

*i*<sup>3</sup> ...Display, Control, Connect...



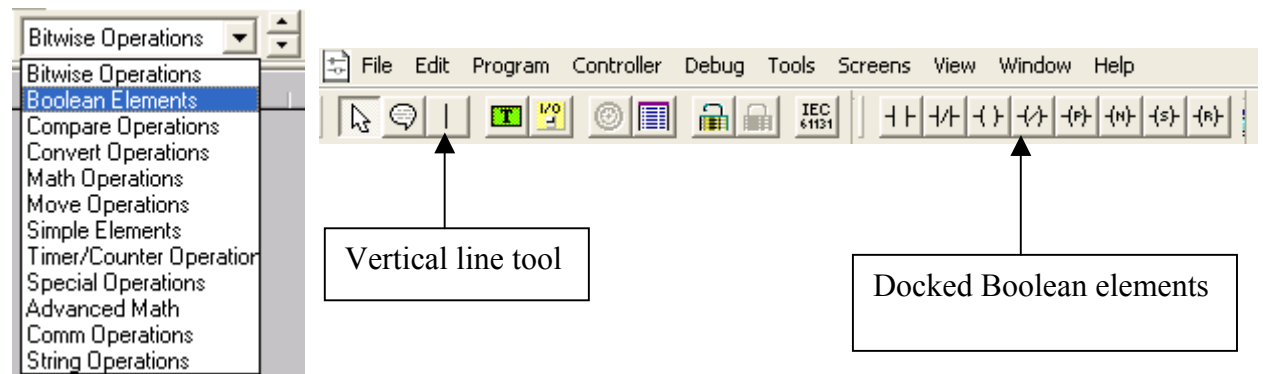
## Software Manual

# *i*<sup>3</sup> Basic Operation

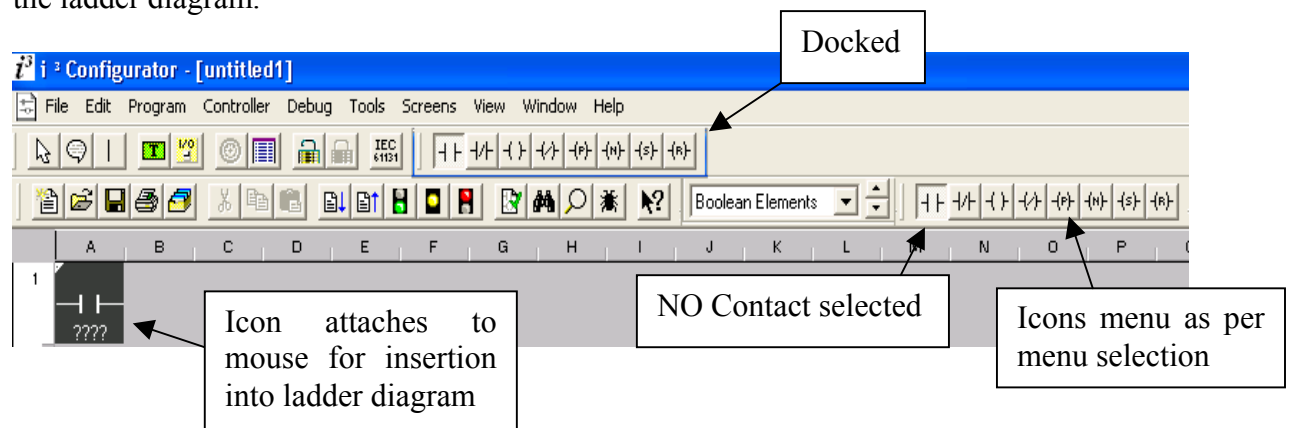
Basic Ladder Elements .....	3
<i>i</i> <sup>3</sup> Memory Locations.....	4
Input / Output Types .....	5
Assigning Function Keys and Screens.....	6
Basic Ladder Logic .....	9
Or Gate.....	11
And Gate .....	11
Basic Function Blocks .....	12
Timers and Counters .....	13
Timer Set up.....	13
Counter Set up.....	17
Move Functions .....	18
Set Points .....	20
Download Options .....	21
Tour of Screen Editor.....	23
Static Text .....	25
Numeric Data .....	26
Time Data.....	28
Password .....	29
Text Table data .....	30
Indicator Lamp.....	32
Button.....	33
Selector Switch .....	34
Screen Jump .....	35
Bar Graph.....	36
Meter Graph .....	37
Gauge Graph .....	38
Static Bitmap.....	39
Animation .....	40
Data Trend .....	41
X – Y Data Graph .....	42
Alarms.....	43
Configure the Alarm Log.....	44
Recipe Editor Object.....	45

## Basic Ladder Elements

All the Functions, contacts and coils for the ladder programming are grouped into similar elements and accessed via a drop down menu on the tool bar. It is possible to scroll through these groups using the up and down arrow keys. Each group can also be docked on to the main programming interface.

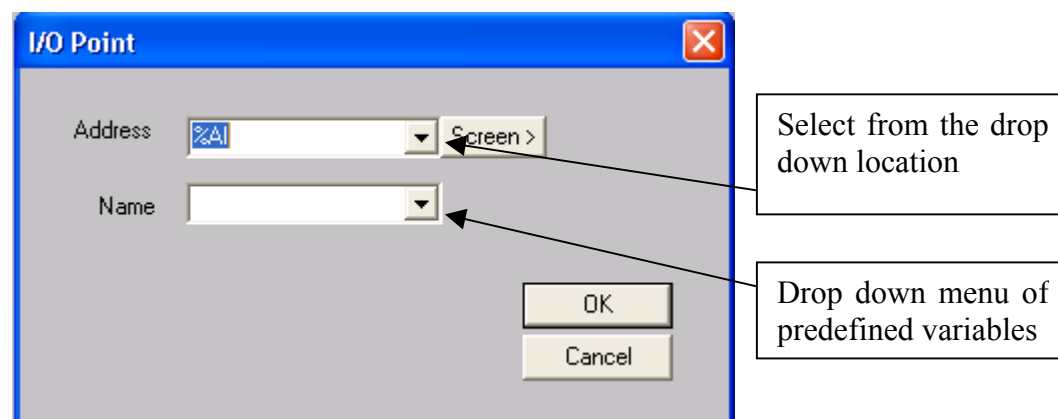


Once you have selected a group it is now possible to click on the icon to insert it into the ladder diagram.



Enter the contact / coil / function into the diagram by clicking it into the Ladder diagram.

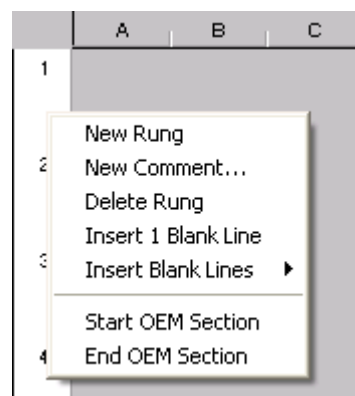
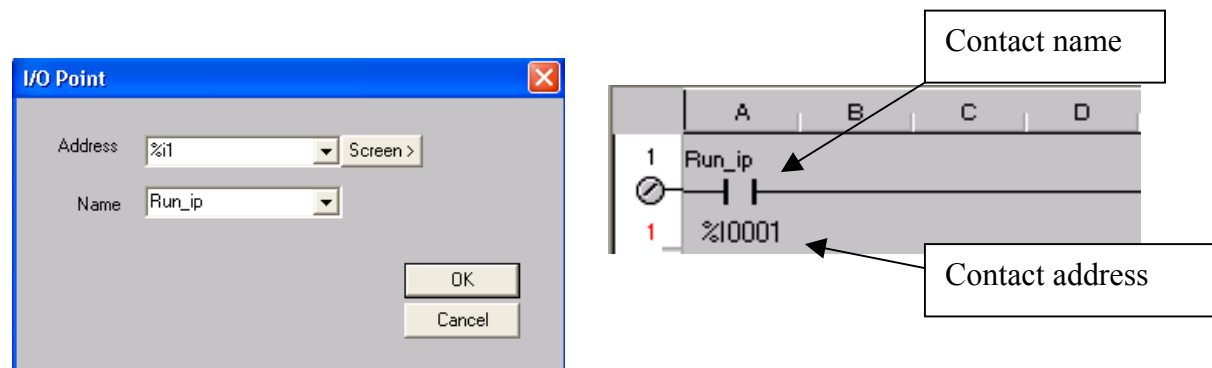
When inserting a Contact or a Coil you will need to assign the variable to a memory location, giving it a name at this point is a good idea for your documentation.



### i<sup>3</sup> Memory Locations

Type	Description and example of what might use the type	Format
%I	Discrete Inputs from the field; proximitor sensors, panel buttons, etc	BOOL
%Q	Discrete Outputs to the field; relays, indicator lamps, etc.	BOOL
%AI	Analogue Inputs from the field; Thermocouples, 4-20mA inputs	WORD
%AQ	Analogue Outputs to the field; 0-10VDC or 4-20mA outputs	WORD
%IG	Global Discrete Inputs from the CAN smart i/o;	BOOL
%QG	Global Discrete Outputs to the CAN smart i/o;	BOOL
%AIG	Global Analogue Inputs from the CAN smart i/o;	WORD
%AQG	Global Analogue Outputs to the CAN smart i/o;	WORD
%T	Internal Temporary bits, use for contacts and coils	BOOL
%M	Internal Temporary bits, use for contacts and coils	BOOL
%R	Internal Registers, use for timers, counters & other data	WORD
%K	Keypad bits, reflect Function Key status	BOOL
%D	Display bits, control screens or indicate screen on/off	BOOL
%S	Internal System Bits (See System Registers)	BOOL
%SR	Internal System Registers (See System Registers)	WORD

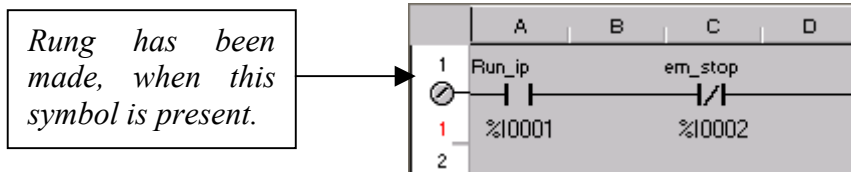
**NOTE:** The allocation of I/O points start from #1, i.e. the first input is %01.



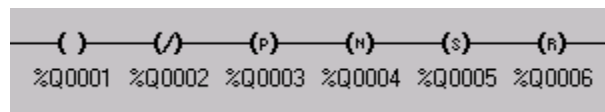
By right clicking the mouse in the left margin we can have some more options. We have the options to add or remove Rungs, insert documentation and start OEM sections (to be discussed later in this document)

## Input / Output Types

The **I<sup>3</sup>** has 2 types of inputs: Normally Open (NO) and Normally Closed (NC).

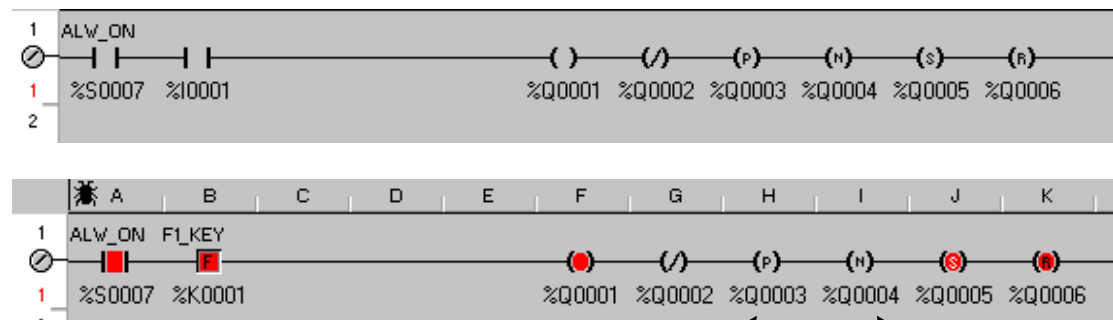


The **I<sup>3</sup>** has 6 types of outputs: NO, NC, Positive Edge, Negative Edge, Set and Reset.



More than one Output can be put on a single Rung. When the input condition is met all of the outputs on the Rung will be actioned.

**Example:** When input %I0001 is enabled, %Q0001 goes high, %Q0002 goes low, %Q0003 switches on the positive edge, %Q0004 will switch on the negative edge (when %I0001 is switched off), %Q0005 will set and %Q0006 will reset.



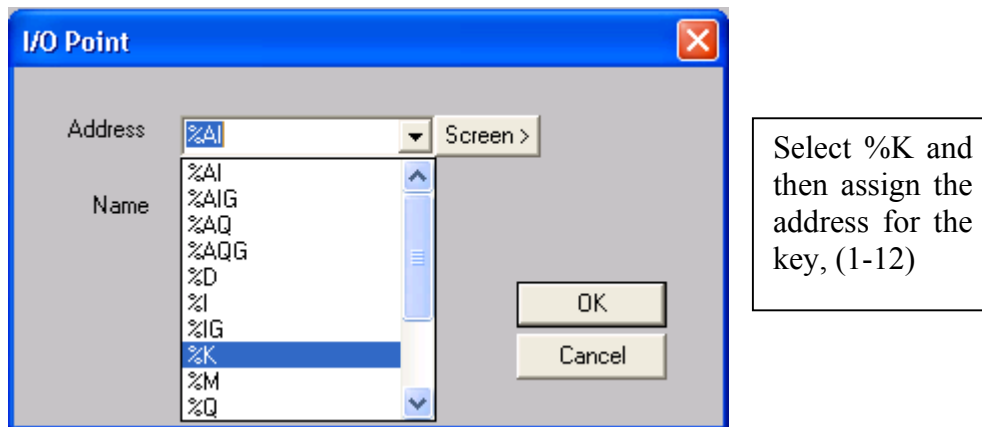
%Q0003 was "ON" only on the positive edge of the input.

%Q0004 was "ON" only on the negative edge of the input.

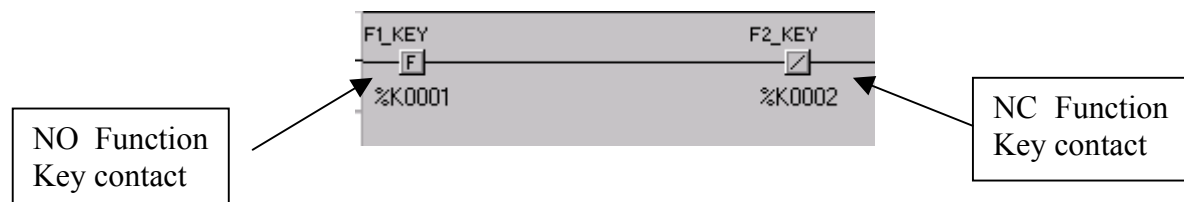
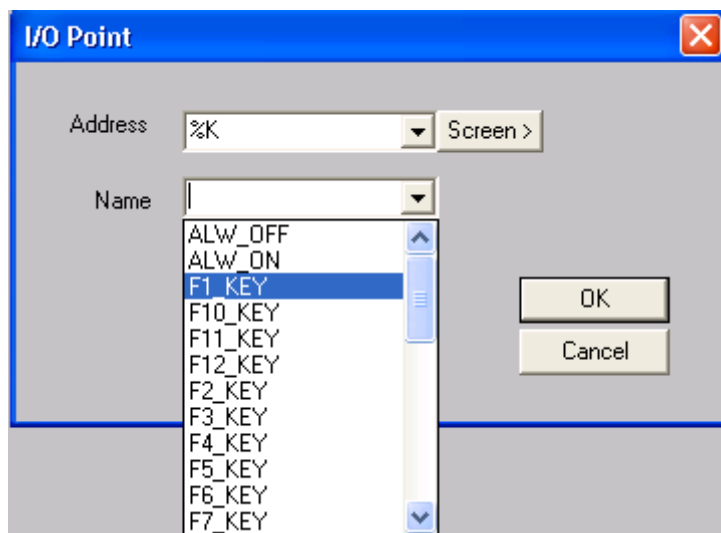
## Assigning Function Keys and Screens

The *i*<sup>3</sup> has 11 function keys that can be programmed into the ladder diagrams. It also has four buttons that can be programmed through the screen editor and these will be discussed in detail later in this document.

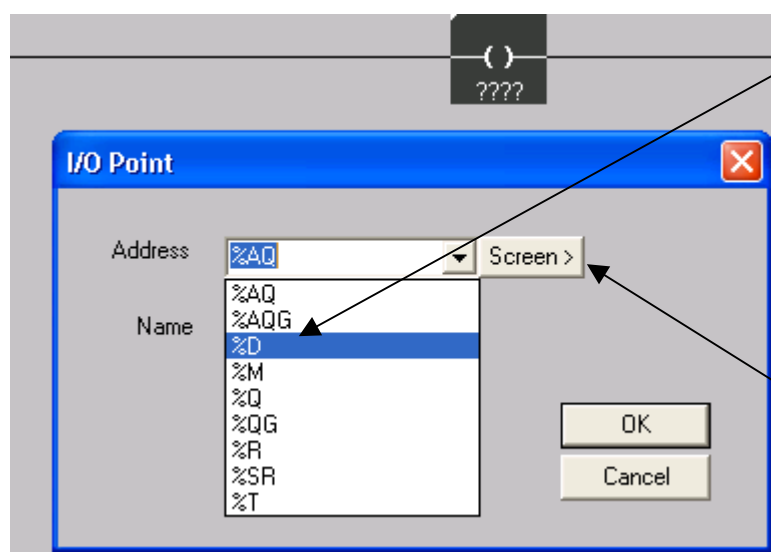
Assigning Function Keys in the ladder diagram is very straightforward. Simply select an input contact and address it to the keys.



Or select it by its predefined name. (This can be changed to a User Defined name.)



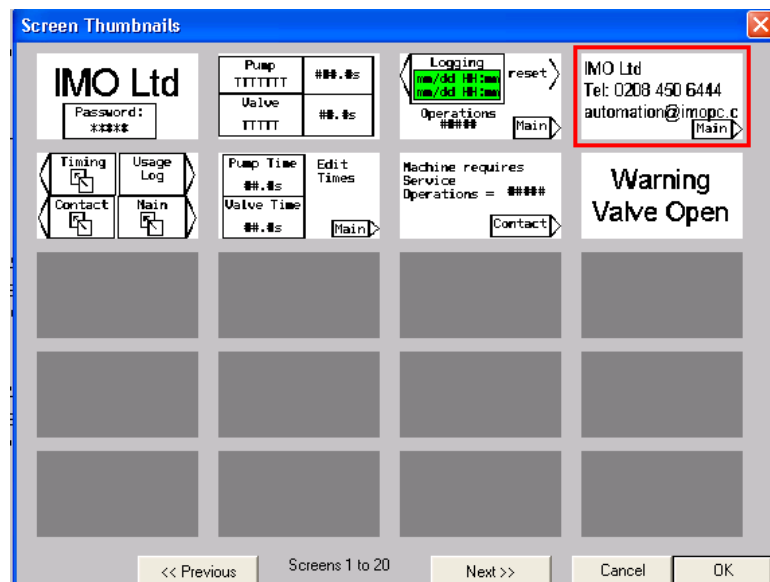
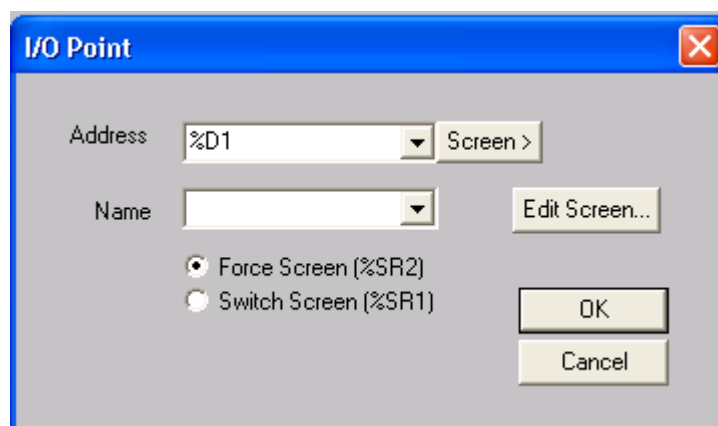
To insert a screen into the ladder logic, select an output coil and click it into the ladder diagram.



Select the %D address and then enter the number for the screen to display i.e.%D0001 is screen 1

Or, by clicking the screen button you can choose a screen from the thumbnails.

*The thumbnails represent screens that have already been created in the program. If no screen has been set up, the window will show as series of blank grey boxes.*

We can open the screen editor from this menu however this will be discussed later.

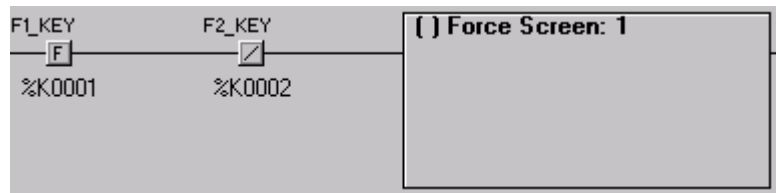
There are now two options for changing to the screen selected:

1. Force Screen:

Displays the screen whilst the condition to make the contact is true.

When the condition is no longer true the display will show to the previously set screen.

Example of use: Warning Message.



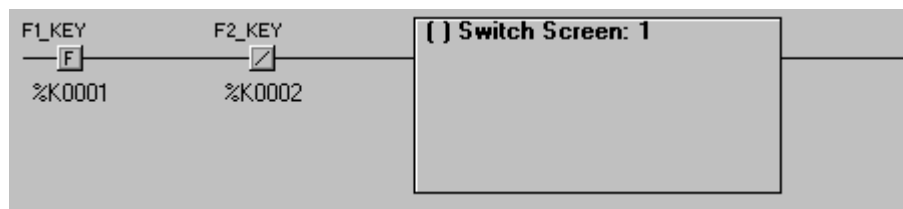
When F1 and not F2 are pressed the *i<sup>3</sup>* will display screen 1.

When F1 has been release the *i<sup>3</sup>* will display the previously set screen.

2. Switch Screen:

Changes the display to the desired screen when the condition is true.

Example of use: Changing screens on a menu selection.



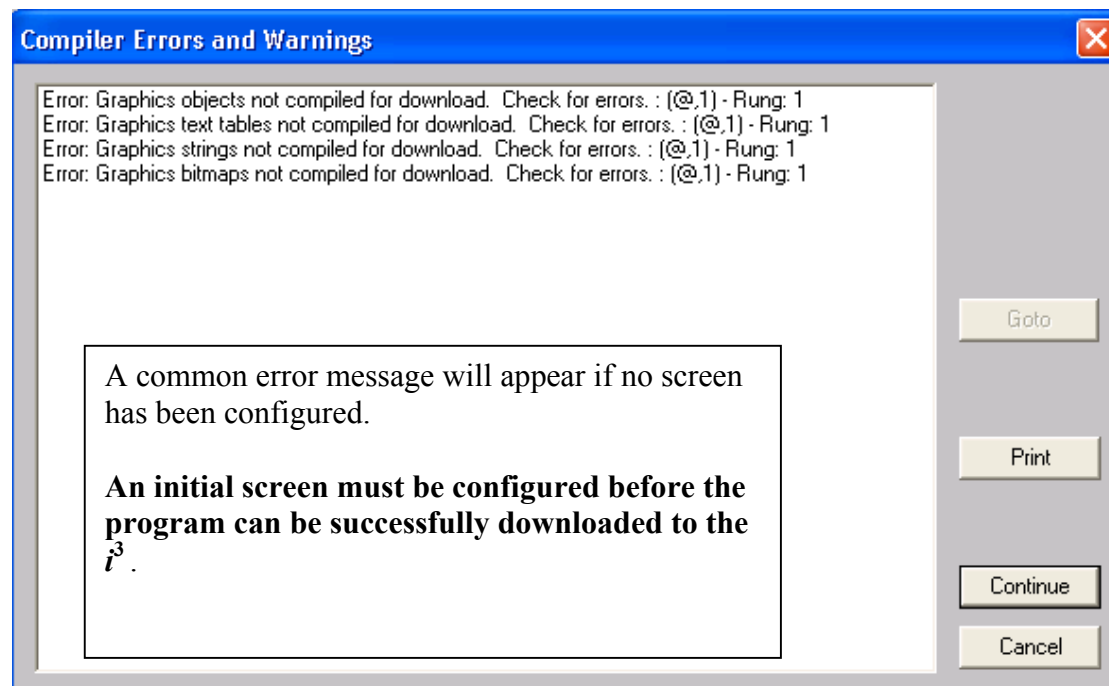
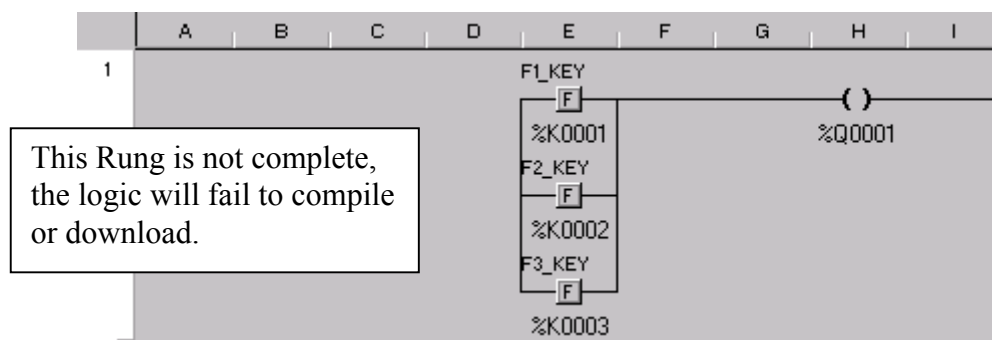
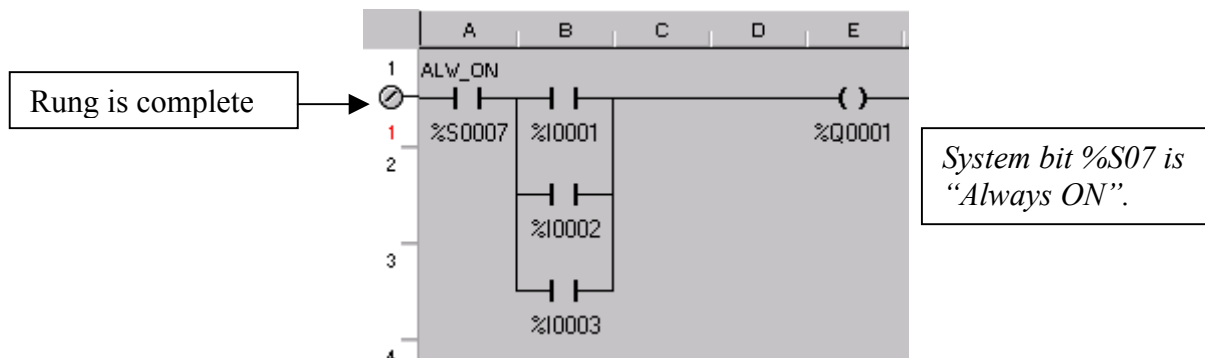
When F1 and not F2 are pressed the *i<sup>3</sup>* will display screen 1.

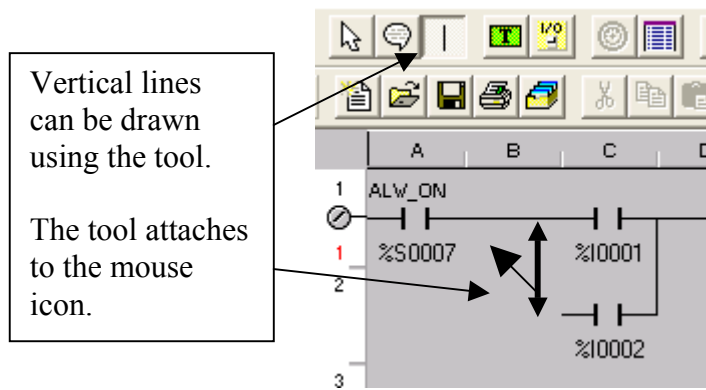
When F1 has been release the screen 1 will remain on the display.



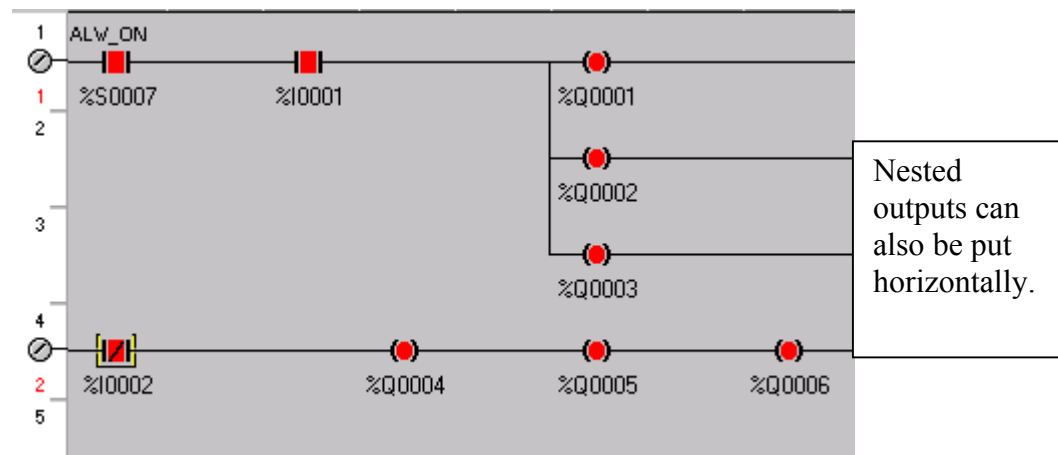
## Basic Ladder Logic

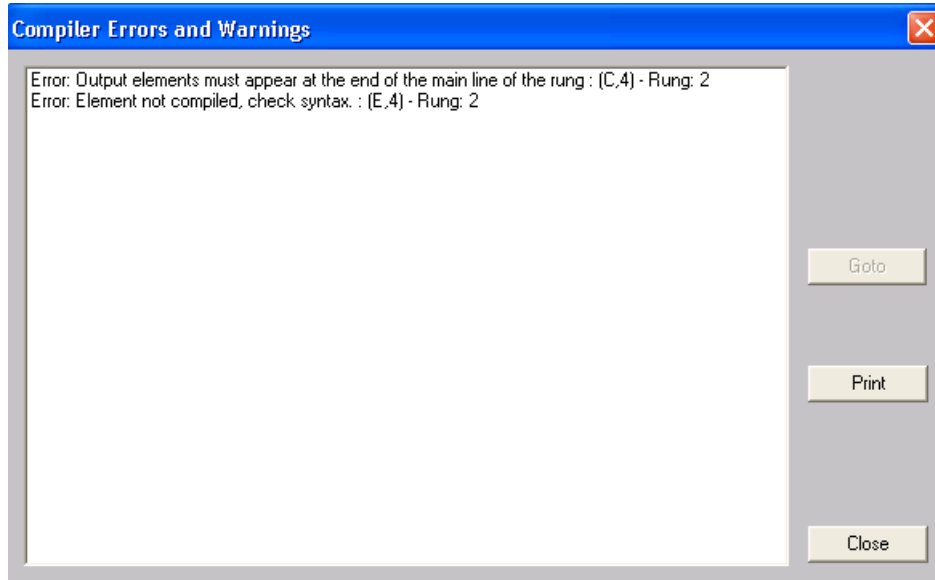
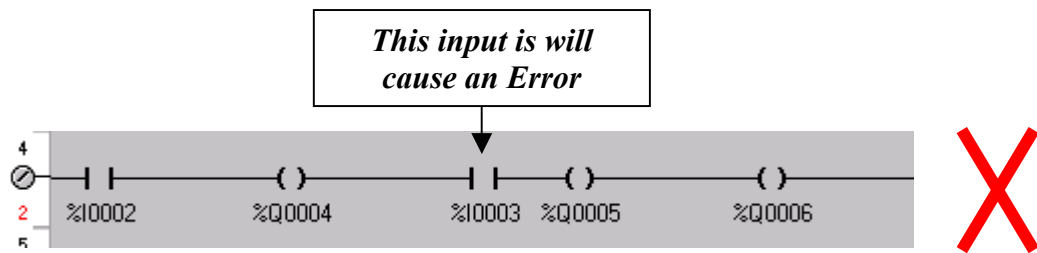
In the *i<sup>3</sup>* ladder logic it is necessary to insert a contact in the first column. Good practice is to insert an “Always ON” contact at the beginning of the first line and then add your logic after.





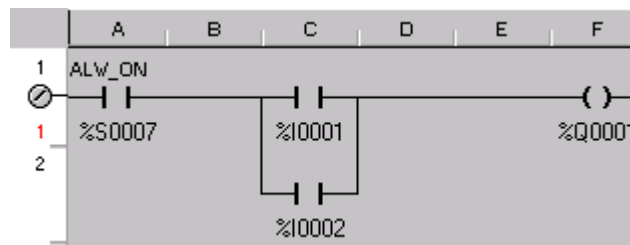
Each Rung can have multiple outputs but they must be at the end of the line.





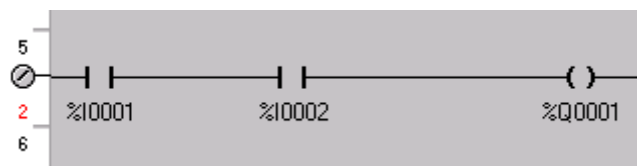
## Or Gate

Either input, %I0001 or %I0002 will switch on the output %Q0001.



## And Gate

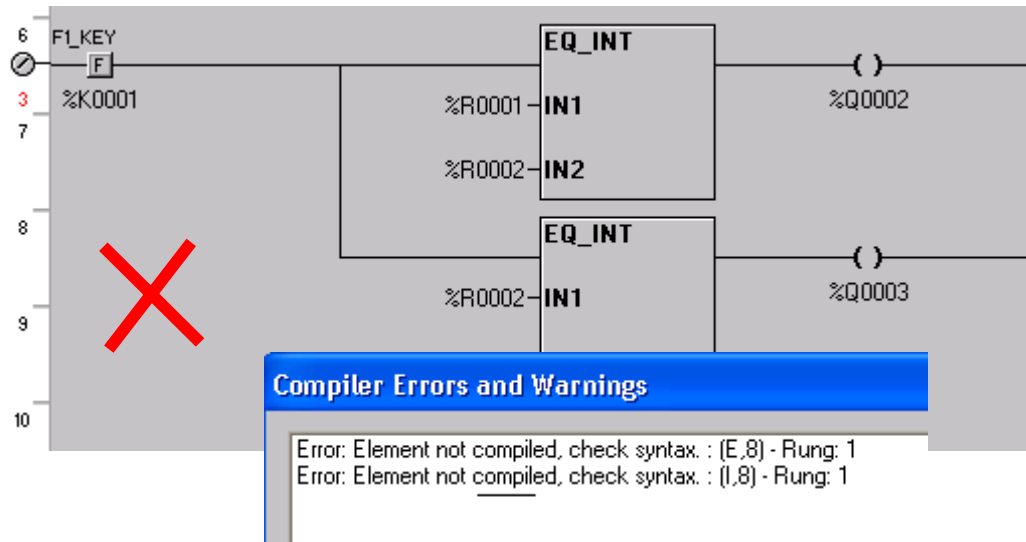
Both inputs %I0001 and %I0002 must be on for the output %Q0001 to be on.



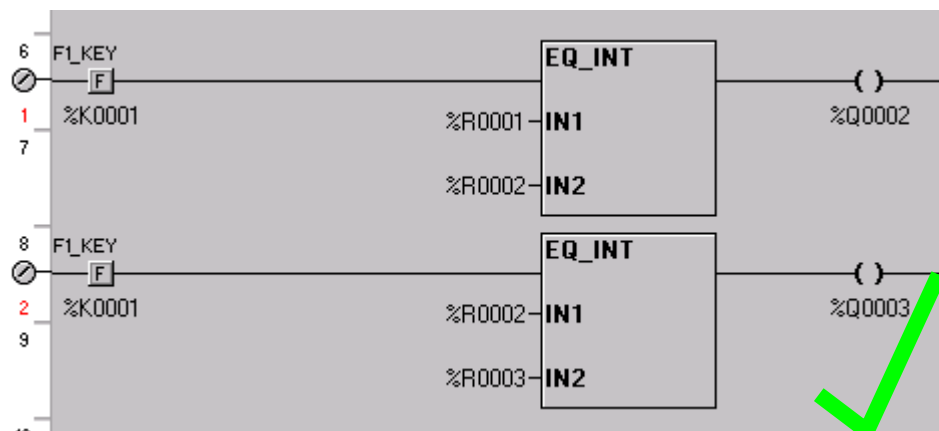
## Basic Function Blocks

All function blocks have an enable input and a control output. Other I/O parameters depend on the type of function block.

**NOTE:** Nesting function blocks will create an error.



The correct way to create the logic is as follows:



## Timers and Counters

Timers and counters require 2 consecutive registers to store their data in memory. The first register contains the current value and the 2<sup>nd</sup> register contains the status bits of the counter/timer.

Register 1 = Accumulated value

Most used status register bits

Register 2 .15 = Function Enabled

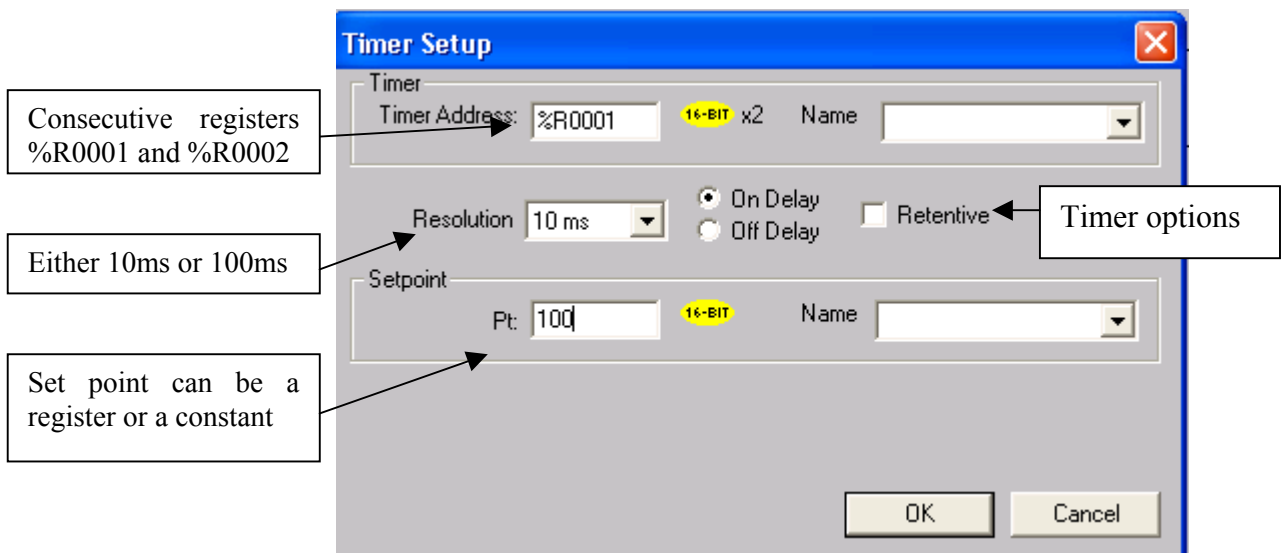
Register 2 .16 = Function Completed i.e. timing elapsed, preset value met.

### Timer Set up

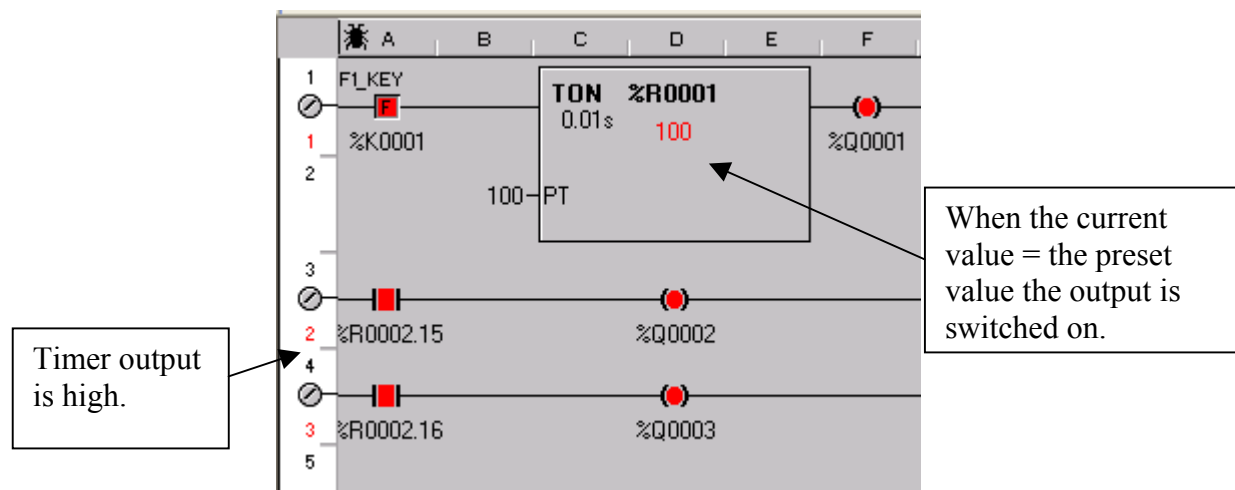
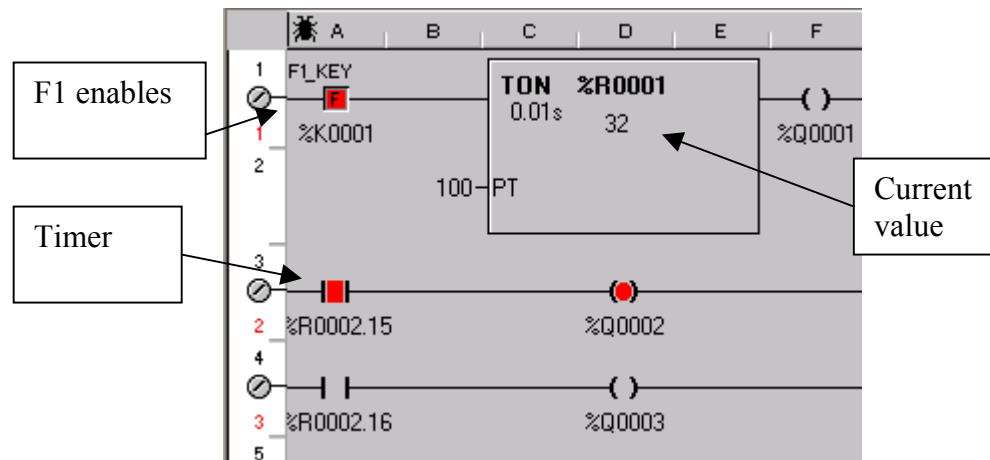
There are two types of timer, On-delay and Off-delay, only the On delay timers can be retentive.

#### On Delay

Delays before the output goes high. On being enabled the count begins, on the count having elapsed the output will go high. The output will remain high until the input has been removed.



**Example:**  
A 1second "ON" Delay

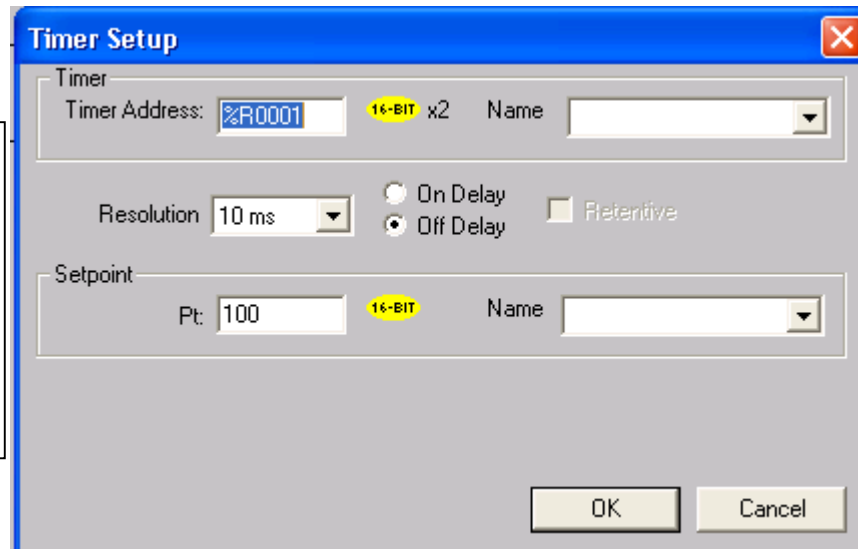


## Off Delay

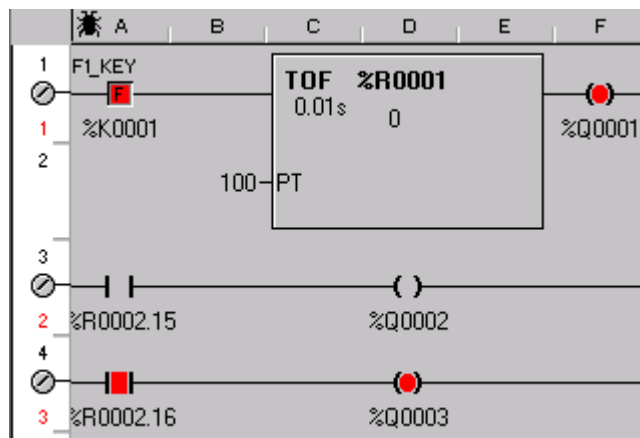
This timer maintains the output for a fixed time after the input has been switched off. On enabling the input to the timer the output is set high. When the input is removed the timing will begin, when the timing has elapsed the output will reset.

In this case the timer is to be an Off Delay type.

Notice the Retentive box is now greyed out.

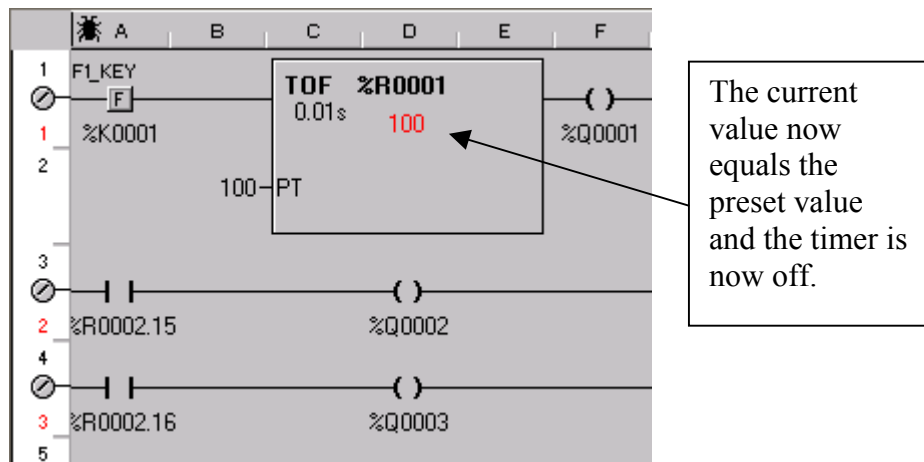
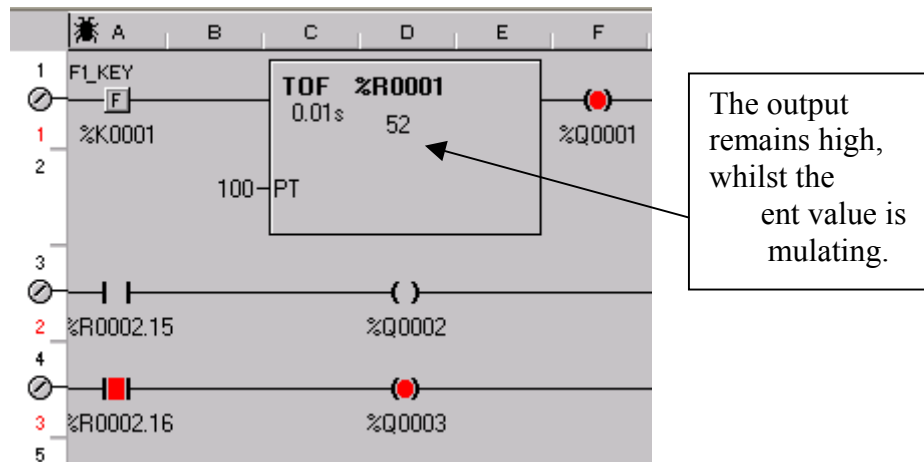


The 'Timer Setup' dialog box is shown. It has a blue title bar with a close button. The 'Timer' section contains 'Timer Address: %R0001' (with a '16-BIT x2' label), a 'Name' dropdown, and 'Resolution: 10 ms'. Below this are radio buttons for 'On Delay' and 'Off Delay' (selected), and a greyed-out 'Retentive' checkbox. The 'Setpoint' section contains 'Pt: 100' (with a '16-BIT' label) and another 'Name' dropdown. At the bottom are 'OK' and 'Cancel' buttons.



The input to the timer has been enabled and the output is high.

When the input is removed the timing begins.





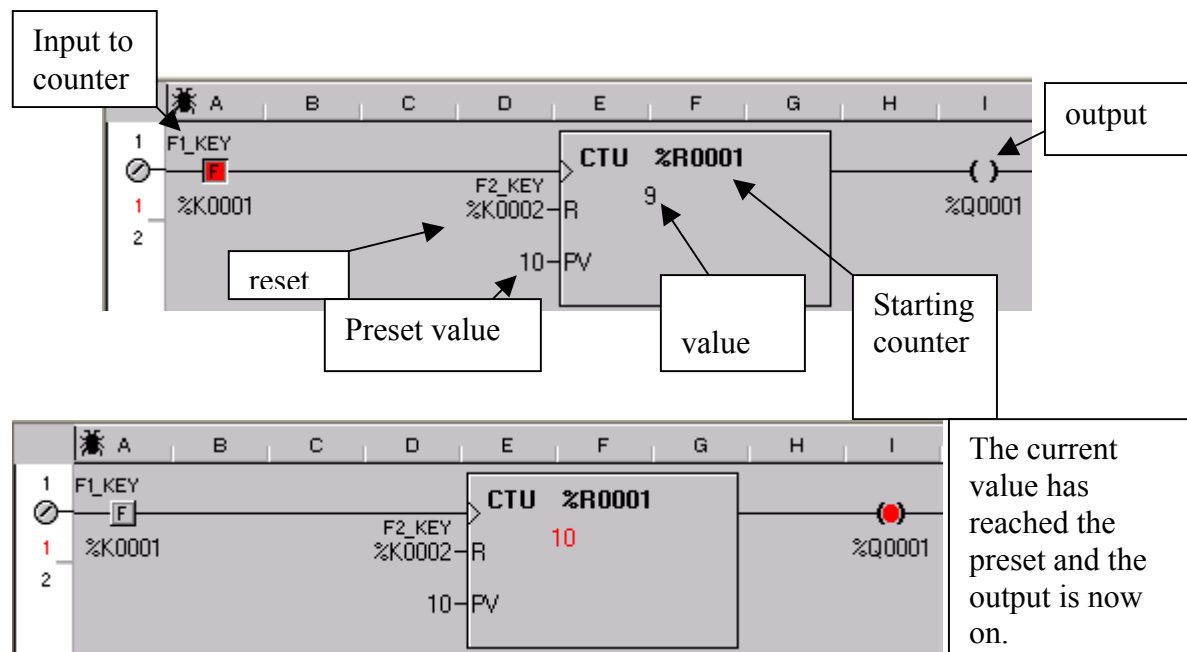
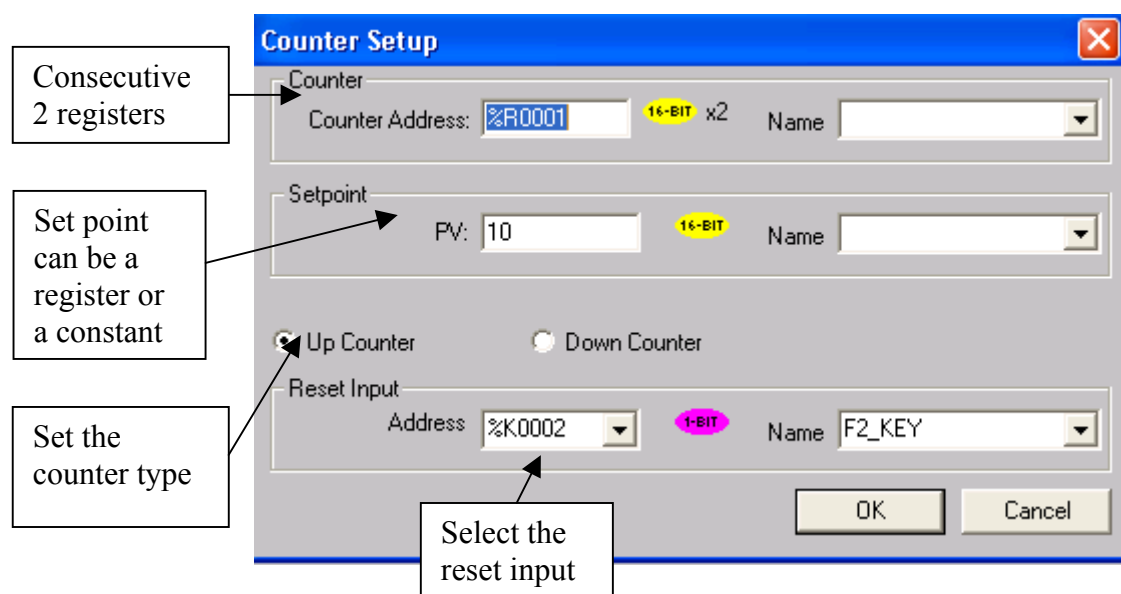
## Counter Set up

There can be two types of counters, “Up” count and “Down” count. All the counters count on the positive edge of an input and require a reset input.

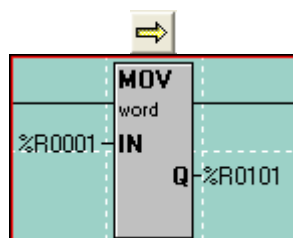
An up counter will count up from 0 to the preset value at which point the output is enabled.

A down counter will count down from the preset to 0 at which point the output will be enabled.

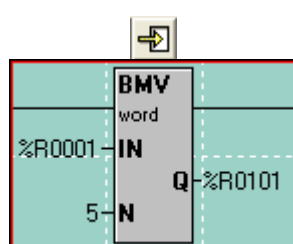
Both counters will overtake the preset if the input continues to give input pulses.



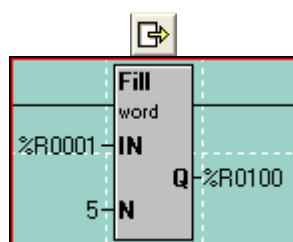
## Move Functions



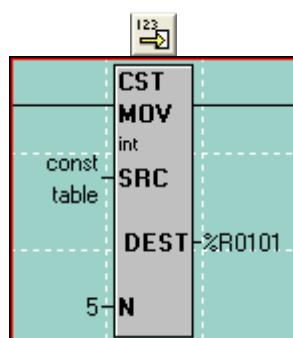
The first type of Move is the ‘Move Word’, or ‘MOV’. It is used to copy a single byte, word or double word from one location to another. The count is locked at 1. In the case of the example to the left, the value in %R0001 is copied into %R0101. This only happens when the ladder rung receives power. The value in %R0101 is reset when power is lost to the rung. The IN can be either a register or a constant value.



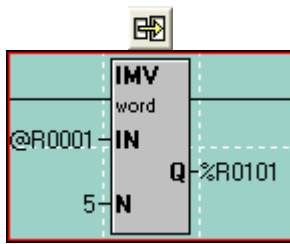
The next type of Move is the ‘Move Data Block’, or ‘BMV’. It is used to copy a group of bytes, words or double words to another location. The count (N) determines how many registers are to be copied. In the example to the left, %R0001 to %R0005 are copied into %R0101 to %R0105. Again, this only happens when the ladder rung receives power. The IN must be a register reference, constant values are not allowed.



The next type of Move is the “Fill WORD”, or “Fill”. It is used to copy the contents of a single register or value into multiple other registers, thus filling that one value into a group of registers. The count (N) determines how many registers to fill that single value into. In the example to the left, the value in %R0001 is copied into %R0101 to %R0105 so that these registers all contain the same value. This can be used to zero or reset a group of registers. The IN can be either a register or a constant value.

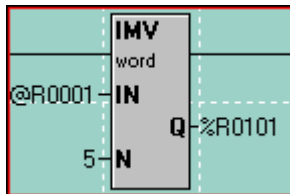


The ‘Constant Move’, or ‘CST MOV’, is used to move a group of constant values into a group of consecutive registers. If, for example, you want to move the values 1, 2, 3, 4 and 5 into %R0101, %R0102, %R0103, %R0104 and %R0105, respectively, then you can use the Constant Move function. The count (N) is automatically determined by how many constant values you enter into the configuration for this function. The source data can ONLY be constant data and cannot be register references.

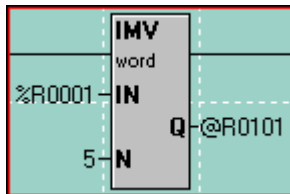


The 'Indirect Move', or 'IMV', is used to move data to and from variable positions. It functions, for the most part, like the Block Move 'BMV' function. If specified as Indirect, the IN and/or the Q are used as pointers to which registers (%R...) to retrieve or place data. When looking at the ladder logic, the '@' symbol will appear next to the IN or Q address if it is specified as Indirect. This function is fundamental when configuring complex data logging functions.

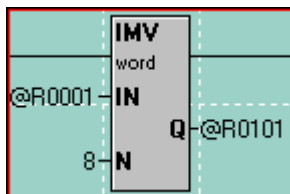
### Indirect Move Examples



In this example, the IN is specified as Indirect. This means the controller will look at %R0001 and see a value within it. If %R0001 has a value of 501 in it, the controller will go to %R501 to get the source data. 5 consecutive registers will then be moved, %R501-%R505 to %R101-%R105.



In this example, 'Q' is specified as Indirect. This means the controller will look at %R0101 and see a value within it. If %R0101 has for example a value of 851, the controller will take the data in %R0001 to %R0005 and move it into %R0851 to %R0855 respectively.

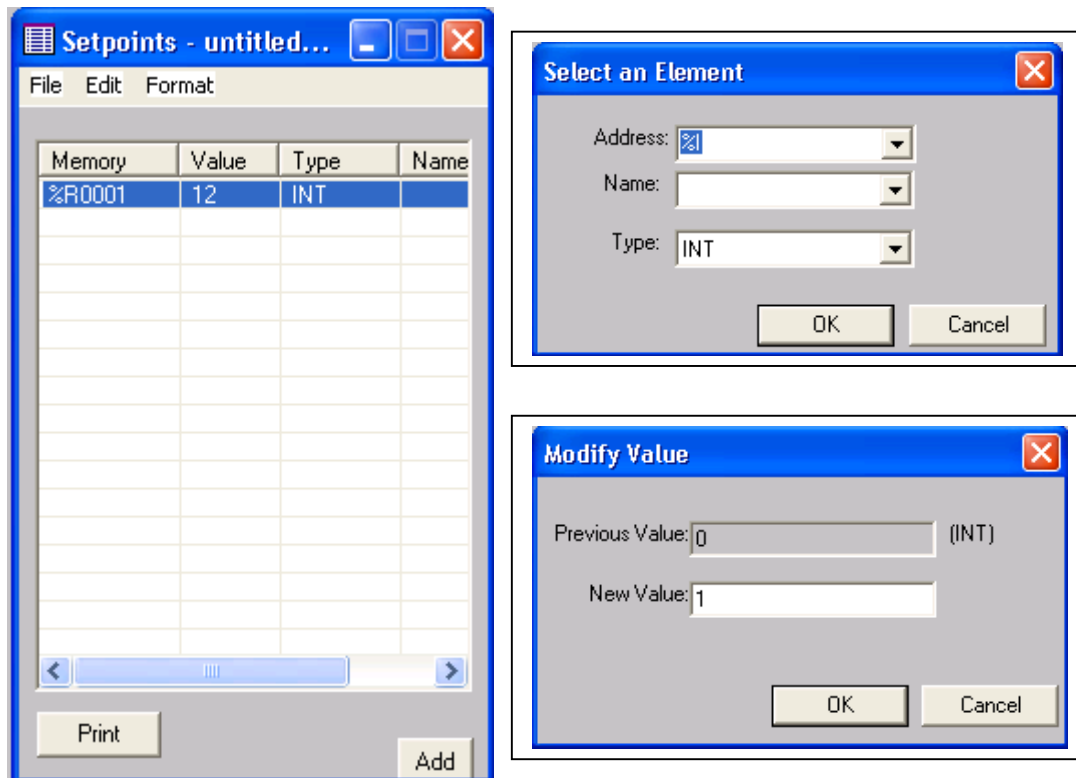


In this example, both the 'IN' and the 'Q' are specified as Indirect. This means the controller will look at %R0001 and see a value. Let's say it is 241. The controller also looks at the value in %R0101. Let's say it is 341. The controller will then take the values in %R0241 to %R0248 and move them into %R0341 to %R0348.

## Set Points


It is possible to set registers with initial values by using the set point editor. To open it select the Set point option from the program menu.

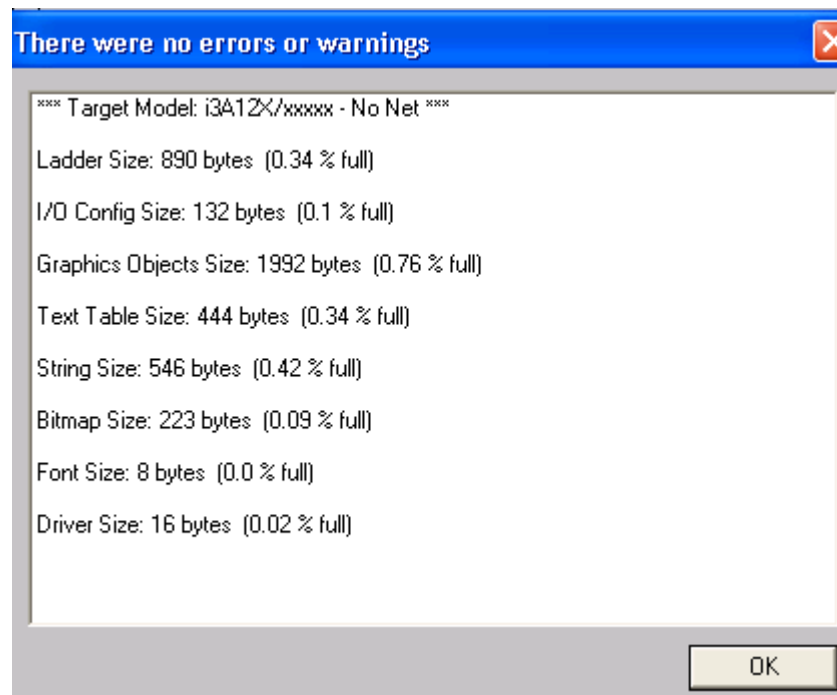
In the Editor we can 'Add' a new register and then double click on the value to enter a set point.



However you must select the download option to send the set points to the controller when downloading the program to the *i<sup>3</sup>*.

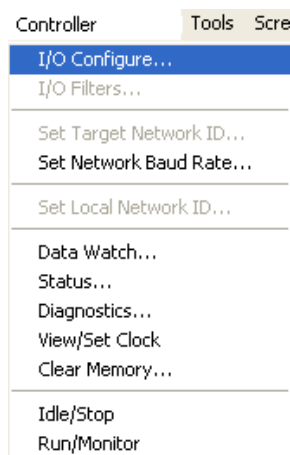
## Download Options

Once you have written the ladder logic (and inserted an initial screen) the program can be verified by clicking the icon . This will check the program for any errors, if there are none then we can continue with downloading the program to the *i³*.

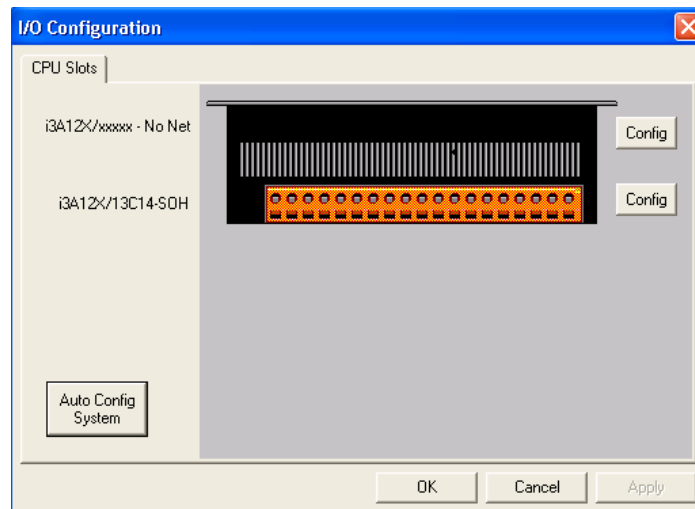


Before downloading the *i³* must be configured to the software.


Select the I/O  
configure from  
the controller  
menu.



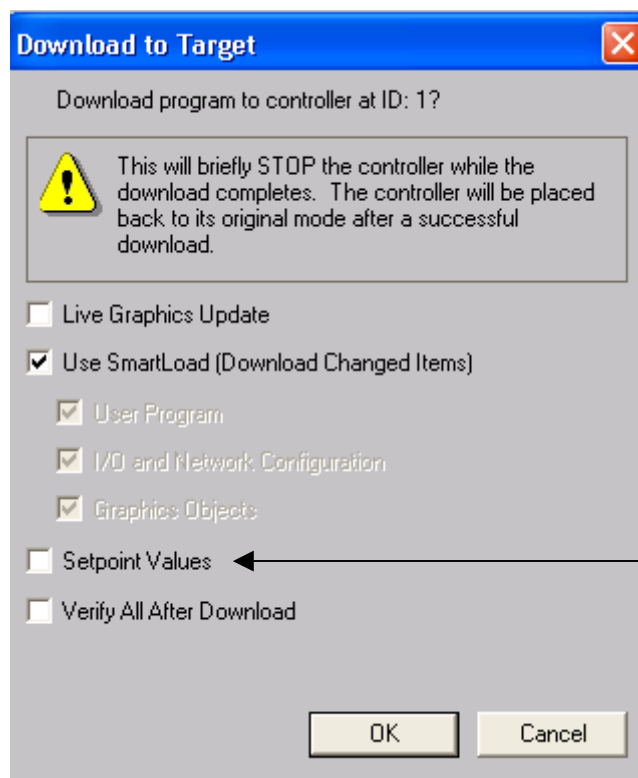
Then select Auto-Configure with the *i*<sup>3</sup> connected to your PC.



When configured the *i*<sup>3</sup> part number will match the unit.

To download to the *i*<sup>3</sup> click the icon  or select download from the program menu.

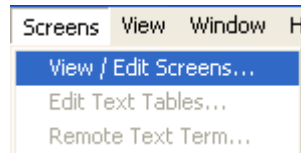
A download menu will pop up. It is simplest and most efficient to select the SmartLoad option.



To download set points, ensure that this option is checked.

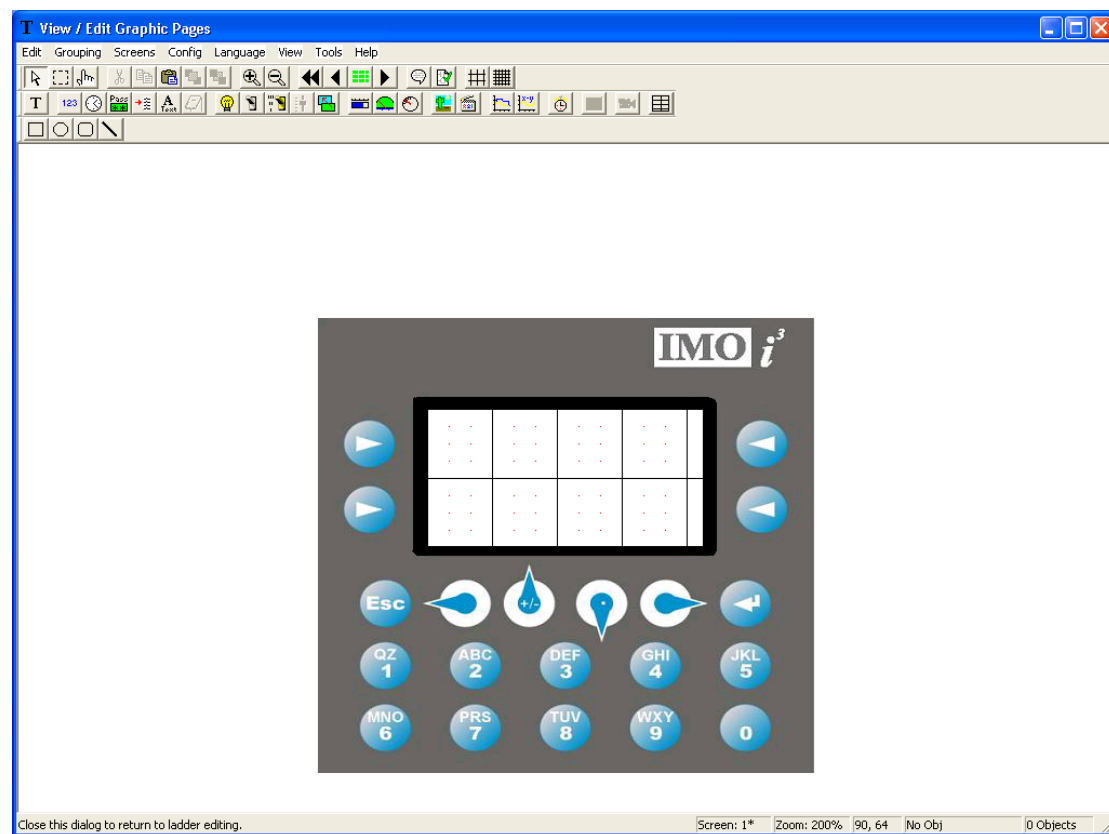
## Tour of Screen Editor

To open the screen editor click on the icon  or select the option from the screens menu.



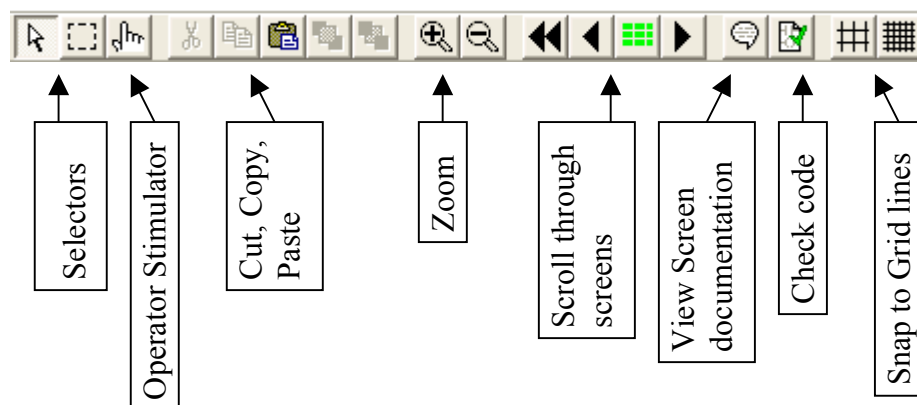
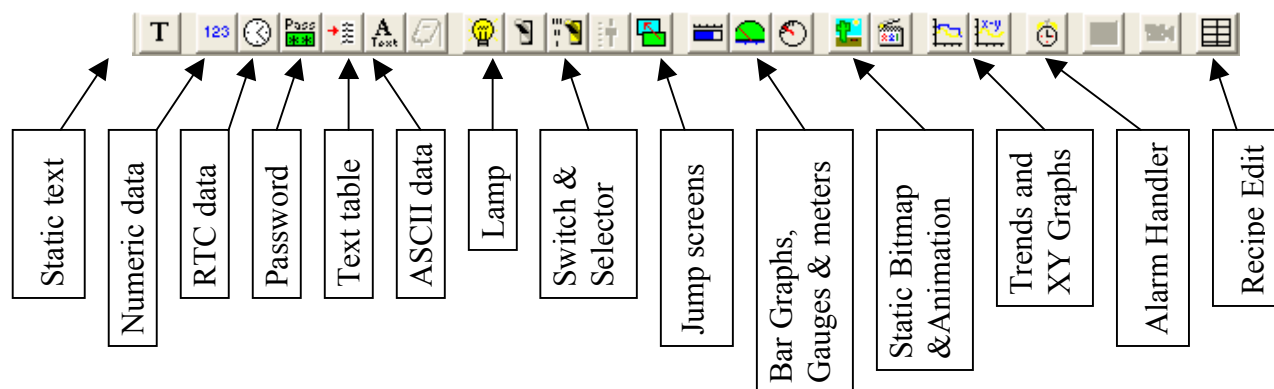
**NOTE:** A default screen must be configured before a program can be downloaded to the *i<sup>3</sup>*.

Before editing screens it is important to configure the I/O and set the correct *i<sup>3</sup>* model.



The screen editor program shows the *i<sup>3</sup>* in the middle of the screen with the programming functions at the top and screen information in the bottom right corner. To exit click the top right corner Windows 'X' button.

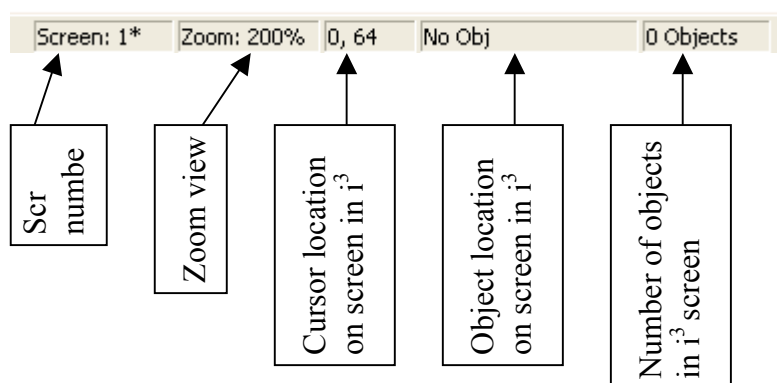
The screen editor tool bar.



Drawing Tools




Bottom right corner of Editor

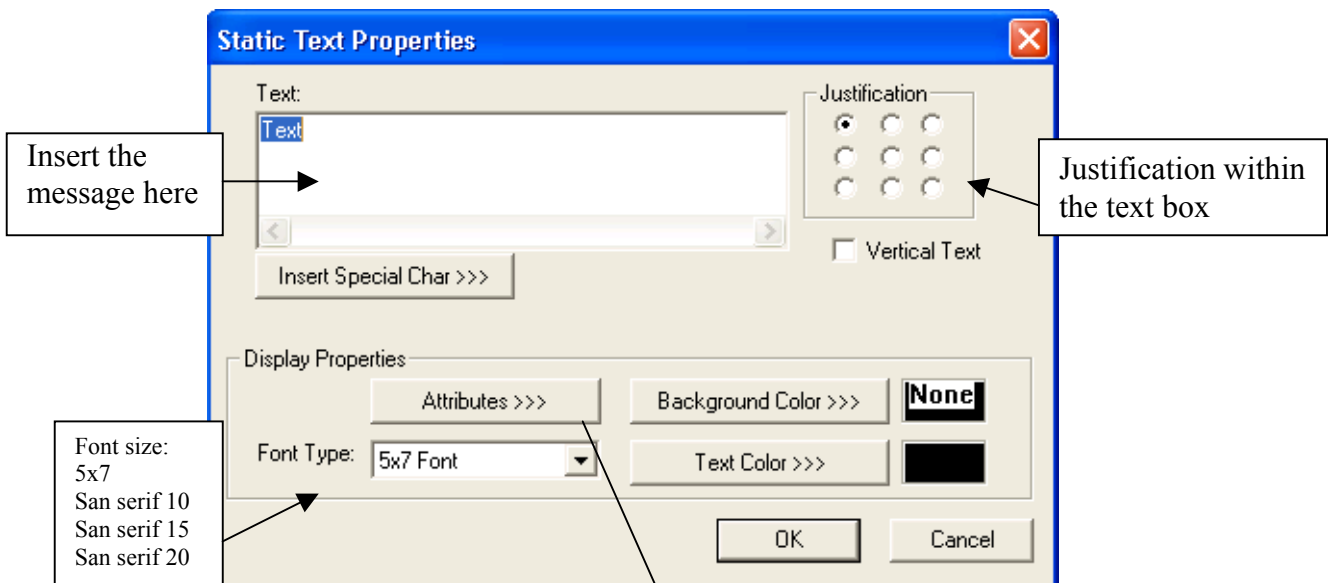
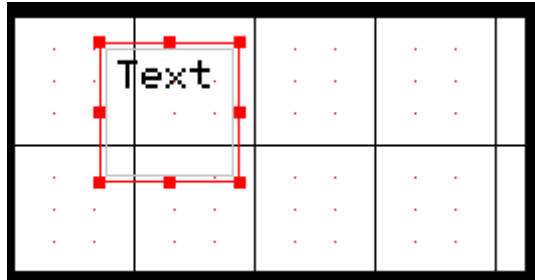




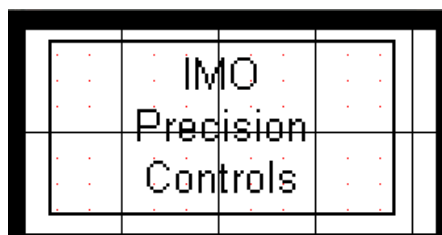
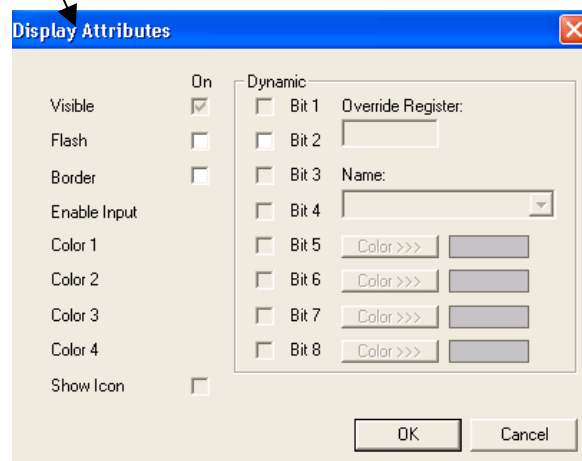
## Static Text

To insert a Static Text message click on the icon  and click it to the screen.


The box can be resized as required. To Enter the details of the Text double click on the box.



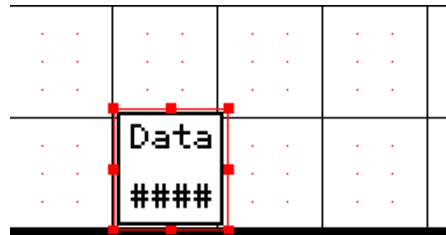
Once you have correctly  
OK to confirm and exit



## Numeric Data

To insert a numeric data display click on the icon  and click it into the screen.

It can be resized as required. To Enter the details of the numeric data double click on the box.



Address where the data is that you want to display

'Editable' allows the user to enter data through the screen.

**Numeric Data Properties**

Controller Register

Address:  Register Width:

Name:

Data Format

Justification: ☐ Left ☒ Center ☐ Right Font:

Digits:  Decimal Pos.:  Format:

☐ Zero Filled ☒ Editable ☐ 3D Sunken Min:  Max:

Scaling >>> Engineering Units:

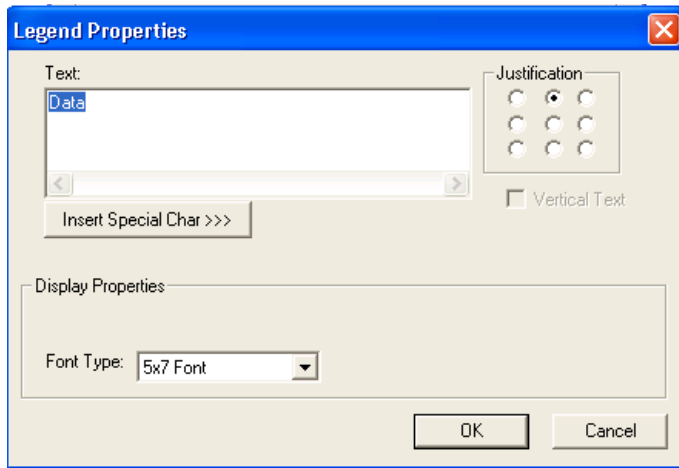
Display Properties

Attributes >>> Background Color >>>

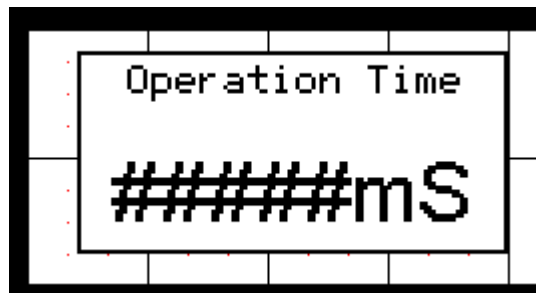
Legend >>> Line Color >>>

Data Color >>>


OK Cancel

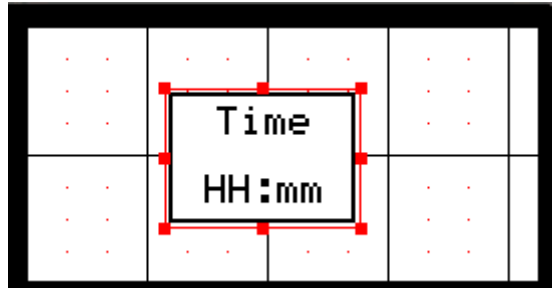


The Legend of the numeric box is a description name that can be given to the displayed data.



## Time Data

To insert a Time Data display click on the icon  and click it into the screen. Double click the box to edit the properties.



The system time is stored in consecutive SR registers from %SR0044

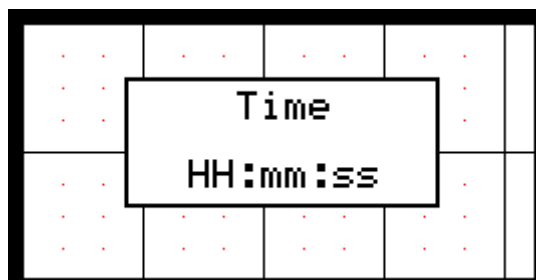
**Time Data Properties**

Controller Register  
Address:  Register Width: 48 bits (3 WORDs)  
Name:


Data Format  
Justification: ☐ Left ☒ Center ☐ Right  
Font:   
Time / Date Format:   
☐ Editable ☐ 3D Sunken

Display Properties

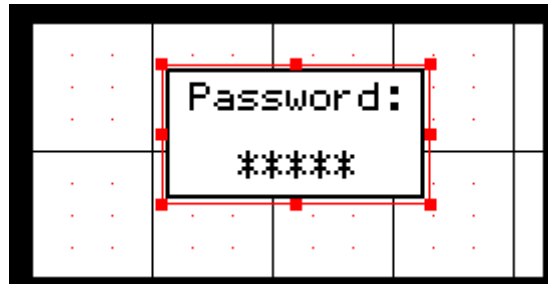
It is simpler to select the system RTC by choosing the name of the SR register



## Password

To insert a Password display, click on the icon  and click it into the screen.

similar too  
the numeric data  
display but the  
numbers are  
stared out.



**Password Data Properties**

Controller Register

Address:  Register Width:

Name:

Data Format

Justification: ☐ Left ☒ Center ☐ Right

Font:

Digits:

☒ Editable ☐ 3D Sunken

Display Properties

Attributes >>> Background Color >>>

Legend >>> Line Color >>>


Data Color >>>

OK Cancel

The address where the  
password will be  
entered.

A simple password logic  
system is to use a  
compare Function block,  
Equal to, to compare the  
enter password to a  
stored password.

## Text Table data

To insert a Text Table display, click on the icon  and click it into the screen.

The text table can be used to display text messages instead of numeric data for a given register.



**Text Table Data Properties**

Controller Register

Address:  Register Width:

Name:

Data Format

Justification: ☐ Left ☒ Center ☐ Right

Font:

Digits:  Text Table >>> Text Table Number:

☐ Editable ☐ 3D Sunken

Display Properties

Attributes >>> Background Color >>>

Legend >>> Line Color >>>

Data Color >>>

OK Cancel

A message table can be selected for a word, byte or a bit.

The number of digits to display must match the maximum length of message to display.

The image shows two overlapping windows from a software application. The background window is titled "Edit/View Text Tables" and contains a table with two columns: "Value" and "Text". The first row shows "0" in the Value column and "Imo" in the Text column. To the right of the table is a "Table Number:" label with a dropdown menu showing "1". Below the table is a large empty rectangular area. At the bottom right of this window is an "OK" button. The foreground window is titled "Text Table Entry" and has a blue header bar. It contains two input fields: "Value:" with "1" entered and "String:" with "Precision" entered. Below these fields are "Cancel" and "OK" buttons. To the right of the "Text Table Entry" window, there are three buttons: "Add", "Edit", and "Remove". Below these buttons, the text "Bytes Used: 16" is displayed.

Value	Text
0	Imo

Table Number: 1

Add  
Edit  
Remove


Bytes Used: 16

OK

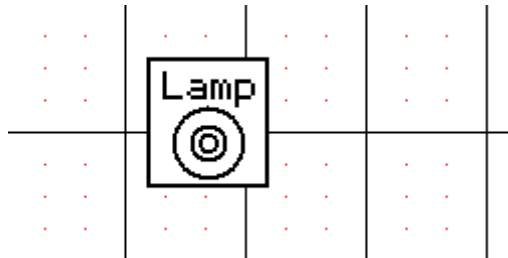
There can be up to 255 tables, each one can be used more than once.

Select a value and enter a corresponding message. The message will be displayed instead of the numeric value.

## Indicator Lamp

To insert an indicator lamp, click on the icon  and click it into the screen.

The indicator lamp can only be assigned to a bit (Q, M, S or T).



**Indicator Properties**

Controller Register

Address:

Name:

Indicator Type:

☒ Legend Plate ☐ 3D Bezel


Display Properties

There are several different options for display of lamp.

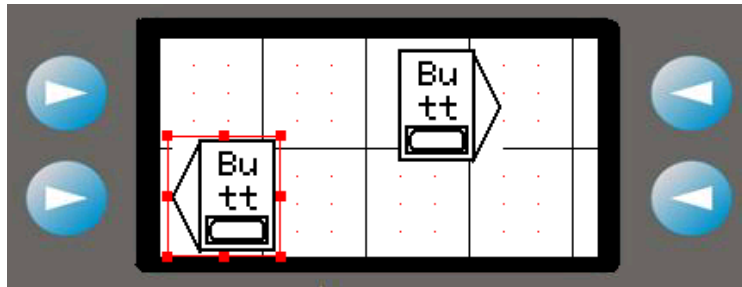
- Round
- Round
- Square
- Bulb
- Radio Button
- Check Box



## Button

There are four buttons that can be programmed through the screen editor; the remaining keys are programmed in the ladder editor. To insert a button, click on the icon  and click it into the screen.

The button will then allocate itself to the nearest key adjacent to the screen. Only one button can be assigned to a single key per screen.



The buttons are binary and can only be assigned to a bit

### Switch type

Standard  
Standard  
Round  
Square  
Rocker

### Action

Momentary  
Momentary  
ON  
OFF  
Toggle

**Switch Properties**

Controller Register  
Address:   
Name:

Keypress Source  
☒ Attach to nearest soft key  
☐ Auxiliary Register  
Address:   
Name:   
☐ Touch Screen

Switch Type:   
Action:   
☒ Legend Plate  
☐ 3D Bezel  
☐ Return to last screen after press  
☒ Show Inside Line Detail

Indicator Properties >>

Display Properties  
Attributes >>> Background Color >>>   
Legend >>> Line Color >>>

OK Cancel

The Keypress Source defaults at the nearest softkey (4 keys either side of the screen).

However they can be set to an alternative address in the *i*<sup>3</sup>.

**Switch State Properties**

Indicator Text  
☐ Enable Indicator Text  
On String:   
Off String:   
Font Type:   
☒ Text follows controller register (switch output)  
☐ Text follows indicator register


Color Indicator  
ON Color >>>   
OFF Color >>>   
☒ Color follows controller register (switch output)  
☐ Color follows indicator register

Indicator Register  
Address:   
Name:

OK Cancel

We can also edit the indicator properties

## Selector Switch

To insert a selector switch, click on the icon  and click it into the screen.

It will then select the nearest buttons to it on either side of the screen. Selector switches are limited to two positions.



A selector switch will be either one of two states.

**Selector Properties**

Controller Register  
Address: %M0001  
Name:

Keypress Source  
☒ Attach to nearest soft key  
☐ Auxiliary Register  
Address:   
Name:   
☐ Touch Screen

Selector Type: List Selector Vert  
Positions: Two

Items >>>


Display Properties  
Attributes >>> Legend >>>  
Background Color >>>   
Line Color >>>   
ON Color >>>   
OFF Color >>>

OK Cancel

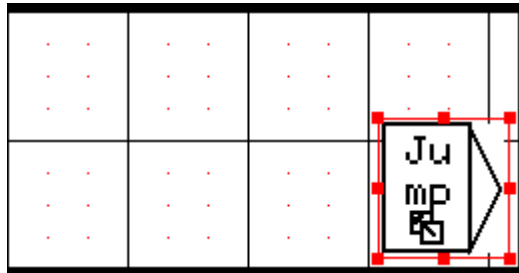
The address can be a bit or a register.

The items can be given meaningful names

## Screen Jump

Screen jumps are allocated like buttons but are for jumping between screens in a menu fashion. To insert a screen jump, click on the icon  and click it into the screen.

The jump button will allocate itself to the nearest key at either side.



Select what screen number to jump to and the display properties of the button.

**Screen Jump Properties**

Jump to Screen Number

Address / Number:  >

Name:

☐ Simulate ESC

☐ Allow ESC to Return

Keypress Source

☒ Attach to nearest soft key

☐ Auxiliary Register

Address:

Name:

☐ Cursor Selectable

☐ Touch

Display Properties

Attributes >>>

Legend >>>

Background Color >>>


Line Color >>>

OK Cancel

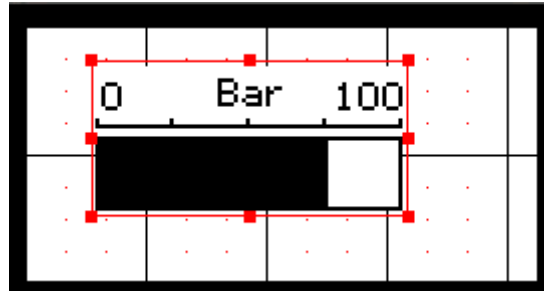
The Keypress Source defaults at the nearest softkey (4 keys either side of the screen).

However they can be set to another address in the *i<sup>3</sup>* or can even be selected using the cursor and arrow keys

## Bar Graph

To insert a bar graph, click on the icon  and click it into the screen.

To graphically display a register value on the screen. Click on the edge of the box and drag to make the bar graph bigger.



**Bar / Meter Properties**

Controller Register

Address:  Register Width:

Name:

Scale

☒ Show Scale Limits Maximum:  Minimum:

Font:  Ticks:

Display Properties

Attributes >>> Background Color >>>

Legend >>> Line Color >>>


Fill Color >>>

OK Cancel

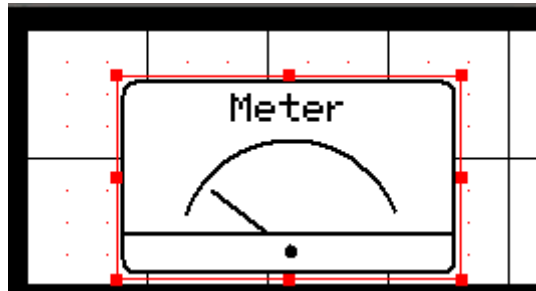
The bar graphs must be a word.

The bar should  
This can be shown  
screen.

## Meter Graph

To insert a meter graph, click on the icon  and click it into the screen.

Another option to the bar graph is to display the data within a meter.



**Bar / Meter Properties**

Controller Register

Address:  Register Width:

Name:

Scale

☐ Show Scale Limits Maximum:

Font:  Minimum:

Ticks:

Display Properties

Attributes >>> Background Color >>>


Legend >>> Line Color >>>

Fill Color >>>

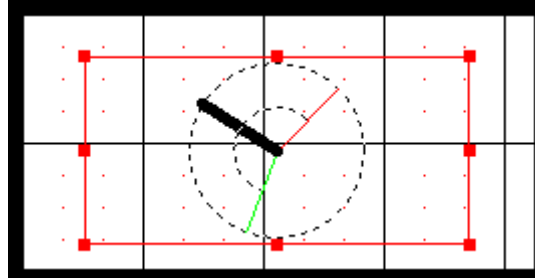
OK Cancel

The parameters are the same as with the bar graph.

## Gauge Graph

To insert a gauge graph, click on the icon  and click it into the screen.

The Gauge is more complex than a bar graph or meter. It is an automotive style gauge and can be placed over bitmaps.



**Gauge Properties**

Controller Register

Address:  Register Width:

Name:

Scale

☒ Clockwise ☐ Counter-Clockwise

Maximum:  Minimum:

Needle Width:  ☐ Display long needle

Start Angle:  End Angle:

While the object is selected for editing use SHIFT+UP or SHIFT+DOWN to adjust the start angle and SHIFT+LEFT and SHIFT+RIGHT to adjust the end angle.

Display Properties


Attributes >>> Needle Color >>>

OK Cancel

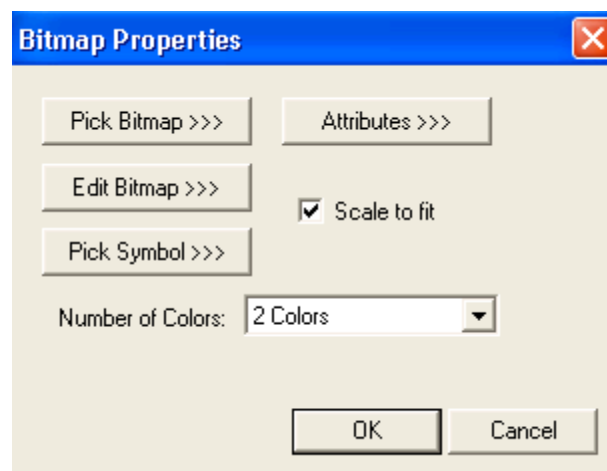
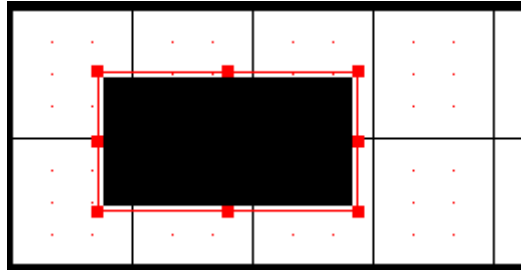
Select starting and ending angle and the needle will rotate within the set scale.

## Static Bitmap

A bitmap can be used as a screen backdrop, where a company logo can be inserted for example.


To insert a static bitmap, click on the icon  and click it into the screen.

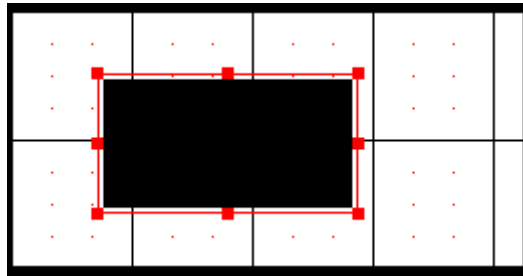
The *i<sup>3</sup>* has a mono screen; therefore the bitmap must comply with these conditions.



## Animation

The animation displays a series of bitmaps depending on the value of a register, double word, word, byte or bit.

To insert an animation, click on the icon  and click it into the screen.



**Animation Properties**

Controller Register

Address:  Register Width:

Name:

Frame Number:

None

Pick Frame >>>

Symbol Frame >>>

Edit Frame >>>

Delete Frame

Insert Frame

☒ Scale to fit


Number of Colors:

Attributes >>> OK Cancel

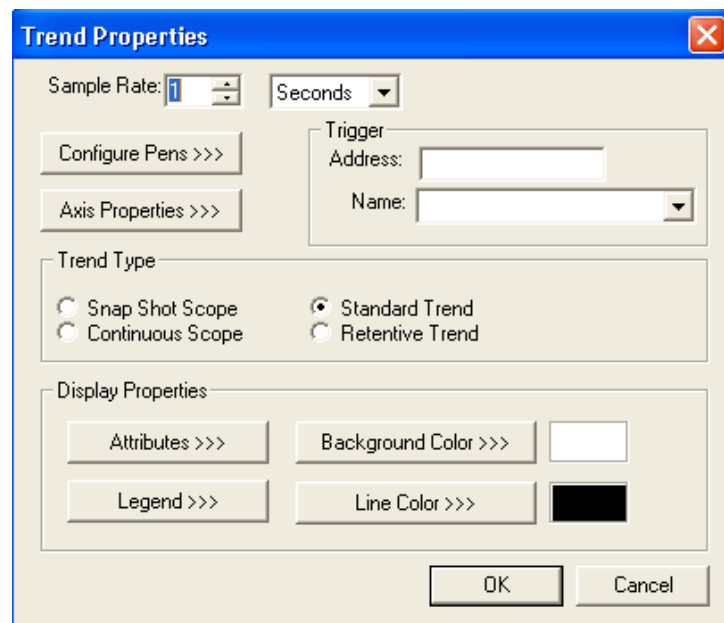
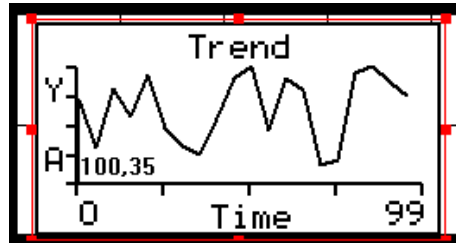
Select the bitmaps for the frames to match the value in the registers, i.e. a bit will have two frames.



## Data Trend

The data trend tracks data over a period of time. To insert a trend, click on the icon  and click it into the screen.

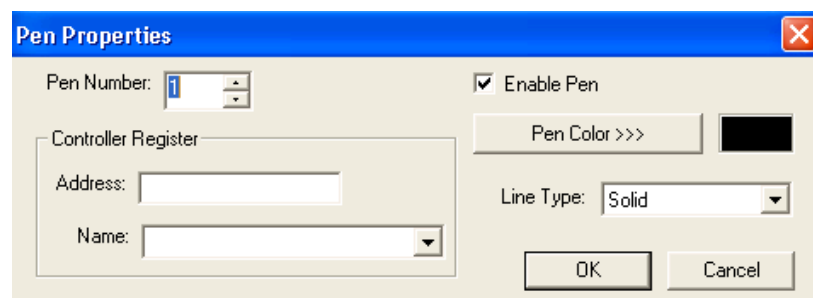
A data trend can track up to four registers over a set period of time.



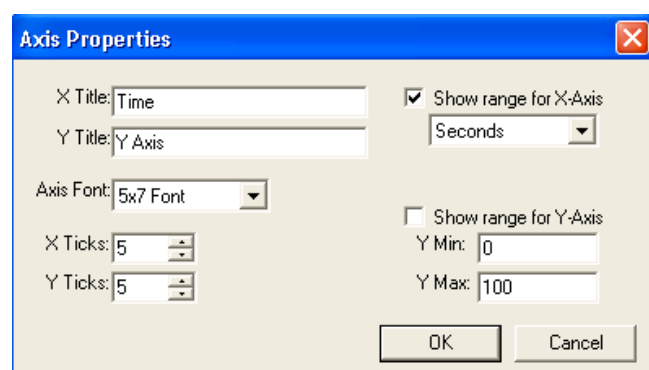
The sample can be in seconds, minutes or hours

The trigger address is required to activate the trend.

The trend can be 1 of 4 different types, see the help file for detailed information




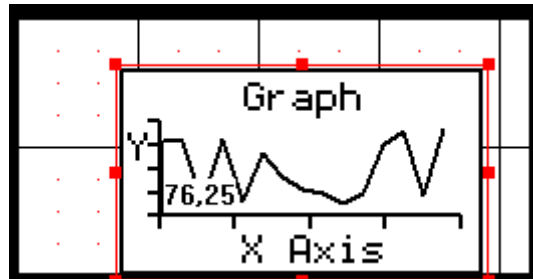
Up to 4 pens per trend can be edited. I.e. 4 data registers.



The axis titles can be edited to something more meaningful, and the scale can be adjusted

## X – Y Data Graph

To insert an X – Y Graph, click on the icon  and click it into the screen. The X-Y graph represents variation of a variable in comparison to variations in one or more variables.



**Graph Properties**

Number of Values to Plot:  (77)

Display Properties:

Trigger:

Address:

Name:

The trigger address is required to reset and refresh the plotting process.

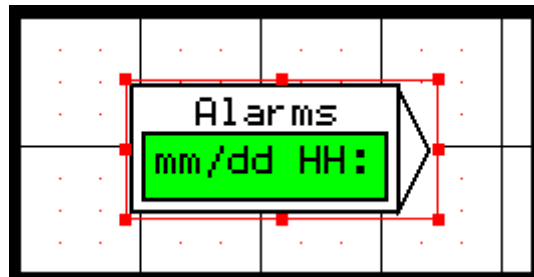
## Alarms

To insert an Alarm Log, click on the icon  and click it into the screen. There are two types of alarm:

1. Summary displays the alarm when it is active
2. History logs the alarm.

There are two steps to setting the alarm, first the button needs to be set up then the log itself.

The alarm will display a message and time stamp it.



**Alarm Object Properties**

**Type of alarm log**

Items to Display

☒ Summary ☐ History

☐ Display alarm button/icon only

☐ Unacked Only

☐ Allow Operator to Clear

**Details to display**

List Format

Font: 5x7 Font

☒ Date mm/dd

☒ Time HH:mm

☒ State (UNACK, ACK...)

Keypress Source

☒ Attach to nearest soft key

☐ Auxiliary Register

Address:

Name:

☐ Cursor Selectable

☐ Touch

Alarm Groups to Display

<input checked="" type="checkbox"/> 1	<input checked="" type="checkbox"/> 5	<input checked="" type="checkbox"/> 9	<input checked="" type="checkbox"/> 13
<input checked="" type="checkbox"/> 2	<input checked="" type="checkbox"/> 6	<input checked="" type="checkbox"/> 10	<input checked="" type="checkbox"/> 14
<input checked="" type="checkbox"/> 3	<input checked="" type="checkbox"/> 7	<input checked="" type="checkbox"/> 11	<input checked="" type="checkbox"/> 15
<input checked="" type="checkbox"/> 4	<input checked="" type="checkbox"/> 8	<input checked="" type="checkbox"/> 12	<input checked="" type="checkbox"/> 16

**Display Properties**

Attributes >>> Background Color >>>

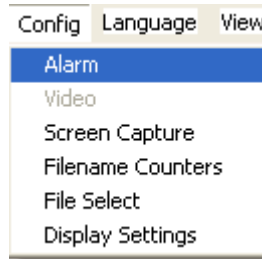
Legend >>> Line Color >>>

OK Cancel

There can be up to 16 different alarm groups

## Configure the Alarm Log

Select Alarm from the Config Menu drop down, to open the Alarm Log editor.



**Alarm Configuration**

Alarm Trigger Block

Address:  Max Numbers of Alarms:

Name:

Summary

☐ RTN implies ACK

History

☒ Log ACK  
☒ Log CLR  
☒ Log RTN

Alarm Point Configuration

Number	Group	Identifier String
1	1	*** Undefined Alarm1 ***
2	1	*** Undefined Alarm2 ***
3	1	*** Undefined Alarm3 ***
4	1	*** Undefined Alarm4 ***
5	1	*** Undefined Alarm5 ***

Copy  
 Paste  
 Edit

Summary List Text

Alarm:   
 ACK/CLR:   
 RTN:

Summary Button

UNACK:   
 ACT:   
 Empty:

History Button

Full:   
 Not Empty:   
 Empty:

OK Cancel

The alarm trigger can be 1 register or consecutive

Choose what to display in the History log

**Edit Alarm Point Configuration**


Identifier String:

Group:


OK Cancel

Alarm message to display and group related to.

## Recipe Editor Object

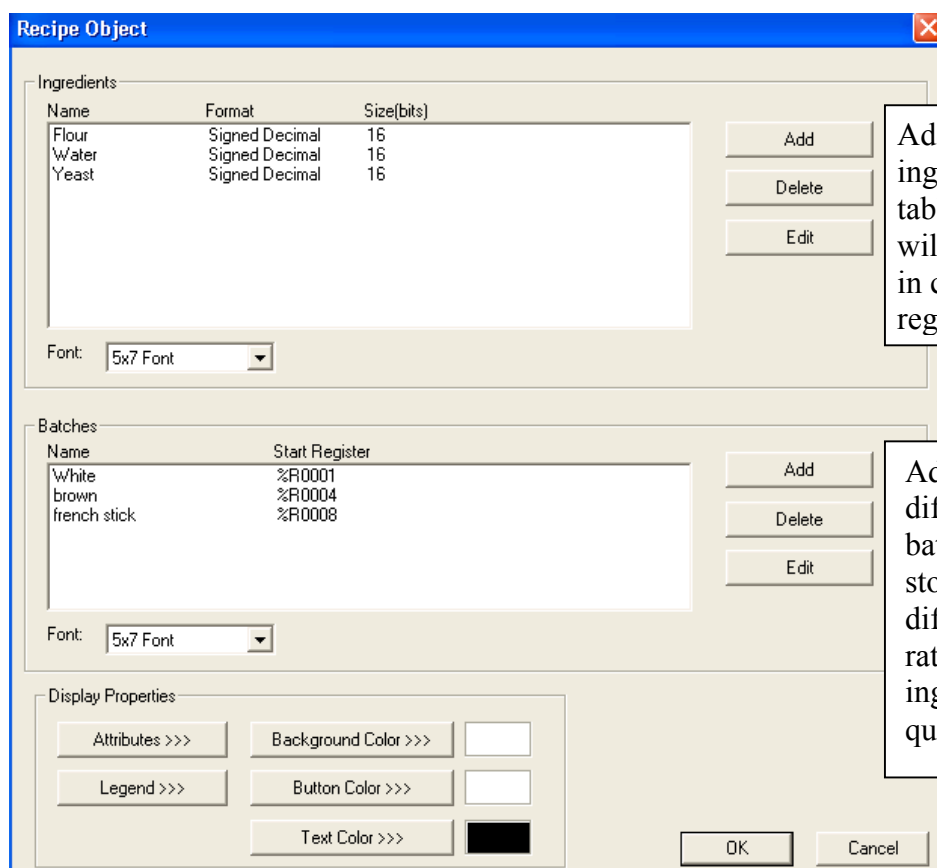
To insert a Recipe Editor Object, click the icon  on to the screen.

A recipe can be created to store the values of variables for different batch quantities.



Recipe Table

	Whit brow fren		
Flour	-###	-###	-###
Water	-###	-###	-###
Yeast	-###	-###	-###



**Recipe Object**

Ingredients

Name	Format	Size(bits)
Flour	Signed Decimal	16
Water	Signed Decimal	16
Yeast	Signed Decimal	16

Font: 5x7 Font

Batches

Name	Start Register
White	%R0001
brown	%R0004
french stick	%R0008

Font: 5x7 Font

Display Properties

Attributes >>> Background Color >>>

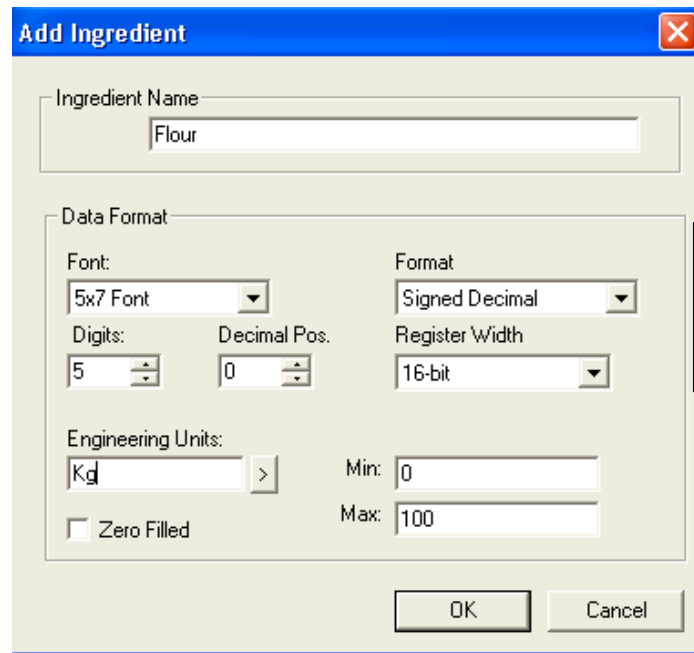
Legend >>> Button Color >>>

Text Color >>>

OK Cancel

Add the ingredients table. These will be stored in consecutive registers.

Add the different batches, to store the different rates of ingredient quantities



**Add Ingredient**

Ingredient Name:

Data Format

Font:  Format:

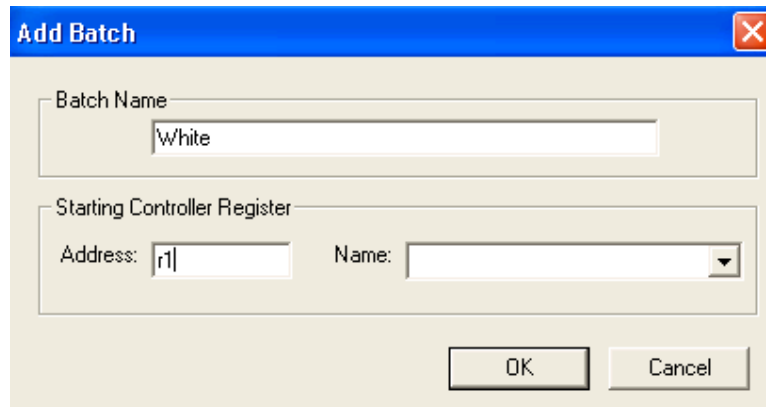
Digits:  Decimal Pos.:  Register Width:

Engineering Units:  > Min:  Max:

☐ Zero Filled

OK Cancel

Set up the ingredient display and scales.



**Add Batch**

Batch Name:

Starting Controller Register

Address:  Name:

OK Cancel

Add the batch properties and storage register.

This recipe function can be used in conjunction with move function blocks to move recipe data from one location to another.





IMO Precision Controls Limited  
1000 North Circular Road  
Staples Corner, London NW2 7JP  
United Kingdom  
Tel: +44 (0)20 8452 6444  
Fax: +44 (0)20 8450 2274  
Email: [imo@imopc.com](mailto:imo@imopc.com)  
Web: [www.imopc.com](http://www.imopc.com)



IMO Deutschland  
Für weitere Einzelheiten zu IMO  
Agenten und Distributoren in Ihrer  
Nähe schreiben Sie bitte ein E-mail  
an folgende Adresse:  
[imo@imopc.com](mailto:imo@imopc.com)



IMO Jeambrun Automation SAS  
Avenue du Mistral, ZI Athelia IV  
13705 La Ciotat Cedex, France  
Tel : 04 42 83 82 00  
Fax : 04 42 83 82 75  
Email : [info@imopc.fr](mailto:info@imopc.fr)  
Web : [www.imopc.fr](http://www.imopc.fr)



IMO Italia  
Viale A. Volta 127/a  
50131 Firenze, Italia  
Tel: 800 783281  
Fax: 800 783282  
Email: [info@imopc.it](mailto:info@imopc.it)  
Web: [www.imopc.it](http://www.imopc.it)



Cam Switches  
Din Terminals  
Drives  
Enclosures  
Fieldbus remote I/O  
Isolators & Switch Fuses  
MCB & RCD  
Motor Circuit Breakers  
Motor Control Gear  
Panel Meters  
Relays  
Signal Conditioning  
Sockets  
Timers  
Transformers &  
Power Supplies



Drives  
Intelligent Terminals/HMI  
Limit Switches  
Photoelectric Switches  
PLCs  
Proximity Switches  
Temperature Controls



Data Acquisition & Control  
Drives  
Intelligent Terminals/HMI  
Limit Switches  
Photoelectric Switches  
Proximity Switches  
PLCs  
Signal Conditioning  
Temperature Controls



Lightguards  
Safety Limit Switches  
Safety Relays



Jaguar VXM 0.37-500KW  
Jaguar VXSM 0.37-7.5KW  
Jaguar CUB 0.37-2.2KW



Audible devices  
Chip-on-Board  
Device programmers  
LEDs & 7 seg. displays  
PCB Terminal blocks  
Relays - automotive  
Relays - power  
Relays - signal  
Switches

All IMO products are tried, tested and approved  
to relevant international quality standards

