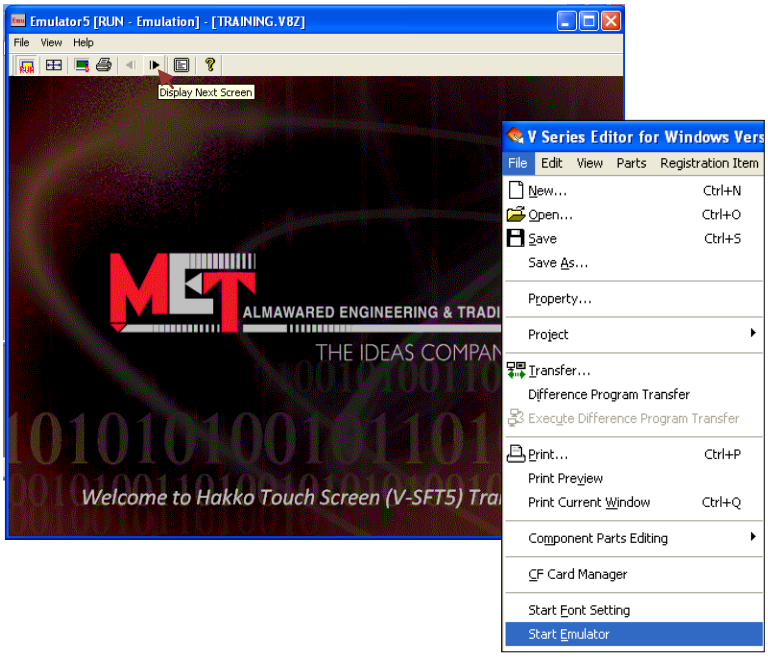
- Also you can make the rectangle corners chamfered and specify the chamfer radius.
- Add text to your screen on the rectangle you just created as its title, "Digital Switches".
- Create the four rectangles, in which we'll put the switches and lamps.
- Create four switches with PLC memories (R0001, R0002, R0003 and R0004) respectively.
- No it's time to add the lamps.
- From [Parts List] under [Parts] menu or in [Catalog View], choose [Lamps].
- There are many patterns from which you can choose different types. This goes for all parts, not only lamps.



- Choose one and drag it to your screen.
- Similarly, choose three other types of lamps and add them.
- Click on any lamp to open its [details] window.
- Similar to the switch, you can specify a memory address for any lamp. Give the four lamps PLC memories from (R0001 to R0004).
- You can type any text on lamps in ON or OFF state. This is possible under [Text] Tab.
- You can also change lamp style by pressing [Customize].
- You can make the lamp [Transparent] during OFF state and then appear during ON state.



- From [File] menu, choose [Start Emulator] to test you application.
- The emulator will start from Screen [0], press the arrow [→] in the emulator window to go to Screen [1] that we have just created.
- Each switch you press should turn on/off a lamp.
- You can change switches output actions and test their behaviors through the emulator.

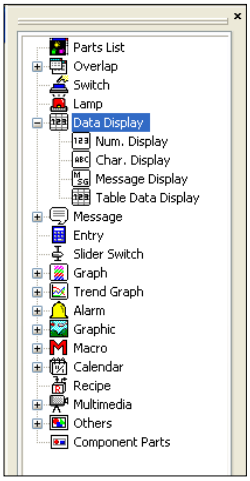


5.5. Creating Data Display Parts

Used for reading values from memory addresses or writing data to them.

You can find data displays in [Parts List] or [Catalog View].
There are four types of Data Displays.

1. **Numerical Display:** reads or writes an analog (numerical) value.
2. **Character Display:** reads or writes a string (characters or letters).



For the two above, through [details] window:

Under [Main] tab:

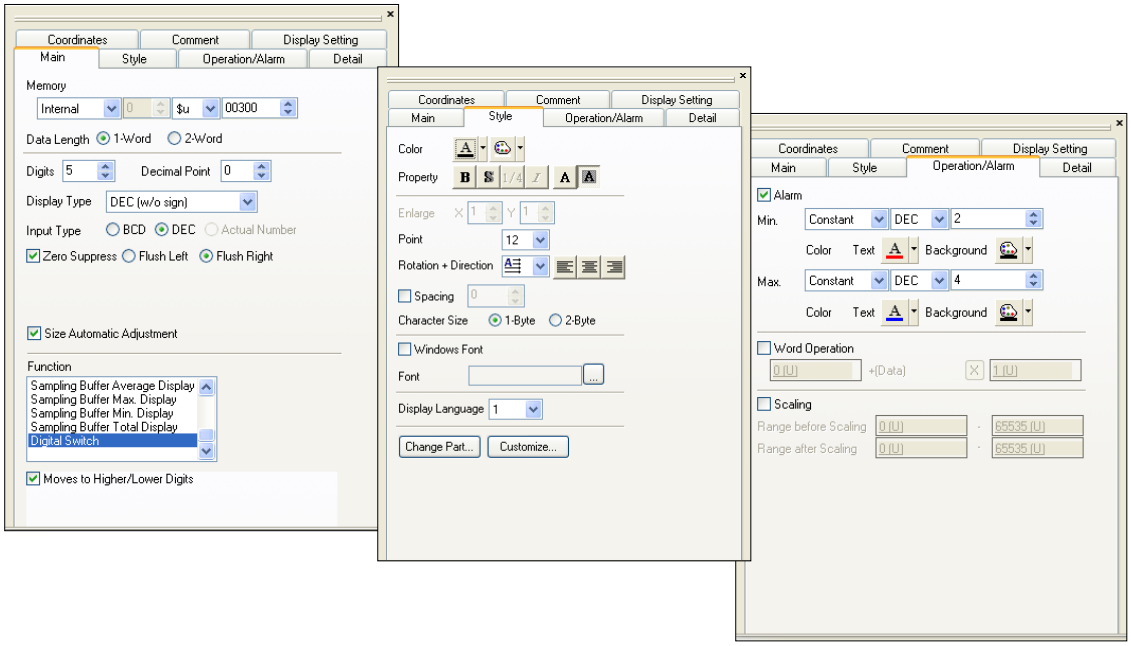
- You can change the PLC memory address which you want to read data or write data in.
- You can change the number of digits and the place of the decimal point.
- You can choose if you want the number to have a negative sign through (Display Type).
- You can also assign functions to the display.

Under [Style] tab:

- You can change font, size, color and direction.

Under [Operation/Alarm] tab:

- You can apply minimum and maximum limits (numerical display only), so that the numerical display won't accept any value out of these limits.



3. **Message Display:** displays a registered message for every bit that goes ON in the memory address you specify.

- Messages can be written under [Main] tab/ Message Edit/Open.

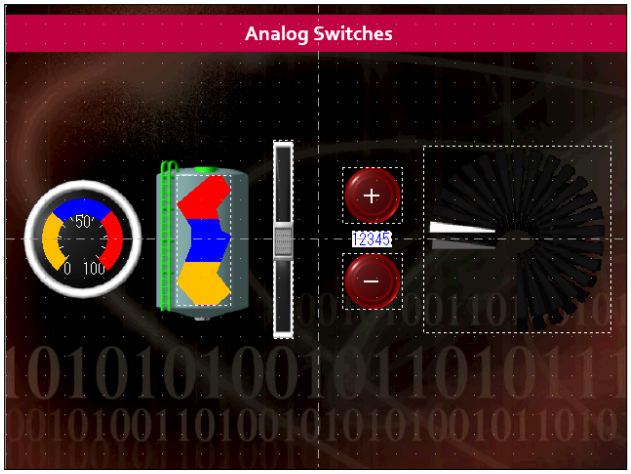
- 00000: This message will be displayed when value in this memory address is 0

- 00001: This message will be displayed when value in this memory address is 1 … and so on.

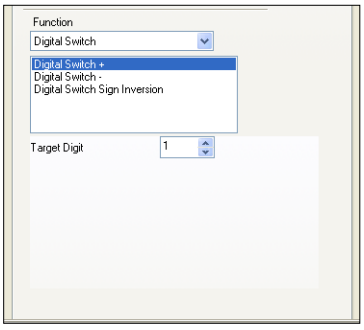
We'll go through Messages later in this tutorial.

4. **Table Display:** just like Numerical Display but in a table form. You can specify number of columns and rows under [Main] tab.

- Now, time for practice. For this section and the following two sections, we want to create this screen as Screen [2]:



- We'll start by adding the numerical display and the (+) and (-) switches. The rest we'll make in the following two sections (Sliders, Bars and Meters).
- Drag a numerical display from [Catalog View].
- Give the numerical display a memory (D0002) for example.
- Let the display type be unsigned with 1 digit and no decimal points.
- Give the numerical display the function [Digital Switch].

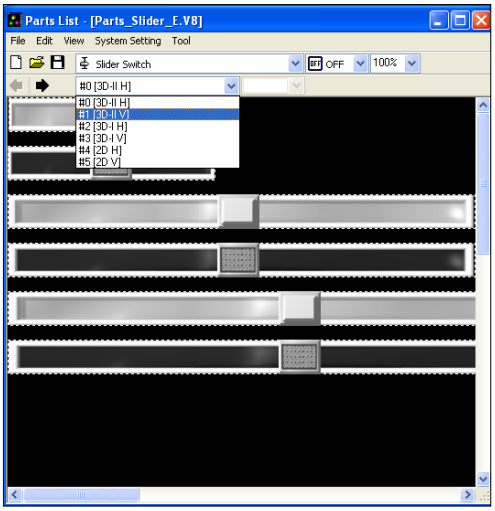


- Adjust minimum value to (0) and maximum value to (6)
- Now drag two switches to you screen, writing (+) and (-) to them respectively.
- Give the (+) switch the function [Digital Switch/Digital Switch +]and Give the (-) switch the function [Digital Switch/Digital Switch -]
- Test the screen using emulator.
- The number in the display should increase or decrease when the switches are pressed.

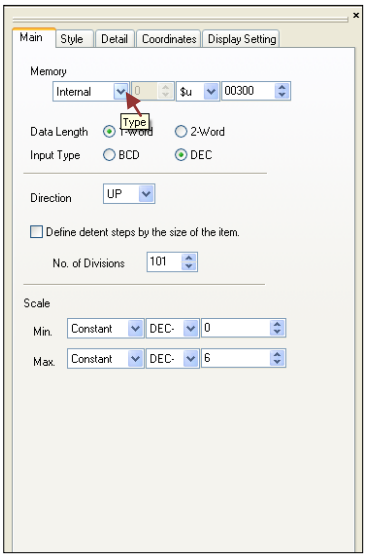
5.6. Creating Sliders

Suppose you need to increase an analog value like motor speed for example and you don't want to keep pressing on a switch or press it several times. That is where sliders come in handy.

- Here we'll add a slider to our Screen [2]
- Drag a slider from [Parts List], make sure it's a vertical slider.
- Click on it to show its details.



Under [Main] tab:



- You can specify the memory address that this slider will change its analog value.
- For this tutorial, write the same memory address of the numerical display we placed in Screen [2] which is (D0002).
- You can choose the direction where the value increases (Right, Left, Up or Down). It will be [Up] in our case.
- You specify the number of divisions of the slider thus its sensitivity or check [Define detent steps ….] to allow the

whole scale of the slider to vary depending on the user's touch.

In our case, we'll leave it unchecked and give it a number of (7)

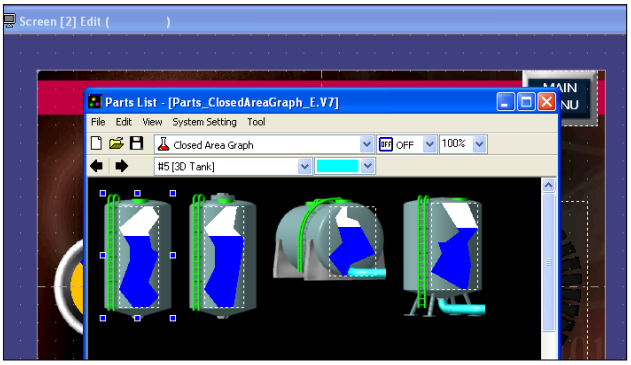
- For minimum and maximum values, we'll write 0 and 6 respectively.

Under [Style] tab:

- You can change colors of the rail and the slide but this is only applicable for 2D sliders.
- Test your screen using emulator, the numerical display should increase when you pull the slider up and decrease when you pull it down.

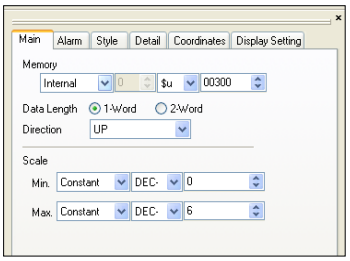
5.7. Creating Bars and Meters

For any analog value in your project you can use a bar, pie graph or panel meter to show this value in a nice form. Also you may use closed area graph if you have a tank getting filled with a liquid for example.



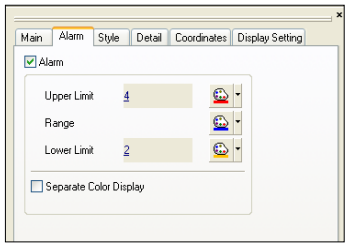
- Back to Screen [2], we want to add the tank you see in the picture.
- From [Parts List], in [Parts] menu. Choose [Closed Area Graph] from the pull down menu.
- Click on any tank shape you'd like and drag it to your screen.
- Click on the tank you chose to view its details.

Under [Main] tab:



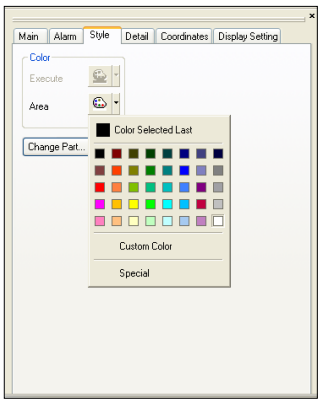
- You can specify memory address from which the analog value will be read. (PLC - D - 0002 in our case)
- Choose the direction in which this value increases (UP in our case).
- Specify minimum and maximum values (0 and 6 respectively in our case).

Under [Alarms] tab:



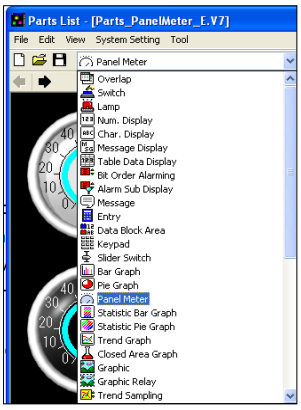
- Make sure [Alarm] is checked.
- Set Upper Limit and Lower Limit values and colors. (In our case, 4 and red for Upper limit, 2 and yellow for Lower Limit).
- These limits can be constant numbers or values from PLC memory addresses.
- Specify a color for Range value (Blue for example).

Under [Style] tab:



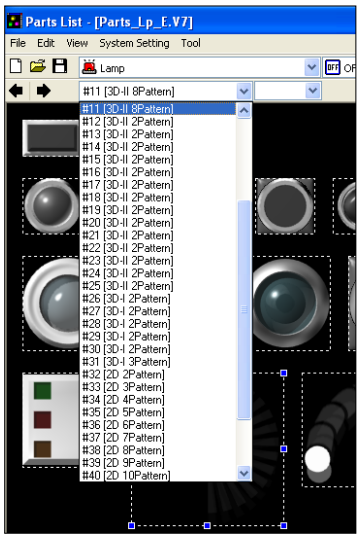
- You can change colors of graph and background.
- Graph color is deactivated because you checked [Alarms] and already gave a color for the range value.
- Give the background [Area] a color of white for example.

Now it's time to make the panel meter that was on the left, remember?



- Go to [Parts List] and choose [Panel Meters] from the pull down menu.
- Click and drag it to your screen.
- Select it to view its details.
- Panel meters are exactly the same as closed area graphs except you can add marks at minimum and maximum points under [Alarm] tab and give them a color.
- Give it the same configuration as the tank you created exactly.
- For the multi-pattern lamp on the right: open [Parts List], choose [Lamp] then from patterns pull down menu choose:

  #11(3D-II 8 Pattern)



- Choose the lamp used in the training and drag it to your screen.
- Click it to view its details.
- Give it the same memory address we use [D0002]
- No open your emulator and test your application.

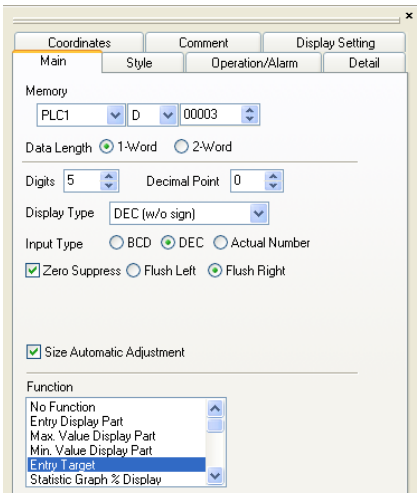
- Using the digital switches or the slider, you should be able to increase or decrease the numerical display, the tank level, the panel meter indicator and the multi-pattern lamp.

5.8. Changing PLC Memory Data through Keypad

Used for entering characters (numbers or letters) into a memory. Keypads can be put in screens or in overlaps depending on the space available in the screen.
In this part, you'll be able to create this screen which will allow you to enter data into a memory in PLC, which is used a lot in applications.

- You should have no problem now moving to Screen [3], adding the background image, drawing the rectangle and writing the text. Remember you can always copy repeated things like backgrounds and shapes and paste them wherever you want. By default they will be pasted at the same position they were originally at.

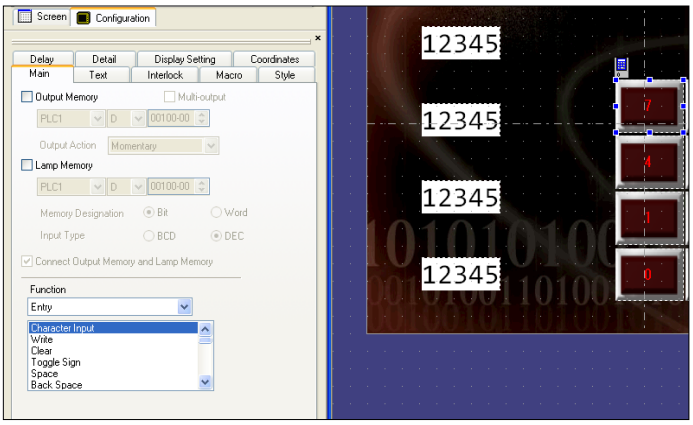
- Start by creating the four numerical displays and the one numerical display above the keypad as you learned before.



- Give the four numerical displays memory addresses from D0003 to D0006.

- This time we'll assign the four numerical displays the function [Entry Target] because we want to write data into them.

- The numerical display above the keypad should be given the function [Entry Display Part] because we only want it to display the number we'll write by the keypad.
- Now go to [Parts List] and choose [Keypad].
- View the different patterns and select any keypad but make sure it has UP and DW (Up and Down) buttons.
- Drag the keypad to your screen.
- If you click any of the keypad buttons to view its details, you'll see that it's a normal switch given the function [Character Input].



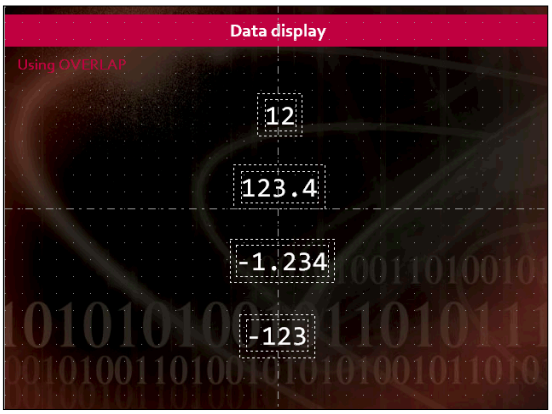
- You can change the color of the text and the style of the switches like any normal switches.
- Now that you're finished, you can test your application using emulator.
- Use arrow key [→] in the emulator window to go to screen [3].
- Click on numbers in the keypad then press enter [ENT], the number will be saved to one of the numerical displays.
- Use Up [UP] and Down [DW] buttons to choose which numerical display you want to enter data in.
- Also notice the behavior of the numerical display above the keypad, it only shows what you're typing.

5.9. Displaying Keypad Only When Necessary [Multi-Overlap]

Overlaps are used for displaying a screen whenever a switch, data display is pressed or a memory from PLC goes ON.
Each screen has 3 overlap screens. So if we have 1024 screens in Hakko touch screens, overlap function will give us a total number of 3072 screens which is more than enough for any application no matter how big it is.

So how do we create an overlap?

- Go to Screen [4] using [→] arrow.
- By now you should be able to create this page:

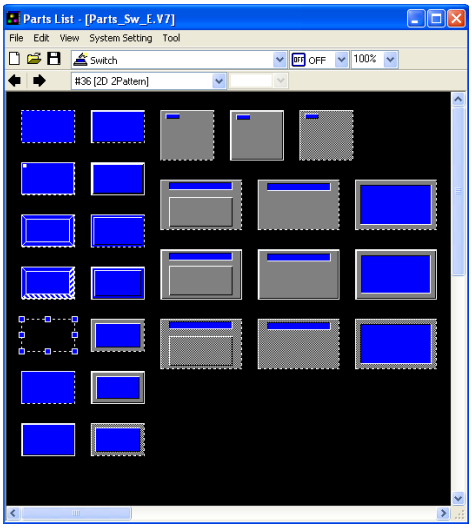


- 1st numerical display has a memory address of (D0007), 2 digits, 0 decimal points, unsigned, minimum value (0) and maximum value (99).
- 2nd numerical display has a memory address of (D0008), 4 digits, 1 decimal point, unsigned, minimum value (0) and maximum value (999).
- 3rd numerical display has a memory address of (D0009), 4 digits, 0 decimal points, signed, minimum value (-9) and maximum value (9).
- 4th numerical display has a memory address (D0010), 3 digits, 0 decimal points, signed, minimum value (-999) and maximum value (999).
- All four numerical displays should be given the function (Entry Target).

- Since numerical displays can't be used as switches to open overlap screens, we will add a transparent switch above each numerical display. When each of these switches is pressed, it will open the overlap screen.

- From [Parts List], in [Switch] patterns, choose pattern group number:

  #36 [2D 2Pattern]

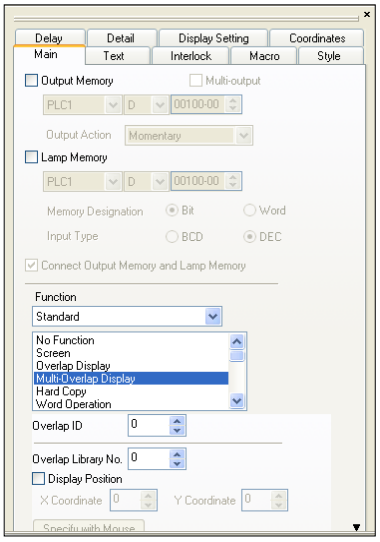


- Select the transparent switch, click it and drag it to your screen.

- Click on the switch to view its details.

- Give the switch the function [Multi-Overlap Display].

- Write (0) in the field [Overlap Library No.], this is the overlap screen number that this switch will go to when pressed.



- Make sure [Display Position] is unchecked.

- Copy the switch, paste it three times and place each copy on a different numerical display.

- One important thing:

  From [*Catalog View*]: Under [Overlap], drag the [Multi-Overlap] icon  to your screen and put it anywhere.

  It will not appear in your application anyway. This informs the software that there's an overlap in this screen.

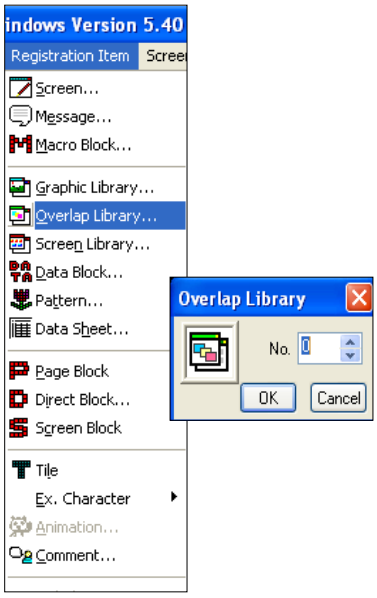
- Now let's make the overlap screen itself.

- From [Registration Item] menu, choose [Overlap Library].

- In [Overlap Library] popup window, write the number you choose for the switch which is in our case "0".

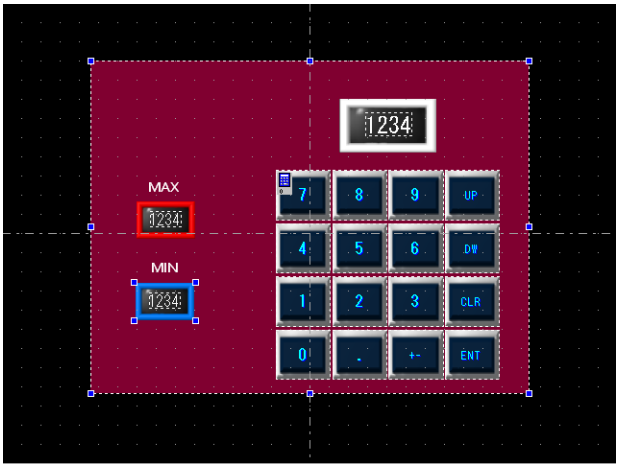
- Now you have the overlap screen open.

- From [*Catalog View*]: Under [Overlap], drag [Normal Overlap] to your screen. This is the actual part of overlap screen that will appear.



- In this overlap, we want to add three numerical displays and one keypad.

- This is the overlap screen that we want to create:

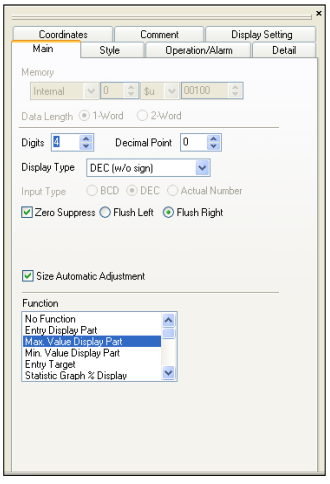


- It should be easy for you now to create every item in this overlap screen.

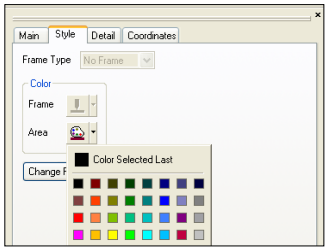
- The numerical display above the keypad will be given the function [Entry Display Part].

- The numerical display below “MAX” will be given the function [Max. Value Display Part].

- The numerical display below “MIN” will be given the function [Min. Value Display Part].



- In case you wanted to change the overlap color, you can easily do that in its details window under [Style] tab.

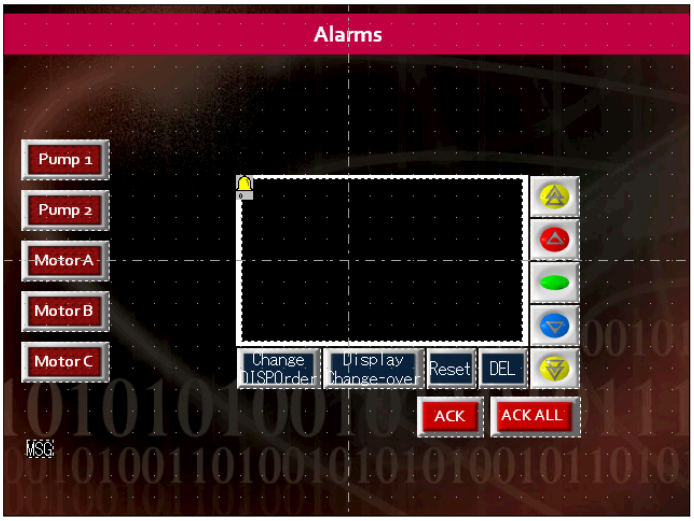


- Test your screen in emulator.
- You should be able to click on each numerical display and change its value through the overlap depending on the minimum and maximum values.

5.10. Displaying Messages

In this section and the following section, you'll learn how to make messages appear on your screen due to any change of a memory address in PLC and how to create an alarm history display to track current and previous alarms.

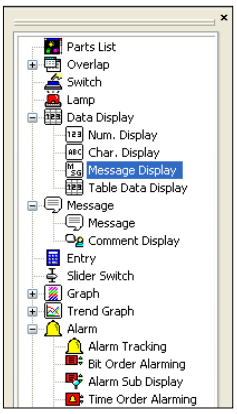
By the end of these two sections, you will have created this screen:



- Start by adding the five switches on the left.
- Give the switches names and memory addresses like you learned before:

  Pump 1 : R0020
  Pump 2 : R0021
  Motor A: R0022
  Motor B: R0023
  Motor C: R0024

- Let's create the message display that will display a message for each switch we press.



- Go to [Catalog View] on the right of your project and select [Message Display] from the [Data Display].

- Drag it to your screen and click it to view its details.
- Under [Style] tab, you can change text color, size and font like any other text.
- Under [Main] tab, specify the memory address that you'll choose.
- Now we'll stop to explain how messages work:

You specify a WORD not a BIT.

When this word contains a certain number it will display a corresponding message. This means for example if switch [Pump 1: R20] is ON and switch [Pump 2: R21] is ON, a value of [3] will be in the word [RW2] because this word has bits from [R20] to [R29] and from [R2A] to [R2F].

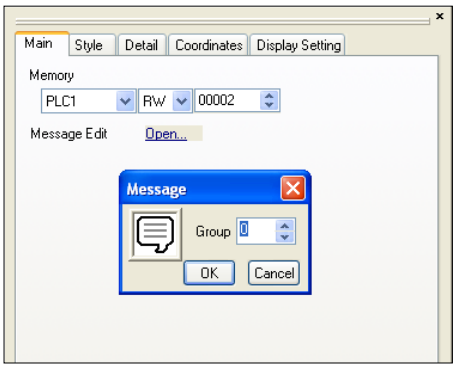
R2F	R2E	R2D	R2C	R2B	R2A	R29	R28	R27	R26	R25	R24	R23	R22	R21	R20
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1

The word RW2

So if you convert the binary number 11 to a decimal number, it will be 3. That's when the message corresponding to number 3 will appear.

- Under [Main] tab, write the memory address [RW0002].

- Beside [Message Edit], click [Open].



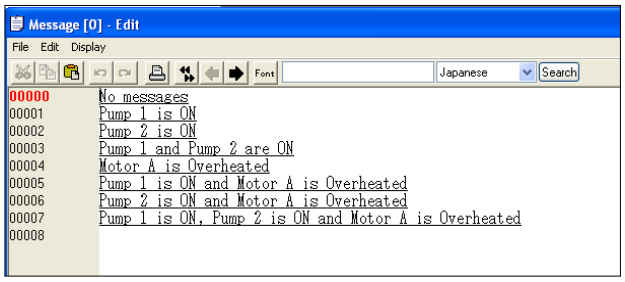
- A pop-up window will appear asking you for the group number.

- If you have more than one message display in your application, you should put the messages of each display in a different group.

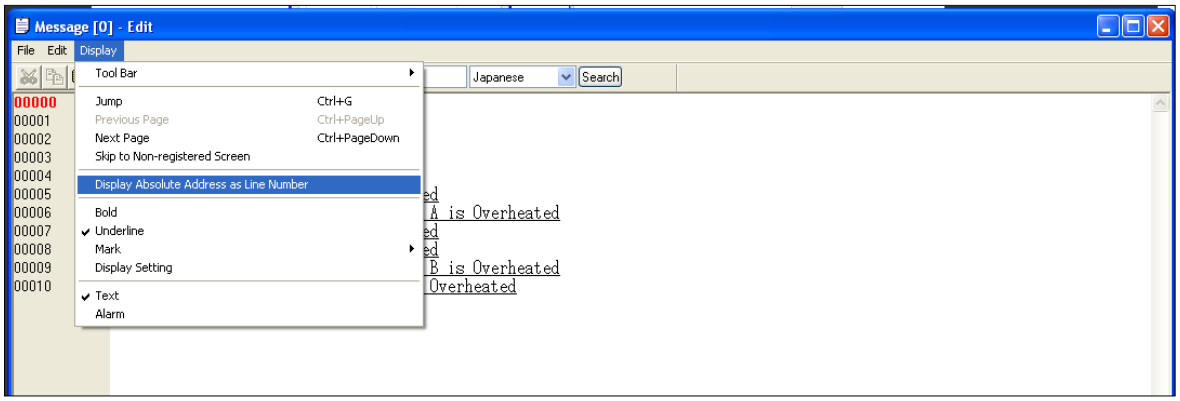
- We'll leave it [0] for now as it's the only display in our application. Click [OK].

- Now in the [Message [0] – Edit] window, write a message for each value in your word.

- For this tutorial, we'll only write 7 messages as long as you get the point.



- Click on [Display] menu; make sure [Display Absolute Address as Line Number] is not checked.



- Starting from line 0, write the following messages:

  00000: No messages
  00001: Pump 1 is ON
  00002: Pump 2 is ON
  00003: Pump 1 and Pump 2 are ON
  00004: Motor A is overheated
  00005: Pump 1 is ON and Motor A is overheated
  00006: Pump 2 is ON and Motor A is overheated
  00007: Pump 1 is ON, Pump 2 is ON and Motor A is overheated

- Close the window after you finish and open the emulator.

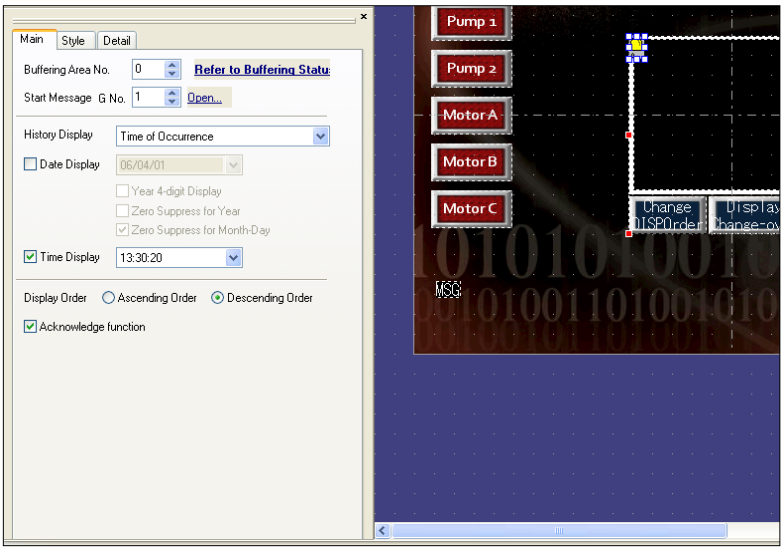
- Test your application.

- If you turn on switches [Pump 1] and [Pump 2], you should get a message saying “Pump 1 and Pump 2 are ON” because then the word RW002 will have a number of [3] inside it.

5.11. Displaying Error History [Alarm Tracking]

There are five types of Alarm Displays available in V-SFT5. But we'll be using only one type which is the most useful one of them that is [Alarm Tracking]. The settings of this Alarm Display are almost the same for other types, only easier.

- Go to [*Catalog View*]. Under [Alarm], you'll find [Alarm Tracking].
- Drag it to your screen.
- Other [Alarm Tracking] shapes are available in the [Parts List].
- Click on the "Bell" icon to view details of the alarm display.

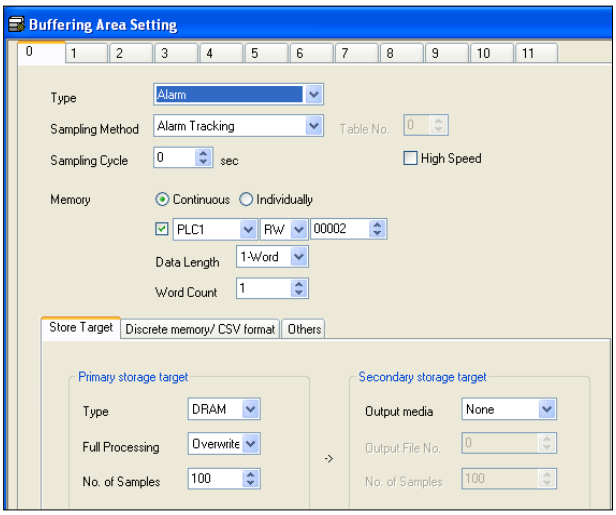


- Under [Main] tab, click [Refer to Buffering Status…].
- Make sure that the following settings are selected:

1. Type: Alarm
2. Sampling Method: Alarm Tracking
3. Memory: Continuous
4. PLC (Checked): Write the memory address of your alarms in PLC (RW02) in our case. Every bit in this memory address (word) will represent a separate alarm.

5. Word Count: If your alarms exceed 16 alarms (One Word), then choose [2] and so on.

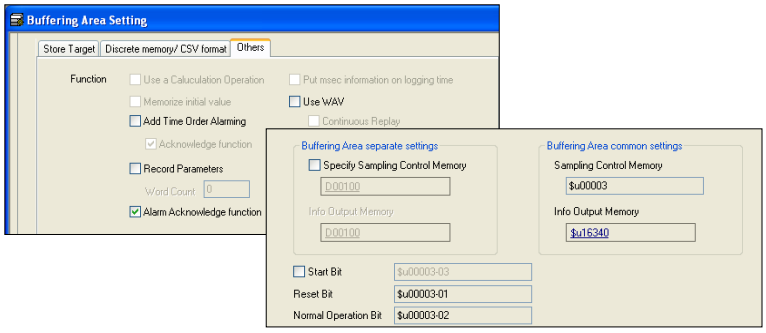
6. Under [Storage Target] tab, No. of Samples: Specify here the number of alarms that will be saved in your alarm display.



7. Under [Others] tab, Check [Alarm Acknowledge Function] to acknowledge alarms.

8. In the bottom of this window, Uncheck [Specify Sampling Control Memory].

9. Uncheck [Start Bit].

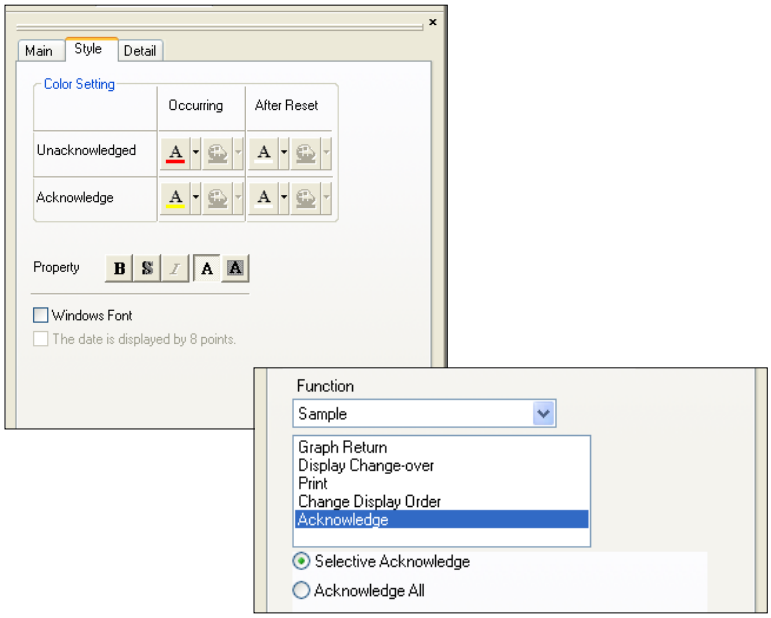


- Exit this window.

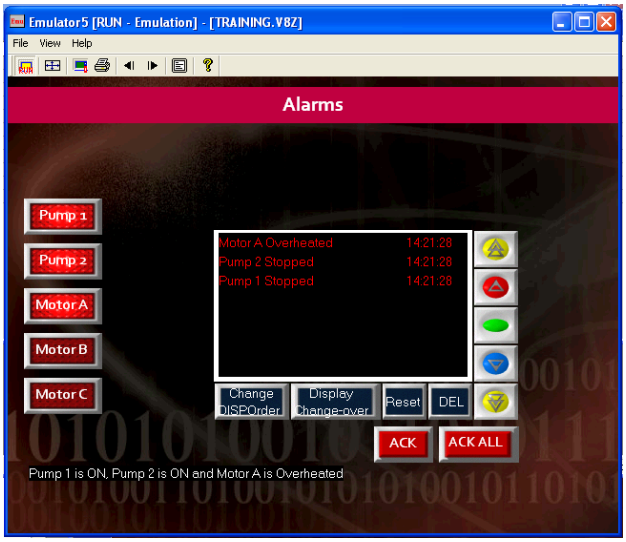
- Back to [Main] tab, beside [Start Message No.], Write [1] because we already used message group no. [0] with the message display.

- Click on [Open…].

- Here you should write the alarm messages that will appear with each bit in the memory address you wrote. Each line represents a bit.
- Back to the details, check [Acknowledge Function]
- You can change the display settings, display time or not.
- Under [Style] tab, you can assign colors to unacknowledged alarms (red in our case) and acknowledged alarms (yellow in our case)
- To make use of the acknowledge function, put two switches in your screen and assign to them [Acknowledge Function] from [Sample] function menu.
- Name the first switch [ACK] and check [Selective Acknowledge] in its details window. This switch will only acknowledge the alarm selected.
- Name the second switch [ACK ALL] and check [Acknowledge All] in its details window. This switch will acknowledge all alarms.

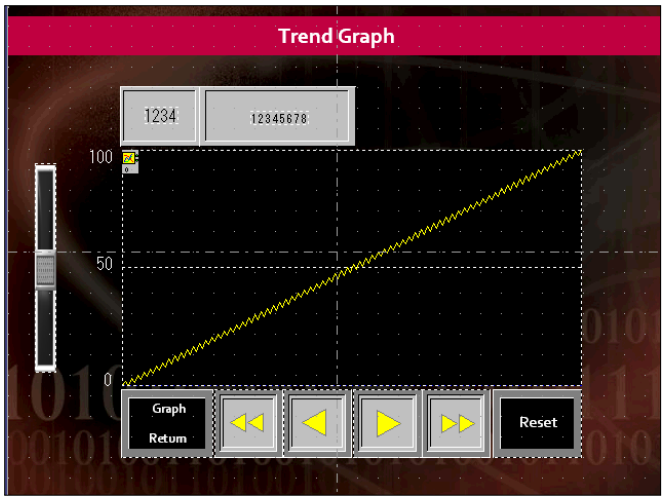


- Check your screen, Screen [5] in the emulator. The screen should look similar to this when pressing the first three switches:



5.12. Expressing Time-Varying Data in a Trend Graph

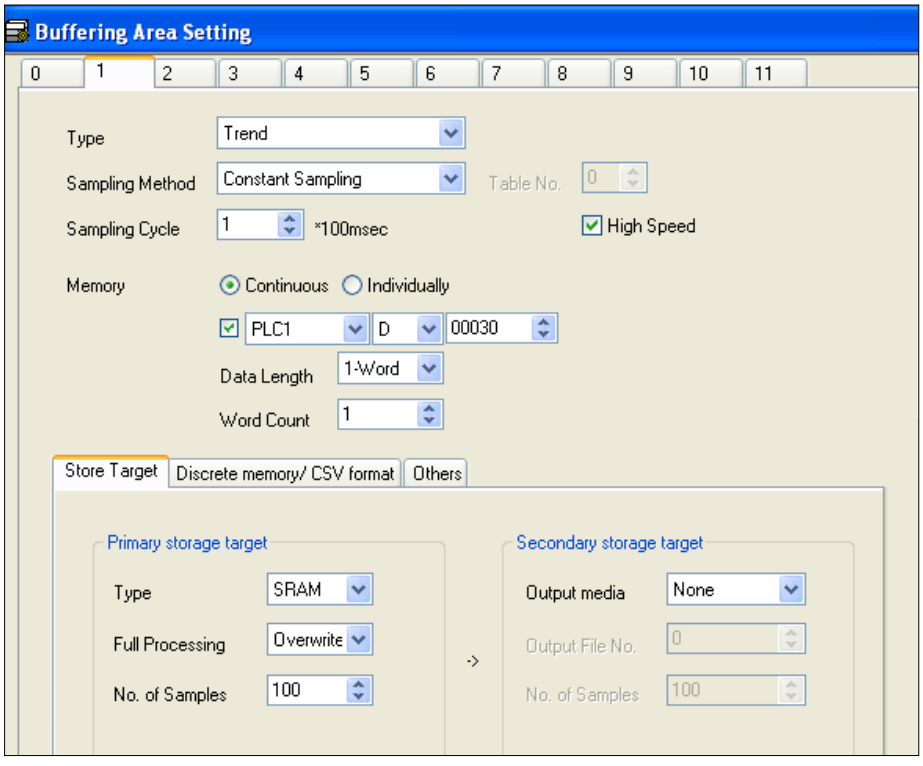
Trends are used to record changing data over time which can be used to monitor analog values like water level, voltage level or motor speed.
By the end of this section, you shall be able to create this screen:



- Go to [*Catalogue View*]. Under [Trend Graph], you'll find [Trend Sampling]. Drag it to you screen.
- Click again on the trend graph to configure it.
- Under [Main] tab, click on [Refer to Buffering Status…]
- We configured [Buffering Area No.0] already for the alarm display, remember? So this time, select the tab number 1 to configure the trend graph.
- Make sure that the following settings are selected:

1. Type: Trend
2. Sampling Method: Constant Sampling
3. Sampling Cycle: Period of graph (1 for graph to update every one second)

   In our case we'll check [High Speed] and write 1 in [Sampling Cycle], thus we get a 100 msec sampling time.
4. Memory: Continuous



5. PLC (Checked): Write the memory address (word) in PLC (D0030 in our case).

6. For Data Length and Word Count, choose 1 since we're only monitoring one PLC memory.

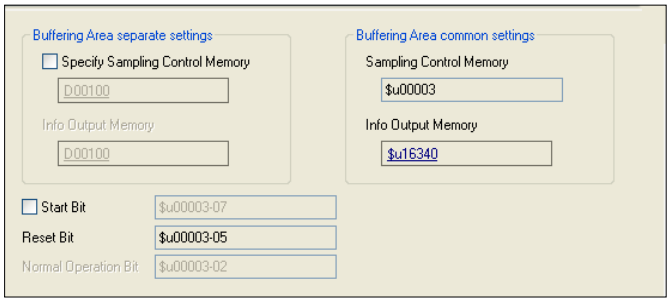
7. Under [Storage Target] tab:

- If you want to save the trend graph to memory choose [Type: SRAM].

- Full processing means when the trend graph samples end, what should the touch screen do? Stop sampling or overwrite the current graph samples (this has nothing to do with the trend graph history saved in the memory; they only overwrite the samples showing on the screen at the moment). We'll choose [Overwrite] for this case.

- No. of Samples: Specify here the number of trend graph points that will be shown in the trend graph display. Choose numbers above 20. We'll choose 100 samples for our case.

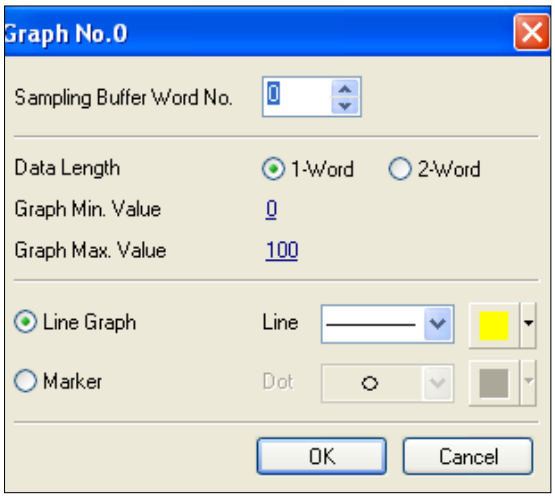
8. Roll down to the end of this page. Uncheck [Specify Sampling Control Memory] and [No. of Samples].



- Close this window and get back to the details window of the trend graph.

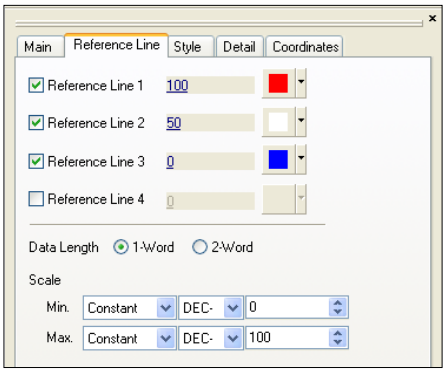
- We're still under [Main] tab. For [X Axis Data Points], write 150. This is the number of points in the X-axis.

- In [Graph Setting], check [No.0] and press [Edit].

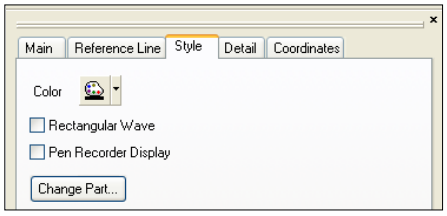


- Specify the [Graph Min. Value] and the [Graph Max. Value] as 0 and 100 respectively.
- Check [Line Graph] and choose line type and color. In our case, we'll choose the thin line with the color yellow.
- Press [OK].

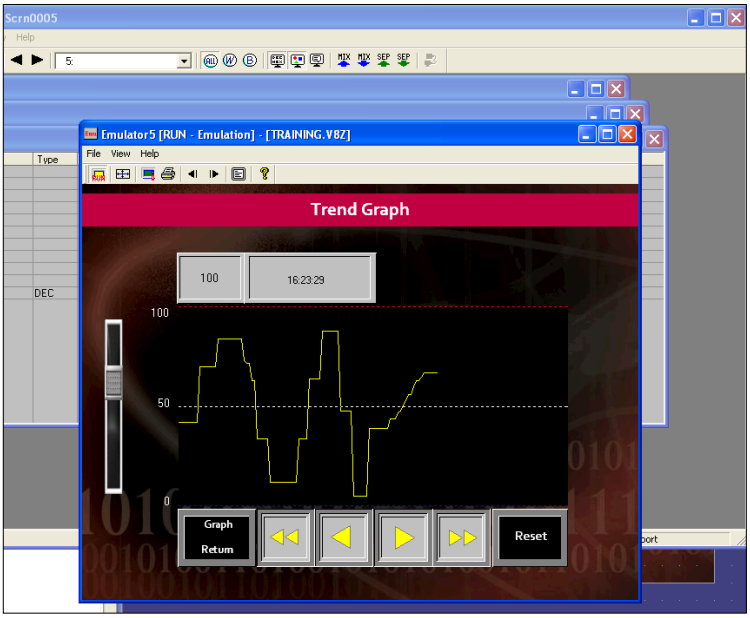
- Back to [Details] window of the trend graph.
- Under [Reference Line] tab, you specify reference lines at certain values and give them a definite color, just make sure you check it first.
- Also make sure the minimum and maximum values are right (0 and 100 respectively).



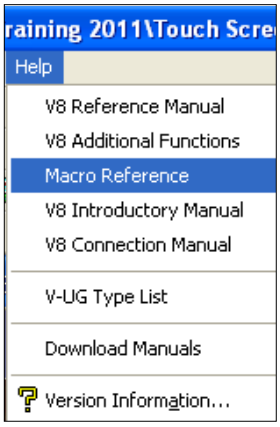
- Under [Style] tab, you can change the background color of the trend graph display (Black in our case).



- Leave [Rectangular Wave] and [Pen Recorder Display] unchecked cause we want a free shape graph.
- Add a slider switch to test your graph.
- Give it a minimum of 0 and a maximum of 100.
- Give the slider the same PLC memory as the graph (D0030).
- You can use text to add labels to your graph.
- You've just created Screen [6].
- Open your emulator and ~~text~~ your screen.
- The graph should change whenever you change the position of the slider.



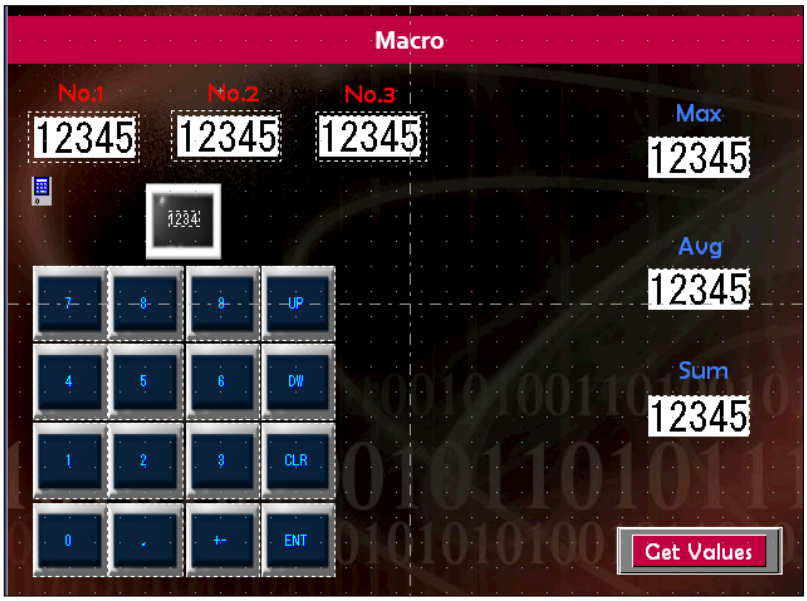
5.13. Macro



Any function that's not available in the functions menu of the switches, lamps or any parts in the screen can be programmed through writing a script in the macro edit window. This script can range from simple to very complex script depending on the functions needed.

In this section of training, we'll only cover a small part of macro functions. You can discover other macro functions through the [Macro Reference] installed with the [VSFT-V5] software, you'll find it in the [Help] menu.

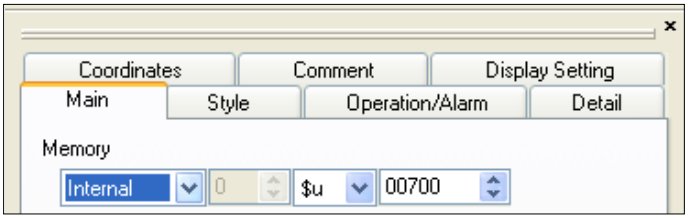
By the end of this training, you should be able to make this as you Screen [7]:



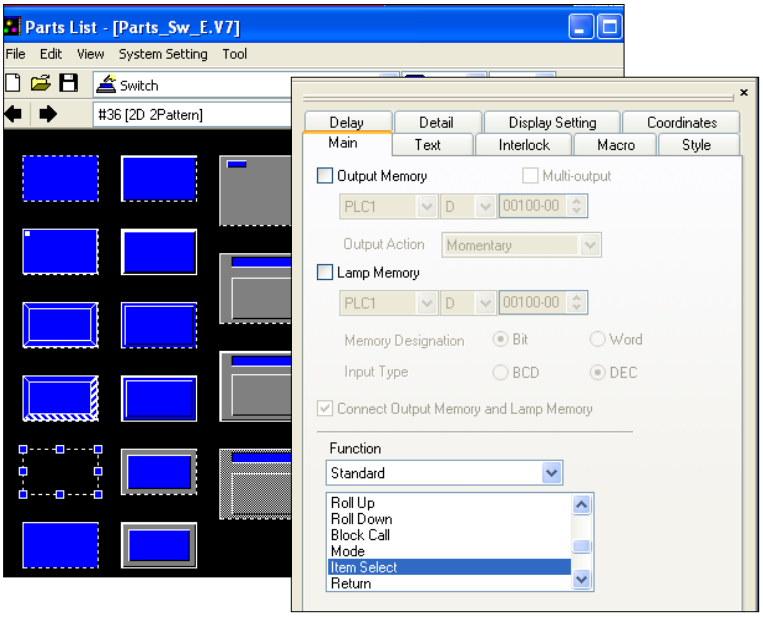
The user should enter 3 numbers [No.1, No.2 and No.3] and presses [Get Values] and their Maximum, Average and Sum will be calculated using a Macro script that we'll write.

- Begin by adding the title, the title bar, the keypad, the numerical displays, text and the [Get Values] switch.
- The three numerical displays under [No.1, No.2 and No.3] will be given the functions [Entry Target] and the internal memories $u700, $u701 and $u702 respectively.

- The [$u] memories are internal memories inside the touch screen itself.

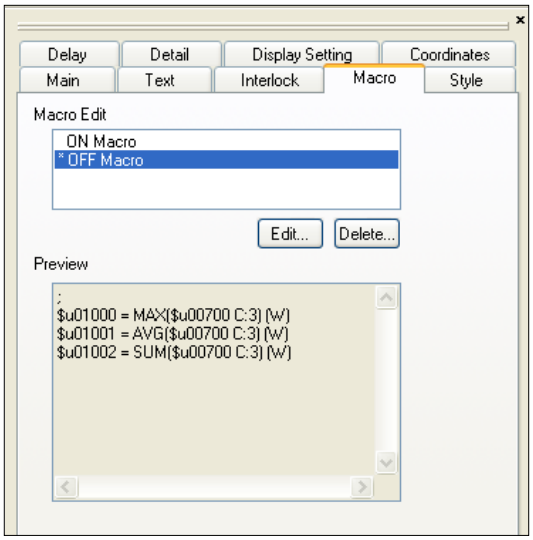


- Add transparent switches above these three numerical displays and give the switches the function [Item Select]. This will enable the user to choose which numerical display he wants to enter value in.



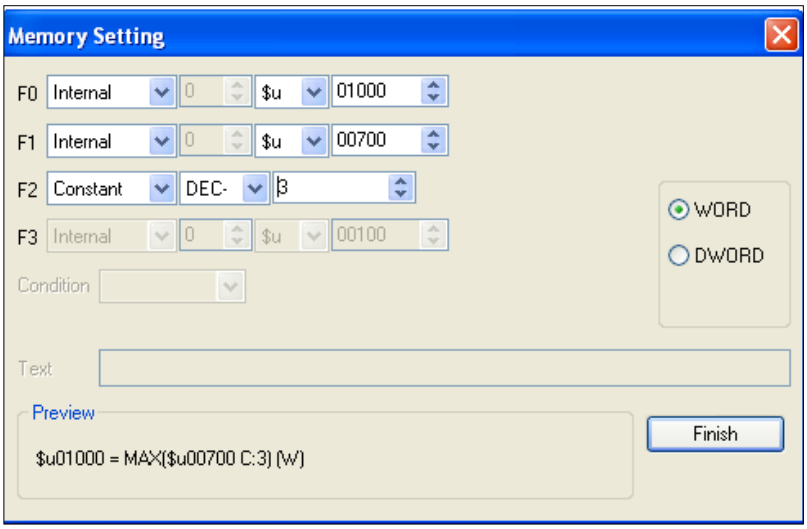
- For the three numerical displays under [Max, Avg and Sum], they will be given the internal memories $u1000, $u1001 and $u1002 respectively and no functions.

- Now comes the important part. Click on the [Get Values] switch you created to view its [Details] window.



- Under [Macro] tab: inside the [Macro Edit] box, you'll find [ON Macro] and [OFF Macro].
- Click [OFF Macro] for example and press the [Edit…] button. This way the script you'll write will be executed when you release the [Get Values] switch.
- The [Switch OFF Macro - Macro Editor] window will appear.
- Write MAX, a small help window will appear.
- Double click the word [MAX].

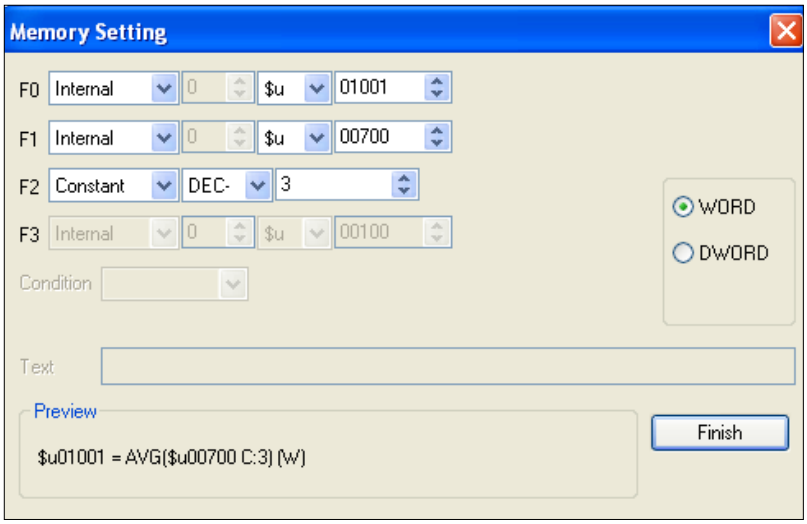
- The following window will appear:



- For F0, write the internal memory $u1000.
- For F1, write the internal memory $u700.
- For F2, write the constant 3
- Click [Finish] you'll get the first line in your script "$u01000 = MAX($u00700 C:3) (W)"
- This means that the maximum of the **3** memories starting **$u700** will be saved to the memory **$u1000.**

- After the line you just created, press [Enter] from the keyboard to move to the next line.
- Write [AVG], a small help window will appear.
- Double click the word [AVG].

- The following window will appear:



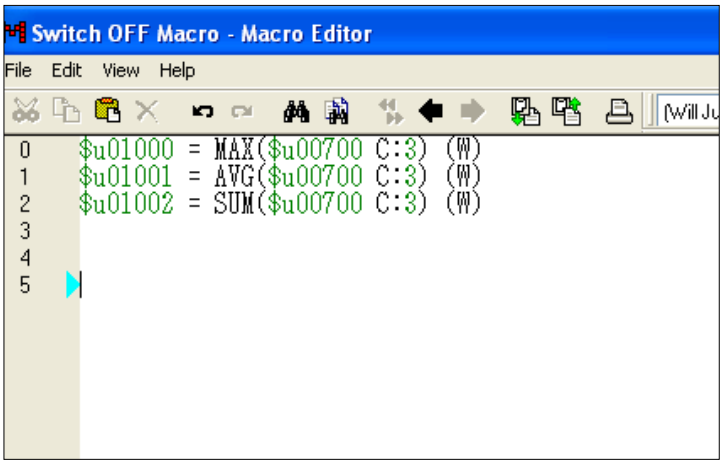
- For F0, write the internal memory $u1001.
- For F1, write the internal memory $u700.
- For F2, write the constant 3
- Click [Finish] you'll get the second line in your script "$u01001 = AVG($u00700 C:3) (W)"
- This means that the average of the **3** memories starting **$u700** will be saved to the memory **$u1001.**

- After the line you just created, press [Enter] from the keyboard to move to the third line.
- Write [SUM], a small help window will appear.
- Double click the word [SUM].

- For F0, write the internal memory $u1002.
- For F1, write the internal memory $u700.
- For F2, write the constant 3
- Click [Finish] you'll get the third line in your script "$u01002 = SUM($u00700 C:3) (W)"

- This means that the sum of the **3** memories starting **$u700** will be saved to the memory **$u1002.**

- Your script should look like this:

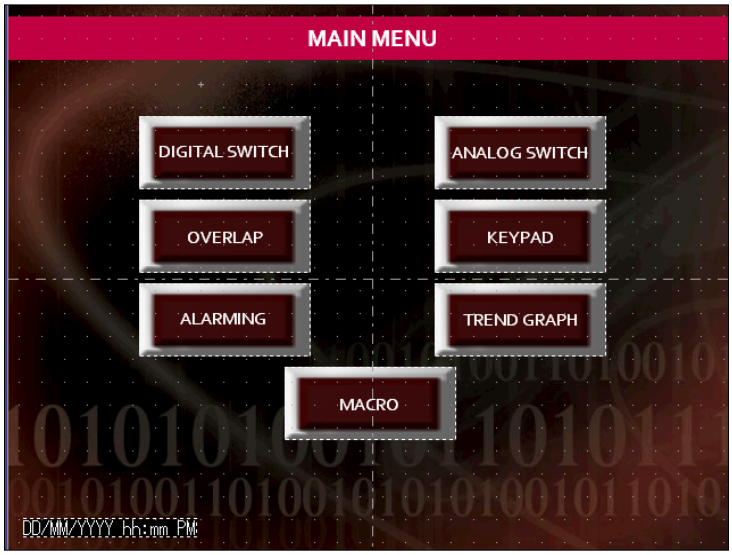


- Close the Macro Editor and test your application through the emulator.

5.14. Time Display/Calendar Display

In some applications, it would be nice to have date or time or even both displayed.

Go to Screen [8] that should look similar to this in the end of this section:



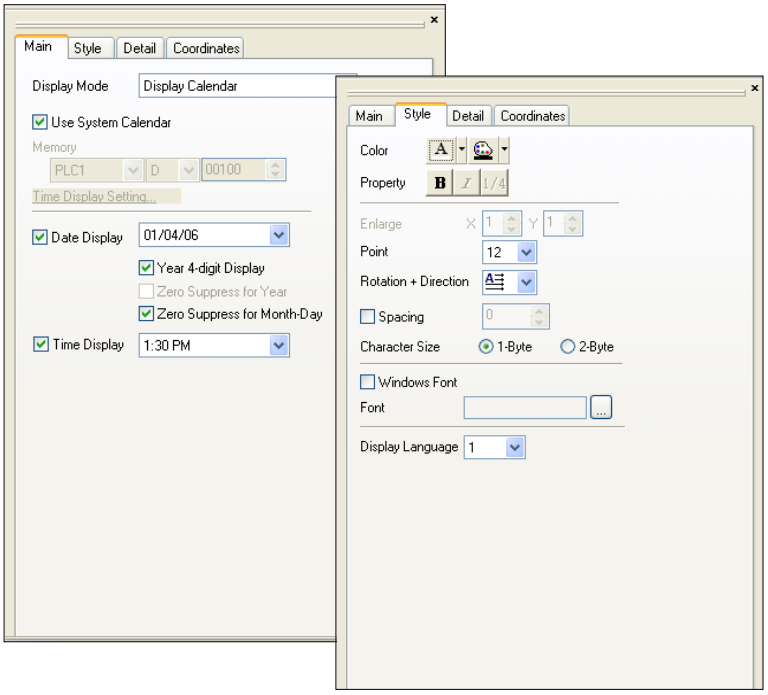
This screen will have the main switches that will go to all the screens we created throughout the tutorial plus a time display to show time and date.

- Create seven switches and give each switch a name and a screen to go to:

DIGITAL SWITCH	----------	Screen [1]
ANALOG SWITCH	----------	Screen [2]
OVERLAP	----------	Screen [3]
KEYPAD	----------	Screen [4]
ALARMING	----------	Screen [5]
TREND GRAPH	----------	Screen [6]
MACRO	----------	Screen [7]

- Screen [0] is the introductory screen, thus main menu won't refer to it.

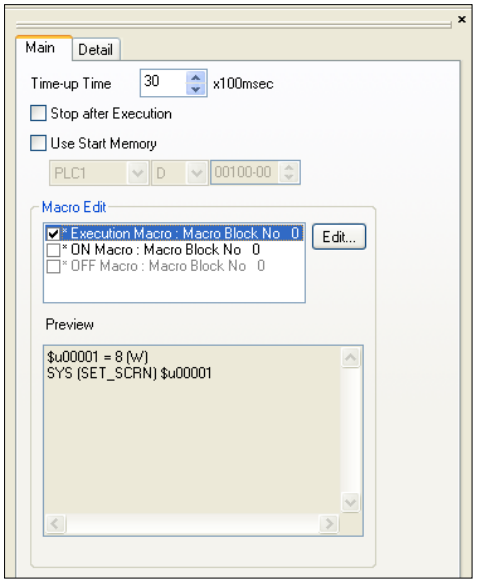
- Now let's add the time display.
- From [*Catalog View*], under [Calendar], you'll find [Time Display].
- Drag it to your screen.
- Other shapes for time displays are available in [Parts List].
- Click the time display to view its details.



- Under [Main] tab, check [Use System Calendar] if you want date and time to be read from PLC internal clock or uncheck it if they will be read from a specific memory in PLC.
- We'll use system calendar in our case.
- You can specify the format in which date or time or both will be written.
- Also under [Style] tab, you can change the font size and color.

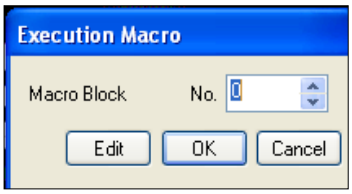
- Now it's time to add an interesting thing.
- Remember the introductory screen we created (Screen [0])?
- Let's add a timer to it so that when this time elapses, Screen [8] appears automatically which is our Main Menu Screen.

- Go back to Screen [0]. In [Catalog View], you'll find [Interval Timer] under [Macro].
- Drag it to you screen and view its [Details] window.
- Write the [Time-up Time] which should pass before the software moves to Screen [8].
- This time will be multiplied by [100msec].
- In our case, we'll write [30] so this will be 3 seconds.

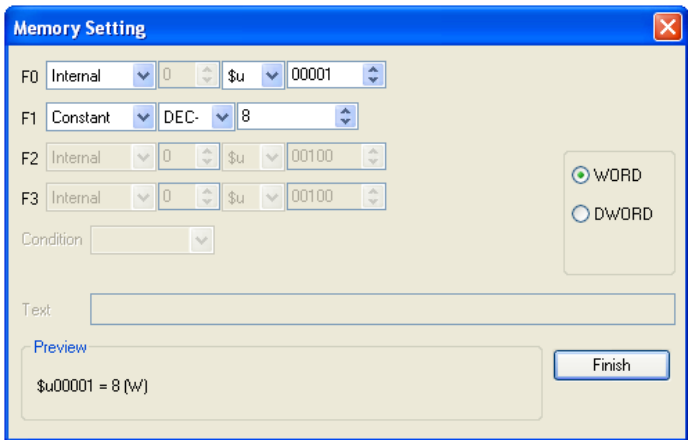


- Press the [Edit…] button to write a script in which you'll specify the screen that will be displayed after the elapsed time.

- In the [Execution Macro] window, press [Edit].

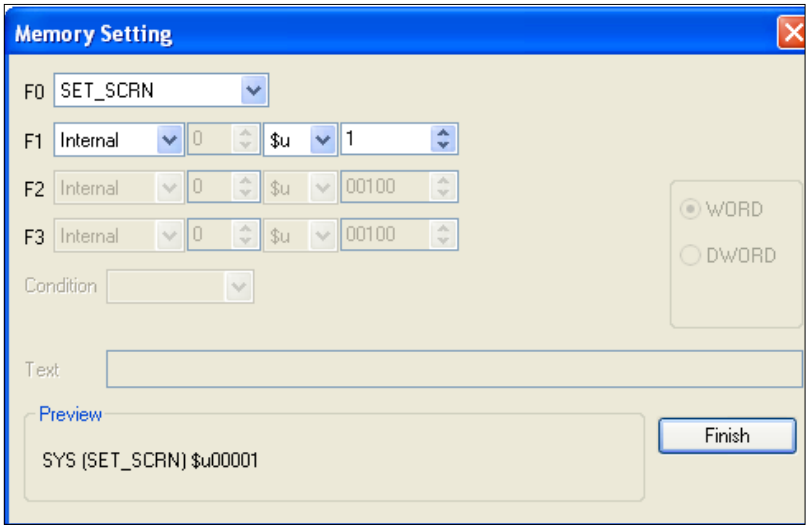


- In the window that appears, write MOV.
- Double click the word [MOV].
- The following window will appear:



- For F0, write the internal memory $u0001.
- For F1, write the constant 8
- Click [Finish] you'll get the first line in your script "$u00001 = 8 (W)"
- This means that the memory **$u0001** has the value **8.**
- After the line you just created, press [Enter] from the keyboard to move to the next line.
- Write [SYS], a small help window will appear.
- Double click the word [SYS].

- The following window will appear:



- For F0, choose the function [SET_SCRN]
- For F1, write the internal memory $u0001
- Click [Finish] you'll get the second line in your script "SYS (SET_SCRN) $u00001"
- This means that when the time you specified passes, the software will move to the screen number which is specified in the internal memory **$u0001** which **8** in our case.

- Now test your application using the emulator.

- Last step we'll do to finish our application, is to add a [MENU] switch to each screen beside its title to get back to the main menu screen (Screen [8]).

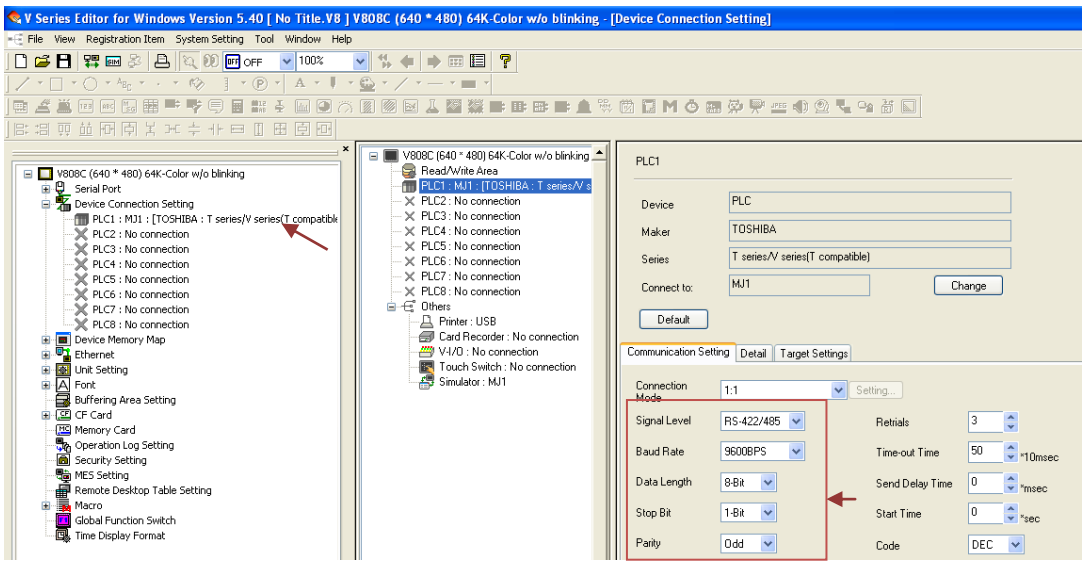
- This shall be easy for you now. Create the switch and give it the function [Screen], then write [8] as a screen number.

Notes:

6. Screen Data Transfer

Screen Data Transfer

- In this part, we want to transfer the application we just created or any other application to the touch screen.
- Connect the touch screen to supply (24VDC in our case).
- Connect the cable between PLC and the screen. You can order the communication cable or make a one by yourself using “V8 Connection Manual”.
- Assume that we’ll connect cable to port MJ1 in touch screen and RS-485 port in PLC.
- Make sure the communication parameters in your touch screen match those in PLC.
- You can adjust the communication parameters in touch screen by clicking on [Device Connection Setting] in [Project View] on the left to expand it. Then double click [PLC1] to get this window:



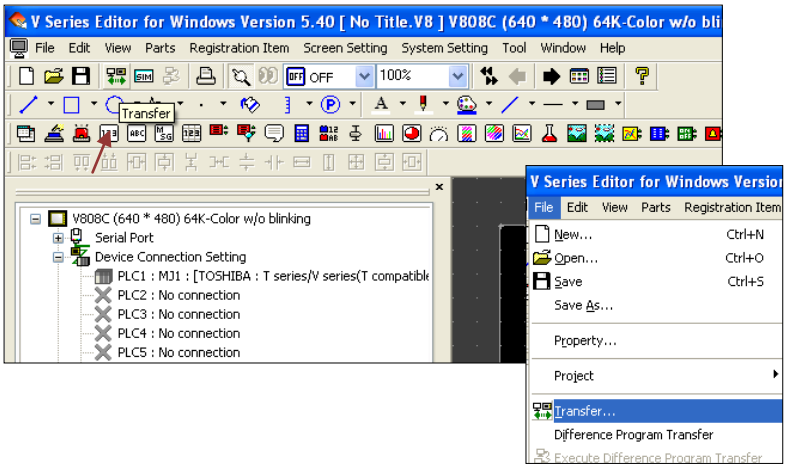
- Signal Level is the communication method (Protocol) you’ll use to connect the touch screen to PLC.

  RS-422/485 supports distance up to 1 km
  RS-232C supports distance up to 15 m only

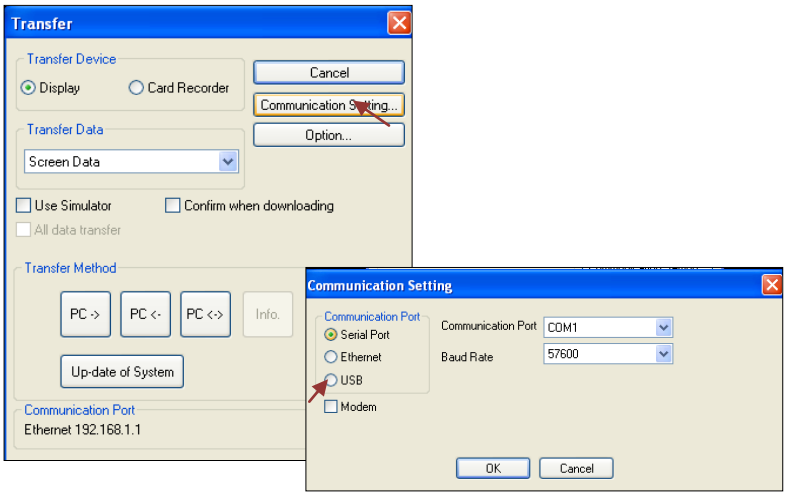
- Adjust Baud Rate, Data Length, Stop Bit and Parity to match those in PLC.
- Once done close this window.

- Now connect Touch Screen programming cable. We'll use the USB cable because it's faster and available anywhere.

- On the V-SFT screen (on the computer), click the [Transfer] icon or select [Transfer] from the [File] menu.

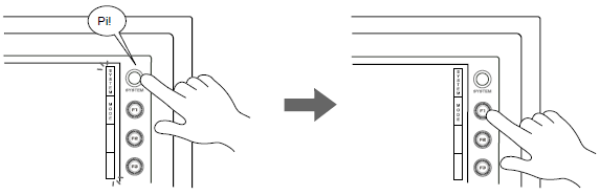


- Once you click [Transfer], you'll get [Transfer] window. Click on [Communication Setting…] to adjust the communication settings between screen and PC.



- In case you decided to connect a serial cable, you'll have to choose the COM number from [Communication Port] and choose the [Baud Rate].

- You can get these two parameters once you connected a cable to PC.
- Right click on [My Computer]/ [Properties]/[Hardware]/[Device Manager]/[Ports].
- As we said, for this training, we'll use the USB cable. Then click [OK].
- Back to [Transfer] window again, click [PC ->] to download.
- In case you want to upload a software for touch screen to PLC, you'll then choose [PC <-].
- Wait until the transfer is done.
- Now you're ready to view your application in screen, press [System] then [F1] hardware switches on screen.



- Press [Menu], then [Run].

- If you get an error screen saying "Communication Error", then the following could be possible:

An incorrect PLC type is selected
Check if a correct PLC type is selected in the [Select PLC Type] dialog when a new screen is opened.

The cable is connected incorrectly
Check if the cable is connected to the correct pins.
Check the cable for breakage.

The communication parameters set up for PLC and V7 are not in conformity.
Check the parameter contents.

Notes:

ALMAWARED
ENGINEERING
& TRADING SAE

2 Ahmed Taysser St., Heliopolis, Cairo
Tel. : 24192480 | Fax : 24192483
almawared@link.net www.met-eg.com