
Getting Started : The WinPAC ISaGRAF PAC

The WinPAC-8xx7 (or WP-8xx7 or WP-8x47) is the abbreviation of the WP-8147/8447/8847.
The WinPAC-8xx6 (or WP-8xx6 or WP-8x46) is the abbreviation of the WP-8146/8446/8846.

Important:

1. **WP-8xx7/8xx6 supports only High profile I-8K and I-87K I/O cards in its slot 0 to 7.**
Refer to WP-8xx7 CD: \napdos\isagraf\wp-8xx7\english_manu\wp-8xx7_datasheet.pdf
2. Please always set a fixed IP address to the WinPAC-8xx7. (No DHCP)
3. Please always set WP-8xx7's LAN2 as disabled if not using it (refer to appendix D).
4. Recommend to use the NS-205 or NS-208 Industrial Ethernet Switch for WP-8xx7 / 8xx6.

Legal Liability

ICP DAS CO., LTD. assumes no liability for any and all damages that may be incurred by the user as a consequence of this product. ICP DAS CO., LTD. reserves the right to change this manual at any time without notice.

ICP DAS CO., LTD. constantly strives to provide our customers with the most reliable and accurate information possible regarding our products. However, ICP DAS CO., LTD. assumes no responsibility for its use, or for any infringements of patents or other rights of third parties resulting from its use.

Trademark & Copyright Notice

The names of products and name of company are used for identification purposes only, and are the registered trademarks of their respective owners or companies.

Development Software

Two options:

- ISaGRAF: Ver. 3.4x (or Ver. 3.5x), IEC 61131-3 standard. LD, ST, FBD, SFC, IL & FC or
- Non-ISaGRAF: Microsoft EVC++4.0 or VS.NET 2008/2005/2003 (VB.net, C#.net)

Reference Guide

- ISaGRAF English User's Manual:

WinPAC-8xx7 CD: \napdos\isagraf\wp-8xx7\english_manu\ "user_manual_i_8xx7.pdf"

- ISaGRAF中文進階使用手冊:

WinPAC-8xx7 CD:

\napdos\isagraf\wp-8xx7\chinese_manu\ "chinese_user_manual_i_8xx7.pdf"

- **More from the Internet:** <http://www.icpdas.com/products/PAC/i-8000/isagraf.htm>

Technical Service:

Please contact local agent or email problem-report to service@icpdas.com .

FAQ : <http://www.icpdas.com/faq/isagraf.htm>

Written by Chun Tsai; Edited by Eva Li.

Copyright Jan. 2009, by ICP DAS CO., LTD. All Rights Reserved.

Table of Contents

GETTING STARTED : THE WINPAC ISAGRAF PAC.....	1
<i>Trademark & Copyright Notice</i>	<i>1</i>
<i>Development Software</i>	<i>1</i>
<i>Reference Guide</i>	<i>1</i>
<i>Technical Service:.....</i>	<i>1</i>
TABLE OF CONTENTS.....	2
REFERENCE GUIDE	5
I/O MODULES SELECTION GUIDE FOR WP-8X47 SERIES.....	6
PERFORMANCE COMPARISON TABLE 1	9
PERFORMANCE COMPARISON TABLE 2	10
SPECIFICATIONS: WP-8147/8447/8847	12
CHAPTER 1 : TYPICAL APPLICATION	1
1.1: New Redundant System with Ethernet I/O	1
1.2: New Hot-Swap and Redundant System.....	2
1.3: Modbus Master: RTU, ASCII, RS-232/485/422	3
1.4: Modbus RTU/TCP Slave Ports	3
1.5: Multiple Web HMI – Monitor & Control Everywhere!	4
1.6: Communicate With Other TCP/IP Server or UDP Client/Server Devices	4
1.7: Send Email with One Attached File	5
1.8: Data-Recorder & Data-Logger	5
1.9: Remote I/O Application	6
1.10: SMS: Short Message Service.....	6
1.11: Auto-report Acquisition & Control Data.....	7
1.12: Motion Control	8
1.13: Stress Monitoring Application of Constructions	8
1.14: Fast FRnet Remote I/O	9
1.15: Integrate with CAN/CANopen Devices & Sensors	10
CHAPTER 2 : SOFTWARE INSTALLATION.....	2-1
2.1: Step 1 - Installing The ISaGRAF Software	2-1
2.1.1: <i>When closing my ISaGRAF window on windows 2000, it holds. Why ?</i>	<i>2-3</i>
2.1.2: <i>One Quick way to avoid the “hold” problem on windows 2000.....</i>	<i>2-4</i>
2.2: Step 2 - Installing The ICP DAS Utilities For ISaGRAF.....	2-5
2.3: Step 3 - Installing The Web Page Editor	2-6
CHAPTER 3 : SETTING UP A WEB HMI DEMO.....	3-1
3.1: Web Demo List.....	3-1

3.2: Steps To Set Up A Web HMI Demo.....	3-2
3.2.1: Step 1 - Setup The Hardware	3-2
3.2.2: Step 2 - Setting The Web Options	3-2
3.2.3: Step 3 - Download ISaGRAF Project.....	3-3
3.2.4: Step 4 - Download Web Pages To The Wincon.....	3-6
3.2.5: Step 5 - Show Time.....	3-6
CHAPTER 4 : PROGRAMMING A WEB HMI EXAMPLE.....	4-1
4.1: Writing A Simple ISaGRAF Program.....	4-1
4.1.1: Open ISaGRAF-Project Management	4-2
4.1.2: Creating An ISaGRAF User's Group.....	4-3
4.1.3: Creating A New ISaGRAF Project	4-3
4.1.4: Declaring The ISaGRAF Project Variables	4-4
4.1.5: Assign Modbus Network Address No to Variables	4-7
4.1.6: Create The LD - "LD1" Program	4-9
4.1.7: Edit The "LD1" Program	4-10
4.1.8: Connecting The I/O	4-14
4.2: Compiling & Simulating The Example Project	4-17
4.2.1: Compiling Error Result In Different ISaGRAF Version	4-17
4.3: Download & Debug The Example Project.....	4-21
4.4: Design The Web Page	4-26
4.4.1: Step 1 – Copy The Sample Web HMI pages	4-26
4.4.2: Step 2 – Building The Main.htm.....	4-27
4.4.3: Step 3 – Adding Control Code To The Main.htm	4-32
4.4.4: Step 4 – Download Web HMI Pages To The Controller	4-38
CHAPTER 5 : WEB HMI BASICS.....	5-1
5.1: Basic Files For The Web HMI	5-1
5.2: Login.htm	5-2
5.3: Menu.htm	5-3
5.4: Main.htm	5-5
5.4.1: A Simple Main.htm Example.....	5-5
5.4.2: More About The refresh_data() Function And Dynamic Data.....	5-7
5.4.3: Post Data To The Controller.....	5-12
5.5: Multi-Pages	5-16
5.5.1: Level 2 And Level 3 Page.....	5-16
5.5.2: Switch One Page To One Another Page.....	5-17
5.6: Web Security.....	5-18
CHAPTER 6 : VB.NET 2008 PROGRAM RUNNING IN WINPAC-8XX7 ACCESS TO ISAGRAF VARIABLES	6-1
6.1 Create a New Project	6-1
6.2 Add Project Reference for an Application	6-2
6.3 Compiling an Application Program	6-5
6.4 QuickerNET.DLL	6-6
6.4.1 Digital R/W Functions	6-6
6.4.2 Analog R/W Functions	6-7
CHAPTER 7 : EVC++ PROGRAM RUNNING IN WINPAC ACCESS TO ISAGRAF VARIABLES.....	7-1

CHAPTER 8 : INDUSOFT PROJECT RUNNING IN WINPAC ACCESS TO ISAGRAF VARIABLES	8-1
CHAPTER 9 : ISAGRAF EXAMPLE PROGRAM	9-1
9.1: Get On-Line Help.....	9-1
9.2: Installing The ISaGRAF Programming Examples	9-4
APPENDIX A: HARDWARE SYSTEM & SETTING	AP-1
A.1: Applying Correct Power Supply	Ap-1
A.2: Modify The NET-ID & Modbus RTU Port Setting	Ap-2
A.3: Setting The IP Address For The WP-8xx7	Ap-3
A.4: Connecting Your PC To The WP-8xx7 Ethernet Port	Ap-4
A.5: Pin Assignment of COM1, COM2, COM3 and COM4 and Multi-Clients Connection to The WP-8xx7	Ap-5
A.6: Connecting PC To WP-8xx7 COM2 or COM3	Ap-6
A.7: Deleting the ISaGRAF Project From The WP-8xx7	Ap-7
A.8: Linking I-7000 and I-87K Modules For Remote I/O	Ap-8
A.9: Linking To An HMI Interface Device	Ap-9
A.10: Linking To Other Modbus Devices	Ap-10
APPENDIX B: UPGRADE WINPAC'S ISAGRAF DRIVER TO NEWER VERSION	AP-11
APPENDIX C: DIMENSION	AP-15
APPENDIX D: HOW TO ENABLE/DISABLE WP-8XX7'S LAN2	AP-16
APPENDIX E: USING EXPANSION RS-232 OR 485 OR 422	AP-17
APPENDIX F: SLOW DOWN ISAGRAF DRIVER'S SPEED	AP-22
APPENDIX G: SETUP MORE MODBUS RTU SALVE PORTS	AP-23
APPENDIX H: COMPILING ERROR RESULT IN DIFFERENT ISAGRAF VERSION	AP-24
APPENDIX I: USING RS-232 SERIAL/USB TOUCH MONITOR.....	AP-25

Reference Guide

ISaGRAF English User's Manual:

WinPAC-8xx7 CD: \napdos\isagraf\wp-8xx7\english_manu\ "user_manual_i_8xx7.pdf"

ISaGRAF 中文進階使用手冊:

WinPAC-8xx7 CD:

\napdos\isagraf\wp-8xx7\chinese_manu\ "chinese_user_manual_i_8xx7.pdf"

Industrial Ethernet Switch : NS-205/NS-208

http://www.icpdas.com/products/Switch/switch_list.htm

Power supply: http://www.icpdas.com/products/Accessories/power_supply/power_list.htm

DP-660 : 24 V / 2.5 A , 5 V / 0.5 A power supply (DIN-Rail mounting)

DP-665 : 24 V / 2.5 A , 5 V / 0.5 A power supply

DP-1200 : 24 V / 5 A power supply



Model: NS-205



Model: NS-208



Model: DP-660



Model: DP-1200



Model: DP-665

FAQ: www.icpdas.com > FAQ > Software > ISaGRAF for Frequently Asked Question

I/O Modules Selection Guide for WP-8x47 Series

WP-8x47 supports the **I-8K/I-87K High Profile** I/O modules and RS-485/FRnet remote I/O modules listed below. For new supporting I/O modules please refer to our web site listed below.

High Speed Local I/O Modules: Parallel Bus

I-8K High Profile Modules:

More at www.icpdas.com > Products > PAC - 8K & 87K I/O Modules

I-8K Analog I/O Modules	
I-8017HW	8-ch. Diff. or 16-ch. Single-ended, 14-bit, High Speed Analog Input Module (current input require external 125 Ω resistor) just like I-8017HS.
I-8024W	4-ch. Isolated Analog Output Module (+/-10 V, 0 ~ +20 mA)
I-8K Digital I/O Modules	
I-8037W	16-ch. Isolated Open Collector Output Module
I-8040W	32-ch. Isolated Digital Input Module
I-8040PW	32-ch. Isolated Digital Input with Low Pass Filter Module
I-8041W	32-ch. Isolated Open Collector Digital Output Module (Sink)
I-8041AW	32-ch. Isolated Open Collector Digital Output Module (Source)
I-8042W	16-ch. Isolated Digital Input & 16-ch. Isolated Open Collector Digital Output Module
I-8046W	16-ch. Isolated Digital Input Module
I-8050W	16-ch. Universal Digital I/O Module
I-8051W	16-ch. Non-isolated Digital Input Module
I-8052W	8-ch. Differential Isolated Digital Input Module
I-8053W	16-ch. Isolated Digital Input Module
I-8053PW	16-ch. Isolated Digital Input with Low Pass Filter Module
I-8054W	8-ch. Isolated Digital Input Module & 8-ch. Isolated Open Collector Digital Output Module
I-8055W	Non-isolated 8-ch. Digital Logic Input Module & 8-ch. Open Collector Digital Output Module
I-8056W	16-ch. Non-isolated Open Collector Output Module
I-8057W	16-ch. Isolated Open Collector Output Module
I-8058W	8-ch. Differential Isolated Digital Input Module, Max. AC/DC Input : 250V
I-8060W	6-ch. Relay Output Module, AC: 0.6 A @ 125 V , 0.3 A @ 250 V; DC: 2 A @ 30 V
I-8063W	4-ch. Diff. Isolated digital input & 4-ch. Relay output module, AC : 0.6 A @ 125 V ; 0.3 A @ 250 V
I-8064W	8-ch. Power Relay Output Module, AC: 5 A @ 250 V, DC: 5 A @ 30 V
I-8068W	4-ch. Form-A, 5 A @ 250 V _{AC} /28 V _{DC} & 4-ch. Form-C, 5 A (NO) /3 A (NC) @ 277 V _{AC} /30 V _{DC} Relay Output Module
I-8069W	8-ch. PhotoMOS Relay Output Module, Max. AC/DC: 1 A @ 60 V
I-8K Counter/Frequency Modules	
I-8080W	4/8-ch. Counter/Frequency Module, Isolated or TTL level. (Can measure 4-ch Encoder without Z-index)
I-8088W	8-ch. PWM Output and 8-ch. isolated DI Module

I-8K Communication Modules	
I-8112iW	2-ch. isolated RS-232 expansion module
I-8114W	4-ch. non-isolated RS-232 expansion module
I-8114iW	4-ch. isolated RS-232 expansion module
I-8142iW	2-ch. isolated RS-422/485 expansion module
I-8144iW	4-ch. isolated RS-422/485 expansion module
I-8172W	2-port FRnet module

RS-485 Remote I/O Modules: Serial Interface; HOT-SWAP

I-87K High Profile Modules:

More at www.icpdas.com > Products > PAC - 8K & 87K I/O Modules

I-87K Analog I/O Modules	
I-87005W	8-ch. Thermistor Input, 16-bit, 8 Hz (Total) and 8-ch. digital output module with Open Wire Detection
I-87013W	4-ch., 16-bit, 10 Hz (Total), 2/3/4 Wire RTD Input Module with Open Wire Detection
I-87015	7-ch., 16-bit, 12 Hz (Total), RTD Input Module with Open Wire Detection (for short sensor distance)
I-87015P	7-ch. RTD Input Module with 3-wire RTD lead resistance elimination and with Open Wire Detection (for long sensor distance)
I-87017R	8-ch. Diff. , 16/12-bit, 10/60 Hz (Total) Analog Input Module with 240 V _{rms} Over Voltage Protection, Range of -20 ~ +20 mA Requires Optional External 125 Ω Resistor
I-87017RC	8-ch. Diff. , 16/12-bit, 10/60 Hz(Total) Current Input Module
I-87017W	8-ch. Analog Input Module
I-87017W-A5	8-ch. High Voltage Input Module
I-87018R	8-ch. Thermocouple Input Module. Recommend to use the better I-87018Z.
I-87018W	8-ch. Thermocouple Input Module. Recommend to use the better I-87018Z.
I-87018Z	10-ch. Diff. , 16-bit, 10 Hz (Total), Thermocouple Input Module with 240 V _{rms} Over Voltage Protection, Open Wire Detection, Range of +/-20 mA, 0~20 mA, 4~20 mA requires Optional External 125 Ω Resistor
I-87019R	8-ch. Diff. , 16-bit, 8 Hz (Total), Universal Analog Input Module with 240 V _{rms} Over Voltage Protection, Open Wire Detection (V, mA, Thermocouple; Range of -20 ~ +20 mA need to set Jumper on board)
I-87024C	4-ch. 12-bit channel to channel isolated current output module with open-wire detection
I-87024W	4-ch. 14-bit analog output module (0 ~ +5 V, +/-5 V, 0 ~ +10 V, +/-10 V, 0 ~ +20 mA, +4 ~ +20 mA)
I-87028C	8-ch. 12-bit current output module
I-87K Digital I/O Modules	
I-87040W	32-ch. Isolated Digital Input Module
I-87041W	32-ch. Sink Type Open Collector Isolated Digital Output Module
I-87046W	16-ch. Non-Isolated Digital Input Module for Long Distance Measurement
I-87051W	16-ch. Non-Isolated Digital Input Module
I-87052W	8-ch. Diff. , Isolated Digital Input Module
I-87053PW	16-ch. Isolated Digital Input Module with 16-bit Counters
I-87053W	16-ch. Isolated Digital Input Module

I-87053W-A5	16-ch. 68 ~ 150 V _{DC} Isolated Digital Input Module
I-87054W	Isolated 8-ch. DI and 8-ch. Open Collector DO Module
I-87055W	Non-Isolated 8-ch. DI and 8-ch. Open Collector DO Module
I-87057W	16-ch. Open Collector Isolated Digital Output Module
I-87058W	8-ch. 80~250 V _{AC} Isolated Digital Input Module
I-87059W	8-ch. Differential 10-80 V _{AC} Isolated Digital Input Module
I-87063W	4-ch. Diff. Isolated Digital Input and 4-ch. Relay Output Module 5 A (NO) / 3 A(NC) @ 5 ~ 24 V _{DC} ; 5 A(NO) / 3 A(NC) @ 0 ~ 250 V _{AC}
I-87064W	8-ch. Relay Output Module, 5 A (47~63 Hz) @ 0~ 250 V _{AC} ; 5 A @ 0~ 30 V _{DC}
I-87065W	8-ch. AC SSR Output Module, AC: 1.0 A _{rms} @ 24 ~ 265 V _{rms}
I-87066W	8-ch. DC SSR Output Module , DC: 1.0 A _{rms} @ 3 ~ 30 V _{DC}
I-87068W	4-ch. Form A Relay Output and 4-ch. Form C Relay Output Module Form A: 8 A @ 250 V _{AC} ; 8 A @ 28 V _{DC} ; Form C: 5 A (NO) / 3 A (NC) @ 277 V _{AC} ; 5 A(NO) / 3 A(NC) @ 30 V _{AC}
I-87069W	8-ch. PhotoMOS Relay Output Module, Max. AC/DC: 0.13 A @ 350 V
I-87K Counter/Frequency Modules	
I-87082W	2-ch. Counter/Frequency Module, Isolated or Non-isolated Inputs

More RS-485 Remote I/O Modules at :

I-7000 : www.icpdas.com > Products > Remote I/O Modules/Units > I-7000 Modules > Selection Guide

M-7000 : www.icpdas.com > Products > Remote I/O Modules/Units > M-7000 Modules > Selection Guide

More FRnet Remote I/O Modules at :

FRnet I/O: www.icpdas.com > Products > Remote I/O Modules/Units > FRnet Remote I/O Modules > Selection Guide

More RS-485 Remote Hot-Swap Expansion Unit at :

RU-87P1/2/4/8 : www.icpdas.com > Products > Remote I/O Modules/Units > RS-485 Remote I/O Unit > Selection Guide

More RS-485 Remote Expansion Unit at :

I-87K1/4/5/8/9: www.icpdas.com > Products > Remote I/O Modules/Units > RS-485 Remote I/O Unit > Selection Guide

Performance Comparison Table 1

PACs	CPU	Normal running Speed (Normal PLC scan-time)	Normal Speed for floating point calculation (scan-time)	Ethernet	ISaGRAF code size limitation (bytes)	Memory for running program (bytes)
WP-8147 WP-8447 WP-8847	PXA270 520 MHz or compatible	About 10~30 (3~15 ms)	About 10~30 (3~15 ms)	2 ports 10/100 Mbps	1 MB	About 20~40 MB
W-8347 W-8747	Strong-ARM 206 MHz or compatible	About 10~20 (3~15 ms)	About 10~20 (3~15 ms)	2 ports 10/100 Mbps	1 MB	About 20~40 MB
W-8337 W-8737	Strong-ARM, 206 MHz or compatible	About 10~20 (3 ~15 ms)	About 10~20 (3 ~15 ms)	1 port 10 Mbps	1 MB	About 20~40 MB
iP-8447 iP-8847	80186, 80 MHz or compatible	About 4 (2~25 ms)	About 0.8 (10~125 ms)	2 ports 10/100 Mbps	64 KB	About 768 KB
I-8437-80 I-8837-80	80186 80 MHz or compatible	About 4 (2~25 ms)	About 0.8 (10~125 ms)	1 port 10 Mbps	64 KB	About 512 KB
I-8437 I-8837	80188 40 MHz or compatible	About 1 (5~100 ms)	About 0.2 (25~500 ms)	1 port 10 Mbps	64 KB	About 512 KB
I-8417 I-8817	80188 40 MHz or compatible	About 1 (5~100 ms)	About 0.2 (25~500 ms)	No	64 KB	About 512 KB
μPAC- 7186EG	80186 80 MHz or compatible	About 4 (2~5 ms)	About 0.8 (10~125 ms)	1 port 10/100 Mbps	64 KB	About 640 KB
I-7188EG	80188 40 MHz or compatible	About 1 (5~100 ms)	About 0.2 (25~500 ms)	1 port 10 Mbps	64 KB	About 512 KB
I-7188XG	80188 40 MHz or compatible	About 1 (5~100 ms)	About 0.2 (25~500 ms)	No	64 KB	About 512 KB

Note: I-8x37/W-8x37 has phased out. Please select compatible iP-8x47/WP-8x47.

W-8x47 will be phased out by the end of Feb. 2009.

Performance Comparison Table 2

PACs	μPAC			iPAC			WinCon	WinPAC
	I-7188 XG	I-7188 EG	μPAC-7186EG	I-8417/ I-8817	I-8x37/ I-8x37-80 *1	iP-8447 iP-8847 *2	W-8x47 W-8x37 *1	WP-8147 WP-8447 WP-8847*2
Support Ethernet I/O (I-8KE4-MTCP I-8KE8-MTCP)	NO			NO			Yes	
Send E-mail (file attached)	NO		Yes *3	NO		Yes	Yes	
Max. amount for linking I-7K/ 87K Remote I/O module (Only 1 port)	64						255 (COM3)	255 (COM2, 3)
	(COM2, 3)		(COM3, 4)		(COM 2, 3, 4)			
Modbus Master Function Block Max. amount	64 (total)		128 (total)	64 (total)		128 (total)	256 (per port)	
Available Modbus Master COM Port (Max. mount) *4	(2 ports)						(10 ports)	
	COM 2, 3	COM 1, 2, 3		COM 1, 3, 4, 5		COM 1~5	COM 2, 3, 5~14	COM 1~14
Available Modbus Slave COM Port (Max. mount) *4	(2 ports)						(5 ports)	
	COM1 or 2/3			COM 1, 2	COM 1, 3	COM 1 or 2/3	COM 2 or 3, 5~8	COM 1, 2/3, 4~8
Modbus TCP/IP Connections	0	4	6	0	4	6	32 *5	
Modbus Address Range	1~4095			1~4095			1~8191	
Data Exchange	Fbus	Fbus, Ebus		Fbus	Fbus, Ebus		Ebus	
Support FRnet I/O	No			No		Yes *6	Yes *6	
Support CAN/CANopen	No		Yes *7	No		Yes *7	Yes *7	
Support VW Sensor	No			Yes			Yes	
Support Redundant Ethernet Port	No			No		Yes *8	Yes *8	
Support Mbus24r & mbus24r1 Function Block	No		Yes	No		Yes	Yes	
Support Mbus_xr & Mbus_xr1	No			No			Yes *9	
Support New Redundant System	No			No			Yes *10	

Annotations:

- *1. I-8x37/I-8x37-80 represents the products of I-8437/8837/8437-80/8837-80.
W-8x47 represents the products of W-8347/W-8747.
W-8x37 represents the products of W-8337/W-8737.
- *2. iP-8447/iP-8847 will be abbreviated as iP-8x47.
WP-8147/WP-8447/WP-8847 will be abbreviated as WP-8x47.
- *3. μ PAC-7186EG has to use an extra X607/608 battery backup SRAM expansion card for sending E-mail with an attached file, or it can only send E-mail without attached file.
- *4. I-8000's COM5~20 & W-8x47/ 8x37's COM5~14 resides at the I-8112/8114/8142/8144/ 8142i expansion modules ;

iP-8x47's COM5~20 resides at the I-8112iW/ 14W/ 14iW/ 42iW/ 44iW expansion modules;

WP-8x47's COM5~14 resides at the I-8112iW/ 14W/ 14iW/ 42iW/ 44iW expansion modules;

I-7188/ μ PAC-7186's COM3 ~ 8 resides at the X-5xx X-board expansion boards.
- *5. The W-8xx7 with driver version 4.02 or older version only supports 8 Modbus TCP/IP connections, while supports up to 32 Modbus TCP/IP connections since the version 4.03.

If the controller is W-8347/8747 (two Ethernet ports), its OS image must update to the version released on July, 1, 2008 to ensure the network communications is correct.

Please refer to www.icpdas.com > FAQ > Software > ISaGRAF > 095 for more information.
- *6. iP-8x47 support Max. **4** pcs. of I-8172W (Max ch. 1024 D/I and 1024 D/O).
W-8x47/8x37 support Max. **7** pcs. of I-8172W (Max ch.1792 D/I and 1792 D/O).
WP-8x47 support Max. **8** pcs. of I-8172W (Max ch.2048 D/I and 2048 D/O).
- *7. μ PAC-7186EG, iP-8x47, WP-8x47 and W-8xx7 supports the I-7530 (RS-232 to CAN converter) to connect to other CAN/CANopen devices.
- *8. If the cable of one Ethernet port is broken or damaged, the PC/HMI can communicate with the other Ethernet port by Modbus TCP/IP protocol.
- *9. The Mbus_xr and Mbus_xr1 can read max. 120 words or 60 long integers or 60 real values. Please refer to www.icpdas.com > FAQ > Software > ISaGRAF > FAQ-101 for more information.
- *10. Only the WP-8x47 and W-8x47 support new redundant system, the W-8x37 doesn't support it.

Specifications: WP-8147/8447/8847

■ System Software

OS	Windows CE 5.0
.Net Compact Framework	2.0
Embedded Service	FTP server, Web server

■ Development Software

ISaGRAF Software	ISaGRAF Version 3 : IEC 61131-3 standard. Languages: LD, ST, FBD, SFC, IL & FC
Max. Code Size	WP-8x47 accepts max. 1 MB ISaGRAF code size (Appli.x8m must < 1 MB)
Non-ISaGRAF	Options: Microsoft EVC++ 4.0 or VS.NET 2003/2005/2008 (VB.NET2003/2005/2008, C#.NET 2003/2005/2008)

■ Web Service

Web HMI	Support Web HMI function, PC running Internet Explorer can access to the WP-8x47 via local Ethernet, Internet or dial Modem to monitor and control.
Security	Web HMI supports three level username and password protection

■ Power Supply

Input Range	+10 ~ +30 V _{DC} (unregulated),
Isolation	1 kV
Redundant Power Inputs	Yes, with one power relay (1 A @ 24 V _{DC}) for alarm
Capacity	WP-8147: 1.0 A, 5 V supply to CPU and backplane; 0.6 A, 5 V supply to I/O expansion slots, total 8 W WP-8447: 1.1 A, 5 V supply to CPU and backplane; 4.9 A, 5 V supply to I/O expansion slots, total 30 W WP-8847: 1.2 A, 5 V supply to CPU and backplane; 4.8 A, 5 V supply to I/O expansion slots, total 30 W
Consumption	WP-8147: 7.3 W (0.3 A @ 24 V); WP-8447: 9.1 W (0.38 A @ 24 V); WP-8847: 9.6 W (0.4 A @ 24 V);

■ General Environment

Temperature	Operating Temperature: -25°C ~ +75°C (-13° F ~ +167° F) Storage Temperature: -30°C ~ +85°C (-40° F ~ +176° F)
Humidity	5% ~ 95% RH (non-condensed)

■ System

CPU	CPU: PXA270, 32-bit and 520 MHz or compatible
Dual Watchdog Timer	Watchdog: Dual Watchdog Timer
RTC (Real Time Clock)	Gives hour, minute, sec, day of week, day of month, month & year, valid up from 1980 to 2079
EEPROM	16 KB; Data retention: 40 years. 1,000,000 erase/write cycles
SRAM	128 MB
Dual Battery Backup SRAM	512 KB (for 5 years data retain while power off)
FLASH	96 MB (64 MB for OS image, 31 MB for built-in Flash disk, 1 MB for registry)

Expansion FLASH Memory	microSD socket with 1 GB FLASH card
Serial Number	Yes, 64-bit hardware unique serial number
NET ID	1 ~ 255, set by software
Rotary Switch	Yes (0~9)
DIP Switch	WP-8447 & WP-8847: Yes, 8 bits DIP Switch; WP-8147 : no DIP Switch
I/O Slots	WP-8147: 1 slot (slot0); WP-8447: 4 slots (slot0 ~ slot3); W-8747: 8 slots (slot0 ~ slot7); Accept High Profile I-8K Parallel & High Profile I-87K Serial I/O boards I/O Module Hot Swap Ability : for High Profile I-87K only

■ VGA & Communication Ports

VGA Port	1 VGA port (800 x 600 resolution)
Dual Ethernet Ports	2 Ethernet Ports: LAN1 & LAN2, 10/100 Base-TX Ethernet Controller (Auto-negotiating, Auto MDI/MDI-X, LED indicator) Please use NS-205/NS-208 Industrial Ethernet Switch.
USB Port	1 USB 1.1 Host ports for USB mouse, keyboard or USB drive
COM0	Internal communication with the I-87K High Profile modules in slots
COM1	RS-232 (RxD, TxD, and GND); Non-isolation; Speed: 115200 bps max.
COM2	RS-485 (D2+, D2-; self-tuner ASIC inside); Speed: 115200 bps max. Isolation : 2500 V _{DC} for 1 slot WP-8147; 3000 V _{DC} for 4/8 slots WP-8447/8847
COM3	RS-232/RS-485 (RxD, TxD, CTS, RTS and GND for RS-232, DATA+ and DATA- for RS-485); Non-isolation; Speed: 115200 bps max. (WP-8147 has no COM3 & COM4.)
COM4	RS-232 (RxD, TxD, CTS, RTS, DSR, DTR, CD, RI and GND); Non-isolation; Speed: 115200 bps max. (WP-8147 has no COM3 & COM4.)

■ Motion

Motion Control	WP-8447/8847 integrate with one I-8091W (2-axis) or two I-8091W (4-axis) can do motion control
----------------	--

■ PWM Output

Pulse Width Modulation	8 ch. max. 250 Hz max. For Off=2 & On=2 ms . Output square curve: Off: 2 ~ 32766 ms, On: 2 ~ 32766 ms. Optional DO Boards: I-8037W, 8041W, 8041AW, 8042W, 8050W, 8054W, 8055W, 8056W, 8057W, 8060W, 8063W, 8064W, 8068W, 8069W (Relay Output boards can not generate fast square pulse)
------------------------	---

■ Counter, Encoder, Frequency

Parallel DI Counter	8 ch. max. for 1 controller. Counter val: 32 bit. 250 Hz max. Min. ON & OFF width must > 2 ms. Optional DI Boards: I-8040W, 8040PW, 8042W, 8048W, 8050W, 8051W, 8052W, 8053W, 8053PW, 8054W, 8055W, 8058W, 8063W.
Serial DI Counter	Counter input: 100 Hz max. Counter value: 0 ~ 65535 (16 bit) Optional Serial I-87K DI Boards: I-87040W, 87046W, 87051W, 87052W, 87053W, 87053W-A5, 87054W, 87055W, 87058W, 87059W, 87063W.
Remote DI Counter	All remote I-7000 & I-87K DI modules support counters. 100 Hz max. value: 0 ~ 65535
High Speed Counter	I-87082W: 100 kHz max. 32 bit; I-87088W: 500 kHz max. 32 bit; I-8080W:

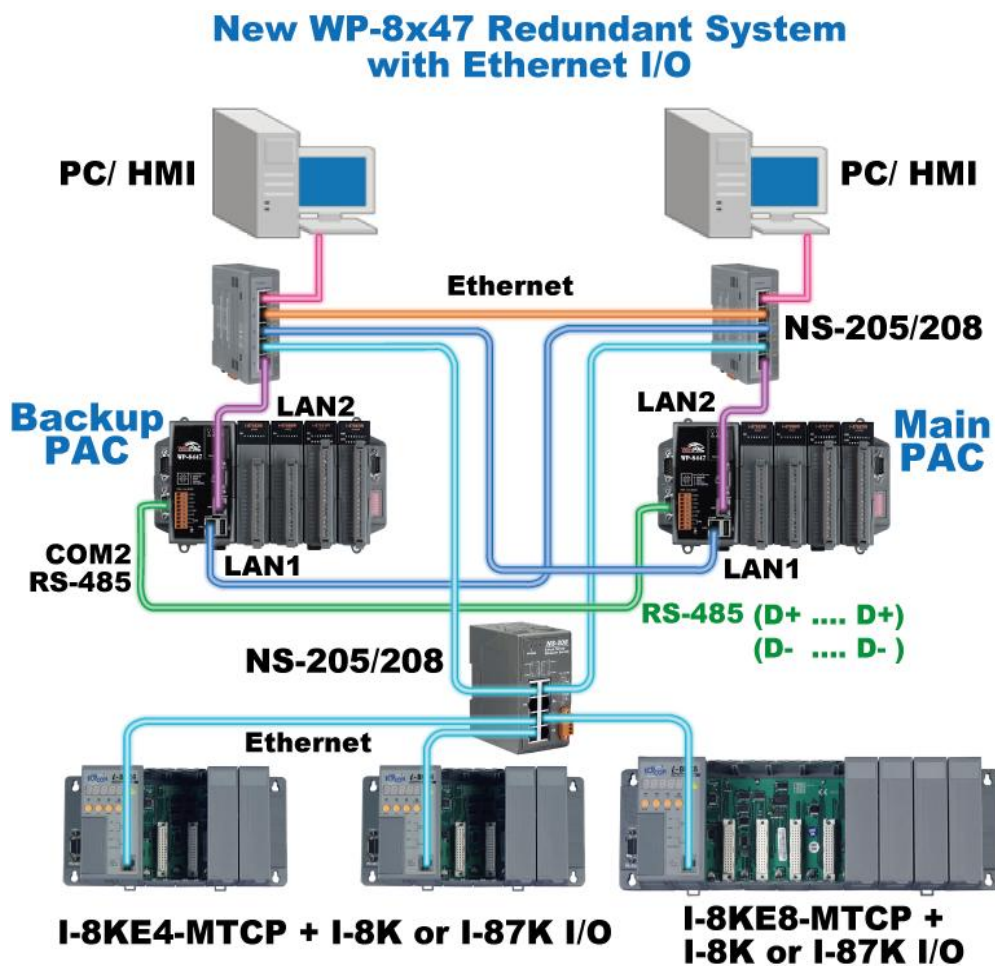
	450 kHz max. 32 bit
Encoder	I-8080W: 4-ch encoder, can be dir/pulse, or up/down or A/B phase (Quard. mode), Not support Encoder Z-index.
Frequency	I-87082W: 2-ch, 1 Hz ~ 100 kHz; I-87088W: 8-ch, 0.1 Hz ~ 500 kHz; I-8080W: 8-ch, 1 Hz ~ 450 kHz;
■ Protocols	
Modbus RTU / ASCII Master Protocol (Multi-Port)	Up to 10 COM Ports (COM1, 2, 3, 4 and <u>COM5~COM14 if multi-serial port boards are plugged in</u>) can support multi-ports of Modbus RTU / ASCII Master protocol to connect to other Modbus Slave devices. (WP-8147 has no COM3 & COM4.)
Modbus RTU Slave Protocol	Up to 5 COM Ports (COM1, one of COM2/3, COM4 and COM5 ~ COM8) can support Modbus RTU Slave protocol for connecting ISaGRAF, PC/HMI/ OPC Server & HMI panels. (WP-8147 has no COM3 & COM4.)
Modbus TCP/IP Slave Protocol	2 Ethernet Ports all support Modbus TCP/IP Slave protocol for connecting ISaGRAF & PC/HMI. LAN1 & LAN2 support total up to 32 connections. (If WP-8x47 uses 1 connection to connect each PC/HMI, it can connect up to 32 PC/HMI; If WP-8x47 uses 2 connections to connect each PC/HMI, it can connect up to 16 PC/HMI; ...) When one Ethernet port is broken, the other one can still connect to PC/HMI.
Web HMI Protocol	Ethernet Ports for connecting PC running Internet Explorer
I-7000 & I-87K RS-485 Remote I/O	One of COM2, COM3 supports <u>I-7000 I/O modules, I-87K base + I-87K Serial I/O boards and RU-87P1/2/4/8 + I-87K High Profile I/O boards as Remote I/O</u> . Max. 255 I-7000/87K Remote I/O modules for one controller. (WP-8147 has no COM3 & COM4.)
M-7000 Series Modbus I/O	Max. 10 RS-485 ports (COM1~4 & <u>COM5~COM14 if multi-serial port boards are plugged in</u>) can support M-7000 series Modbus I/O. Each port can connect up to 32 M-7000 Modules.
Modbus TCP/IP I/O	LAN2 supports ICP DAS Ethernet I/O : I-8KE4-MTCP and I-8KE8-MTCP listed at www.icpdas.com > FAQ > Software > ISaGRAF Ver.3 (English) FAQ-042. If LAN2 is broken, it will switch to LAN1 automatically to continuously work. (This need LAN1 & LAN2's IP are set in the same IP domain)
FRnet I/O	Support max 8 pcs. I-8172W boards in slot 0 to 7 to connect to FRnet I/O modules, like FR-2053, FR-2057 FR-32R, FR-32P, at www.icpdas.com > FAQ > Software > ISaGRAF Ver.3 (English) FAQ-048. Each I-8172w board can connect up to 256 DI plus 256 DO channels.
Send Email	Supports mail_snd and mail_set functions to send email with one attached file via Ethernet port.
Ebus	To exchange data between ICP DAS's ISaGRAF Ethernet PACs via Ethernet port. (LAN2 Port only)
SMS: Short Message Service	One of COM4 (or COM5 <u>if multi-serial port board is plugged in</u>) can link to a GSM Modem to support SMS. User can request data/control the controller by cellular phone. The controller can also send data & alarms to user's cellular phone. Optional GSM Modems: Supreme (850/900/1800/1900 GSM/ GPRS External Modem)
User Defined Protocol	User can write his own protocol applied at <u>COM1~COM4 & COM5~COM14(if multi-serial port boards are plugged in)</u> by Serial

	communication function blocks.
Modem_Link	WP-8x47 series does not support Modem_Link.
MMICON/LCD	<u>COM4</u> or <u>COM5(if I-8112W/8114W is found)</u> supports ICP DAS's MMICON. The MMICON is featured with a 240 x 64 dot LCD & a 4 x 4 Keyboard to display picture, string, integer, float, & input a char, string, integer & float.
UDP Server & UDP Client : Exchange Message & Auto-Report	LAN1 or LAN2 support UDP Server and UDP Client protocol to send / receive message to / from PC/HMI or other devices. For example, to automatically report data to InduSoft's RXTX driver.
TCP Client : Exchange Message & Auto-Report	LAN1 or LAN2 support TCP Client protocol to send / receive message to / from PC/HMI or other devices which support TCP server protocol. For example, to automatically report data to InduSoft's RXTX driver, or to connect a location camera.
New Hot-Swap and Redundant System	<p>This redundant system has setup two "Active IP" address point to the active WP-8x47's LAN1 and LAN2 ports always. One or two or more PC / HMI / SCADA can communicate with this redundant system via one of the two given active IP. So the PC / HMI / SCADA can access to the system easily without any notice about which WP-8x47 is currently active.</p> <p>Moreover, the new redundant system can integrate with the RU-87P4 and RU-87P8 expansion unit plus the I-87K high-profile I/O cards to support the hot-swap application. If the I/O card is damaged, the maintenance person just takes one good-card with same model number to hot-swap the damaged one without stopping this redundant system.</p> <p>Please refer to www.icpdas.com > FAQ > Software > ISaGRAF Ver.3 (English) FAQ-093.</p>
CAN/CANopen	<p>WP-8x47 can use its COM1,3,4 or <u>COM5~COM14 resides at the I-8112iW/8114W/8114iW RS-232 expansion board</u> to connect one I-7530 : the RS-232 to CAN converter to support CAN and CANopen devices and sensors. One WP-8x47 supports max.10 RS-232 ports to connect max.10 I-7530.</p> <p>Please refer to www.icpdas.com > FAQ > Software > ISaGRAF Ver.3 (English) FAQ-086.</p>

Chapter 1 : Typical Application

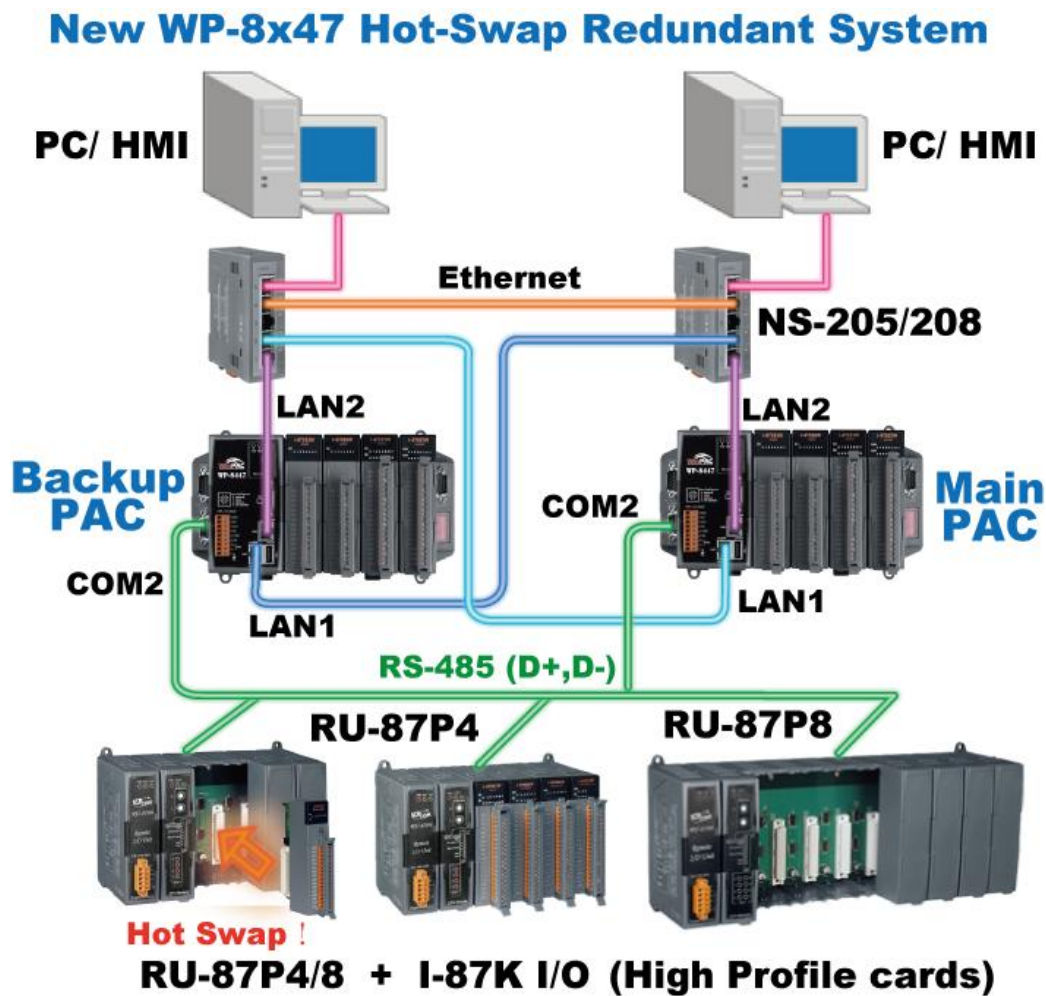
1.1: New Redundant System with Ethernet I/O

- If one Ethernet cable is broken or damaged, the other one will still handle the Ethernet I/O and exchange data with the other redundant controller.
- The scan of Ethernet I/O is much faster than that of RS-485 I-7000 or I-87K I/O.
- More at www.icpdas.com > FAQ > Software > ISaGRAF Ver.3 (English) - 093, 042



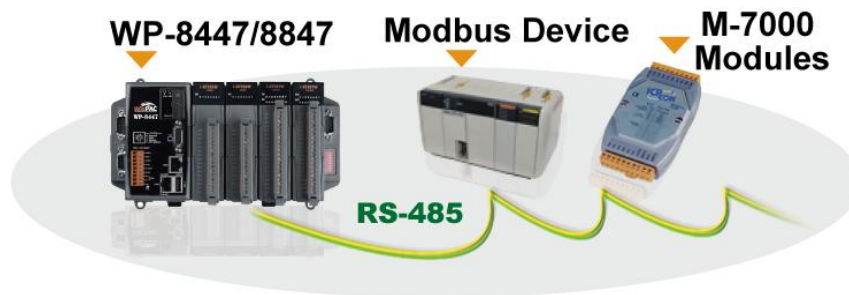
1.2: New Hot-Swap and Redundant System

- If one Ethernet cable of WP-8x47 is broken or damaged, the other one will still work.
- If one controller is dead, the other one will take over the control of the RS-485 I/O.
- PC/HMI can connect to this redundant system by one or two active IP.
- More at www.icpdas.com > FAQ > Software > ISaGRAF Ver.3 (English) - 093



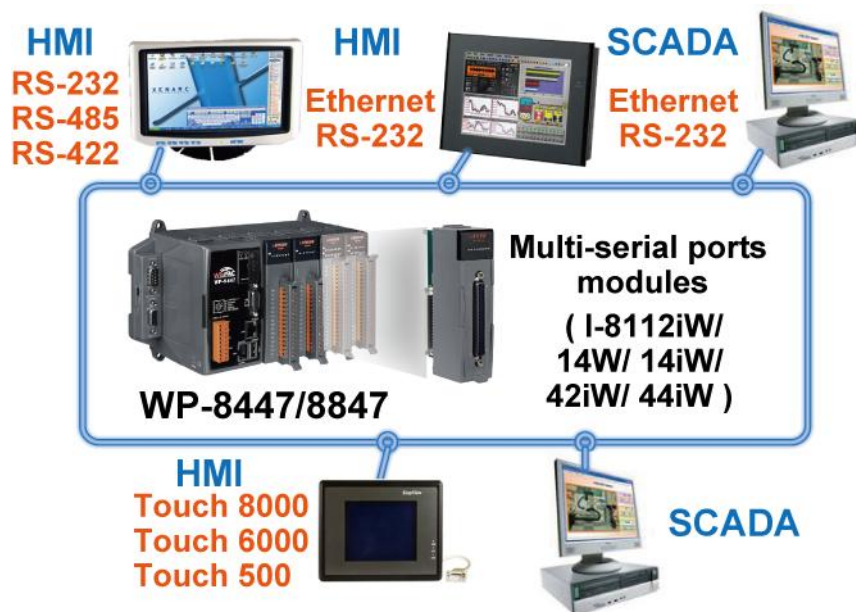
1.3: Modbus Master: RTU, ASCII, RS-232/485/422

- Support up to 10 ports: COM1~COM4 & COM5~COM14 (if I-8112iW/ 14W/ 14iW/ 42iW/ 44iW in Slot0~2)
- Can link to Modbus PLC or M-7000 I/O or Modbus devices (Power meter, temperature controller, inverter etc.)

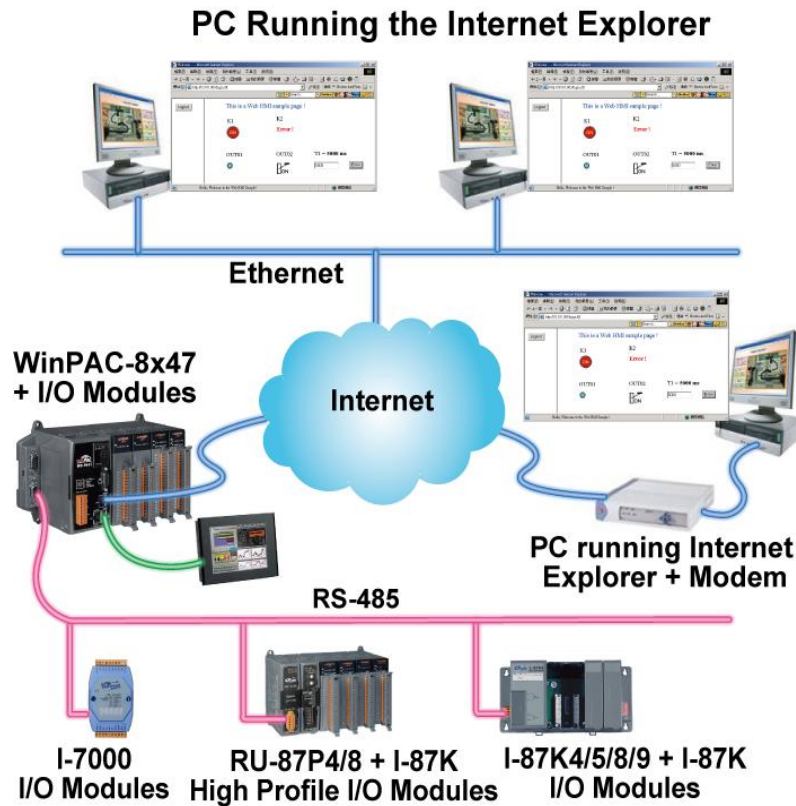


1.4: Modbus RTU/TCP Slave Ports

- Modbus RTU (RS-232/485/422): max. 5 ports
- Modbus TCP/IP: max. 32 connections



1.5: Multiple Web HMI – Monitor & Control Everywhere!

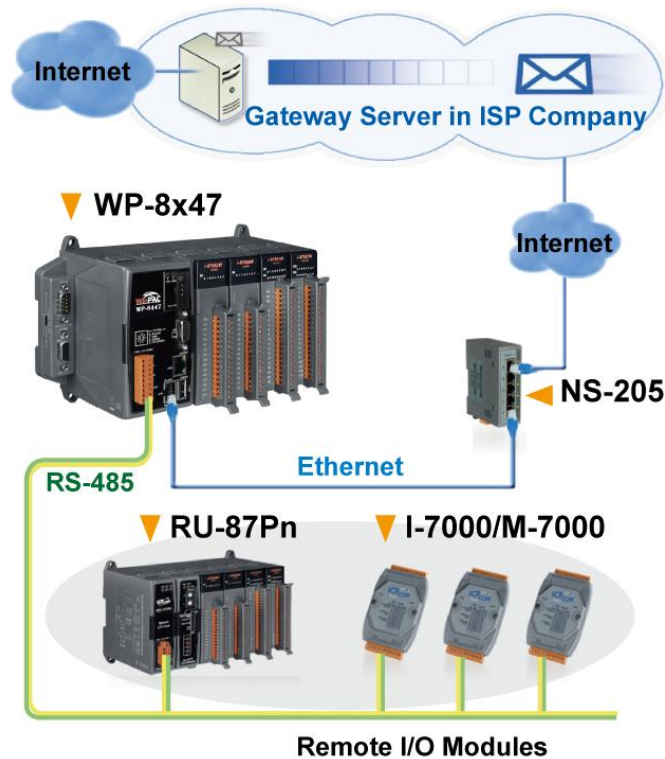


1.6: Communicate With Other TCP/IP Server or UDP Client/Server Devices

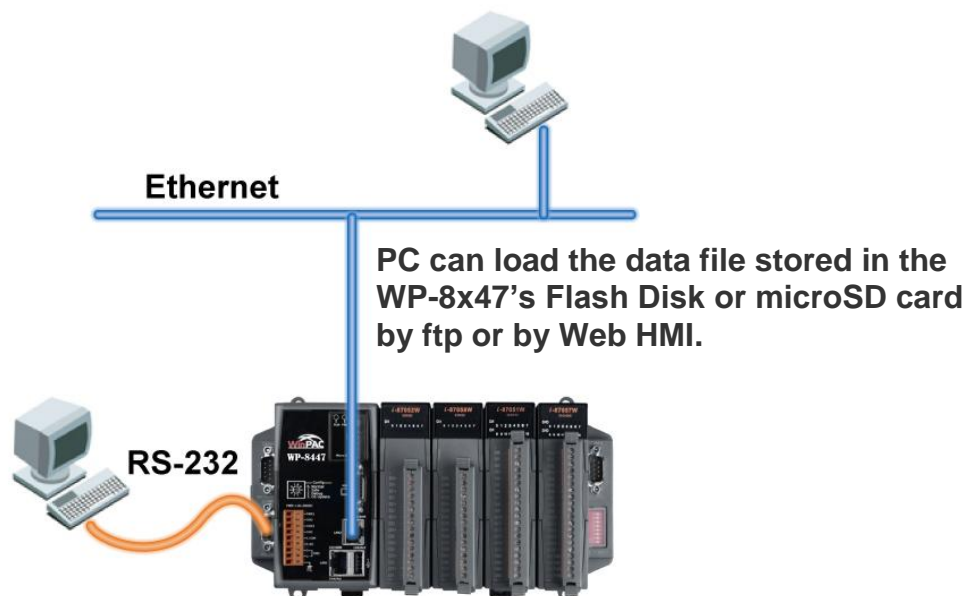


1.7: Send Email with One Attached File

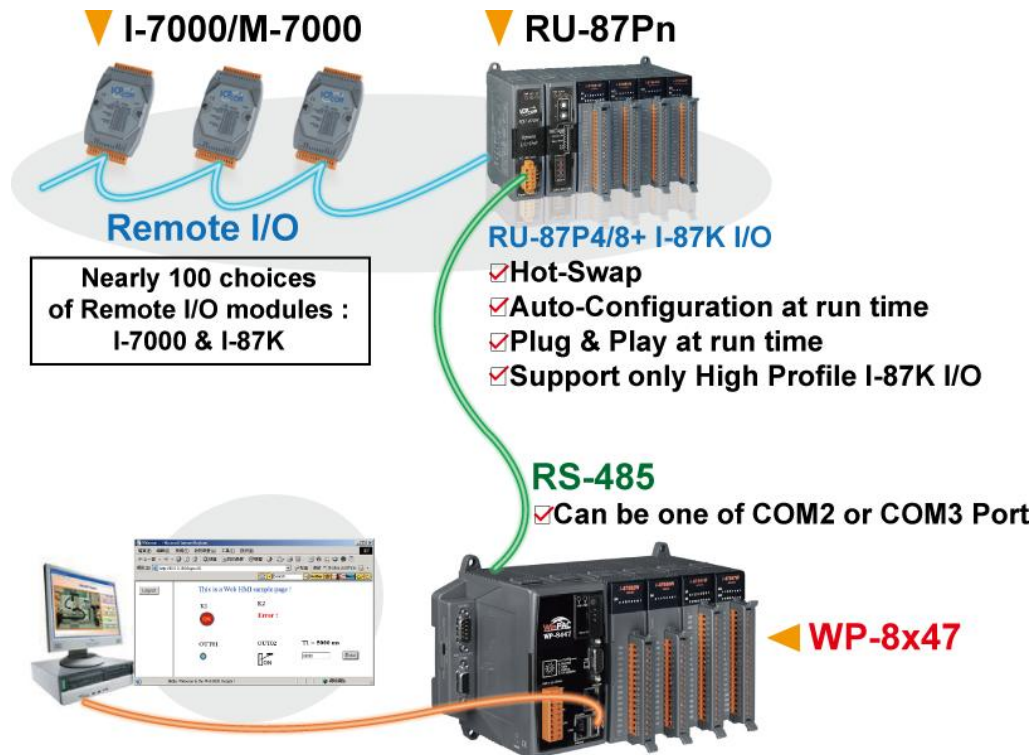
- More at [www.icpdas.com > FAQ > Software > ISaGRAF Ver.3 \(English\) - 067](http://www.icpdas.com > FAQ > Software > ISaGRAF Ver.3 (English) - 067)



1.8: Data-Recorder & Data-Logger



1.9: Remote I/O Application



1.10: SMS: Short Message Service

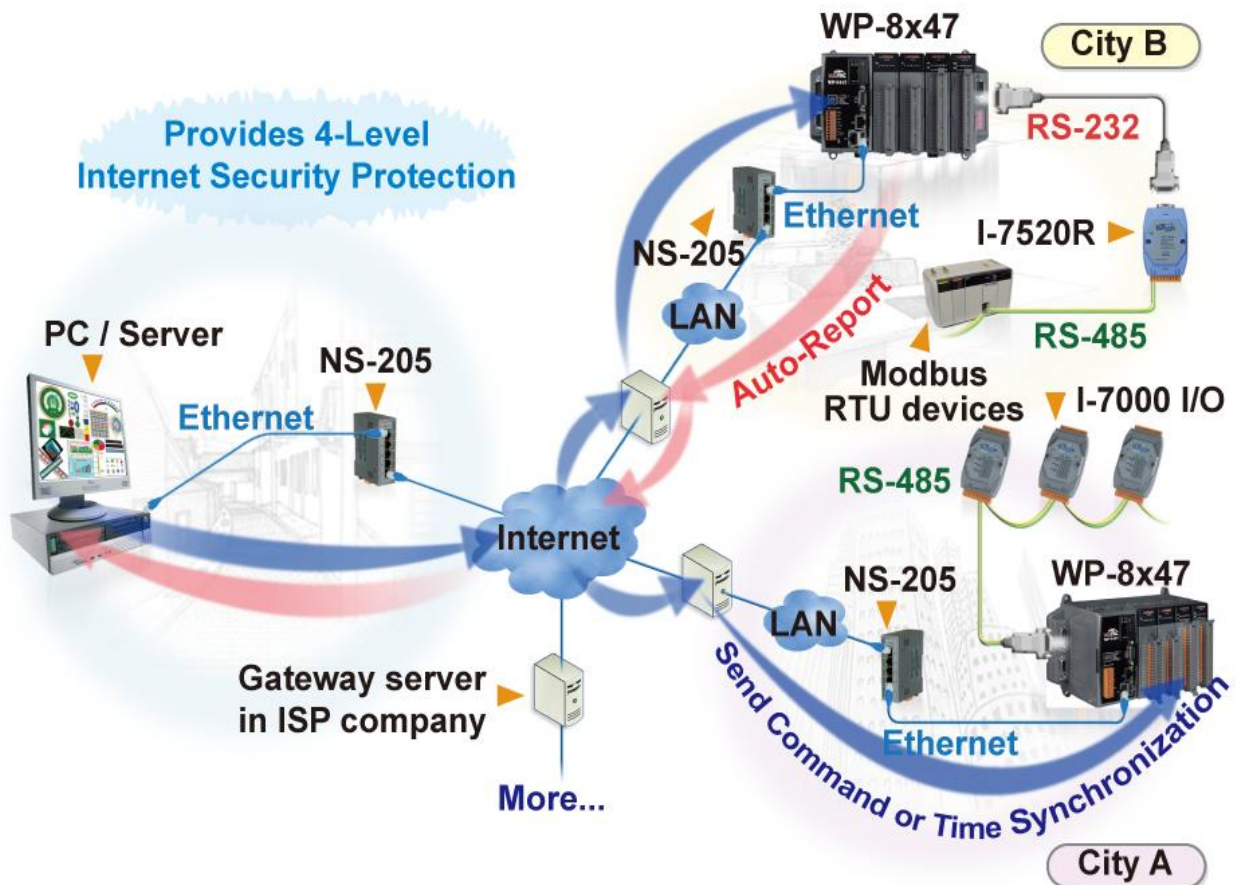


1.11: Auto-report Acquisition & Control Data

WP-8447/8847 can use UDP IP Client to auto-report acquisition data & control data to local or remote internet PC/Server.

Advantage : Every WinCon in the different location doesn't need a fixed Internet IP

Stable and Cost-effective Data Acquisition Auto-Report System



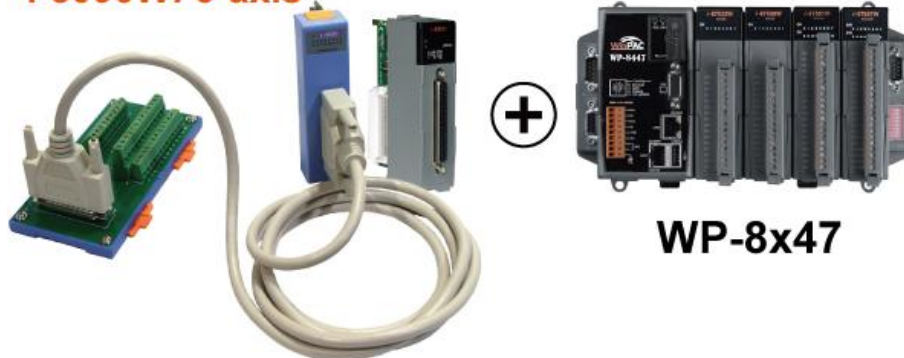
1.12: Motion Control

- One I-8091W can control 2 axes: X-Y plane, or 2 axes independent
- Two I-8091W can control 4 axes: X-Y plane + 2 axes independent, or 4 axes independent

Encoder Modules:

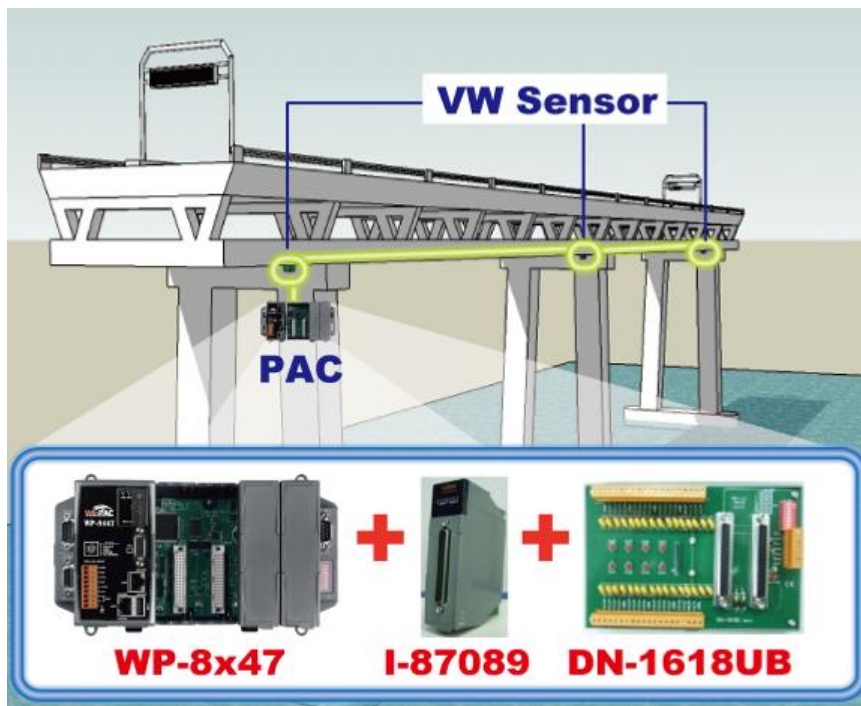
I-8080W: 4-axis, without Z-index

I-8090W: 3-axis



1.13: Stress Monitoring Application of Constructions

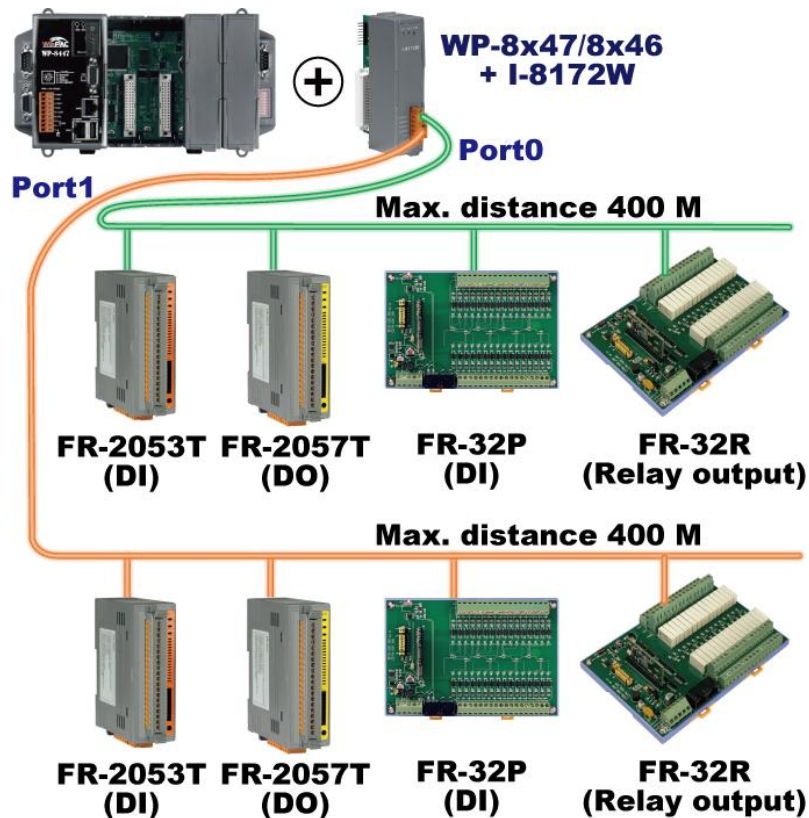
More at www.icpdas.com > FAQ > Software > ISaGRAF Ver.3 (English) - 091



1.14: Fast FRnet Remote I/O

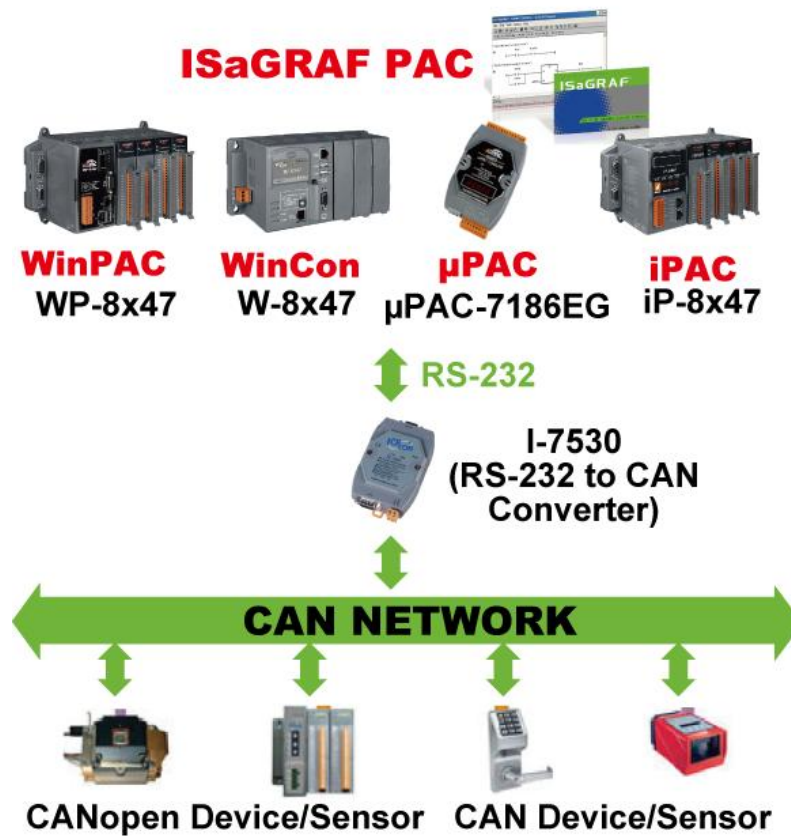
Advantage of FRnet I/O:

- Fast I/O scan: About 3 ms/scan.
(It depends on your program's PLC scan time. Ex: If the ISaGRAF program's PLC scan time is about 9 ms, then the scan time for all will be 9 ms, not 3 ms)



1.15: Integrate with CAN/CANopen Devices & Sensors

- WP-8X47 supports max. **10** I-7530 (RS-232 to CAN Converter)
- Please refer to www.icpdas.com > FAQ > Software > ISaGRAF Ver.3 (English) > 086



Chapter 2 : Software Installation

The WinPAC-8xx7 (or WP-8xx7) is the abbreviation of the WinPAC-8147 / 8447 / 8847.

Important:

1. **WP-8xx7 / 8xx6 supports only High profile I-8K and I-87K I/O cards in its slot 0 to 7. Refer to WP-8xx7 CD: \napdos\isagraf\wp-8xx7\english_manu\ wp-8xx7_datasheet.pdf**
2. Please always set a **fixed IP** address to the WinPAC-8xx7. (No DHCP)

Please refer to below location for detailed ISaGRAF English User's Manual.

WinPAC-8xx7 CD: \napdos\isagraf\wp-8xx7\english_manu\ "user_manual_i_8xx7.pdf"

NOTE:

The WinPAC-8xx7/8xx6 supports ISaGRAF programming method & provides Web HMI solution by default.

If user would like to program the WinPAC-8xx7 by using both ISaGRAF & (EVC++ 4.0 or VS.net 2008), it is also possible. Please refer to Chapter 6 or Chapter 7.

2.1: Step 1 - Installing The ISaGRAF Software

The user has to install two software before he can program the WinPAC-8xx7 controller system. They are

- A. **ISaGRAF Workbench &**
- B. **ICP DAS Utilities For ISaGRAF**

User has to purchase at least one pcs. of ISaGRAF (Ver. 3.4x or Ver. 3.5x ISaGRAF-256-E or ISaGRAF-256-C) to install on his PC to edit, download, monitor & debug the controller system. Item (B) is free and it is burned inside the CD-ROM which is delivered with the WinPAC-8xx7.

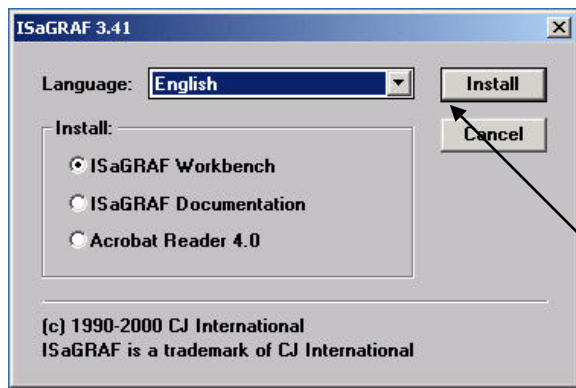
Operating system Requirements:

One of the following computer operating systems must be installed on the target computer system before you can install the ISaGRAF Workbench software program.

- Windows 98
- Windows NT Version 3.51 or Windows NT Version 4.0
- Windows 2000 Or Windows XP

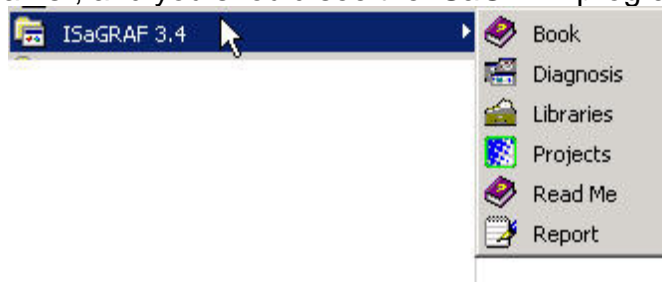
Steps To Installing The ISaGRAF Workbench:

Insert the ISaGRAF Workbench CD into your CD-ROM drive. If your computer does not have the auto-start feature active, use the Windows Explorer and go to the CD-ROM drive where the Workbench CD is installed, then double-click on the "install.bat" file listed on the ISaGRAF CD. If the "install.bat" file is not found on your ISaGRAF CD, then double-click on the "ISaGRAF.exe" file to start the installation process.

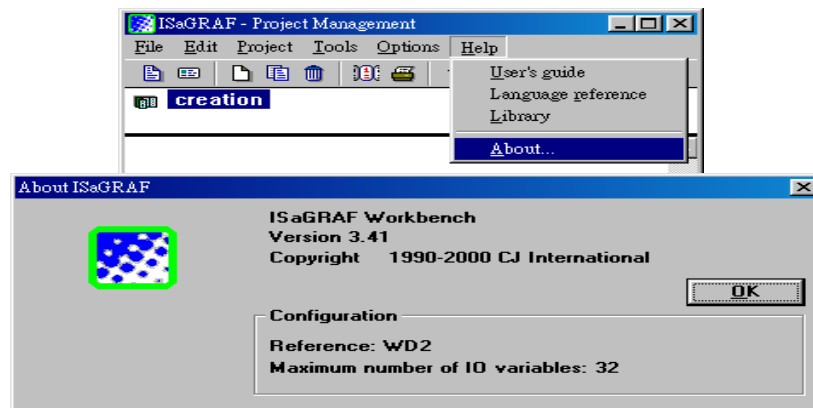


Select the language. Recommend to use "English" because this manual using English version.

To begin the ISaGRAF 3.x software program, click on the Windows "Start" button, then on "Programs", and you should see the ISaGRAF program group as illustrated below.



NOTE: You must install the hardware protection device (dongle) provided with the ISaGRAF software on your computers parallel port to for the ISaGRAF program to achieve fully authorized functionality.



While using ISaGRAF and the dongle is plugged well, if the "Help" – "About" says "Maximum number of IO variables: 32", it means ISaGRAF workbench cannot find the dongle well. Please reset your PC and then check the "Help" – "About" again. If it still displays "Maximum number of IO variables: 32", the dongle driver may not be installed well. Please execute the ISaGRAF CD_ROM \Sentinel5382\setup.exe for ISaGRAF-80 or \Sentinel\setup.exe for other ISaGRAF version and then reset the PC again.

If your ISaGRAF Key-Pro is USB type, please follow below steps to install the proper USB driver.

1. To make your PC recognize the ISaGRAF USB protection-key, please **un-plug** the USB protection-key from your USB port first, then run "**Sentine\SSD5411-32bit.exe**" in the ISaGRAF 3.55 CD-ROM (or later version) after you have installed the ISaGRAF. Then please reset your PC.
2. To run ISaGRAF Ver. 3.5x, please always plug the USB protection-key in the PC's USB port.

Important Notice For Window NT Users

If your computer is using the Windows NT operating system, you will need to add one line to the "isa.ini" file in the ISaGRAF Workbench "EXE" subdirectory.

C:\isawin\exe\isa.ini

You can use any ASCII based text editor (such as Notepad or UltraEdit32) to open the "isa.ini" file. Locate the [WS001] header in the "isa.ini" initialization file (it should be at the top of the file). Anywhere within the [WS001] header portion of the "isa.ini" initialization file, add the entry shown below within the [WS001] header:

[WS001]

NT=1

Isa=C:\ISAWIN

IsaExe=C:\ISAWIN\EXE

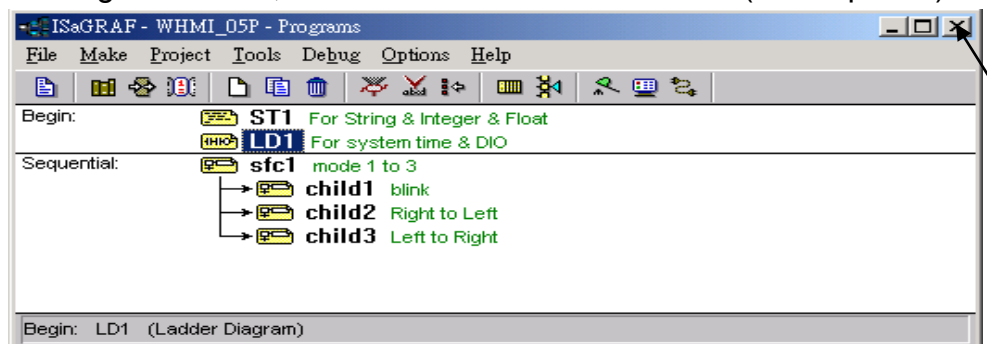
Group=Samples

IsaApl=c:\isawin\smg

IsaTmp=C:\ISAWIN\TMP

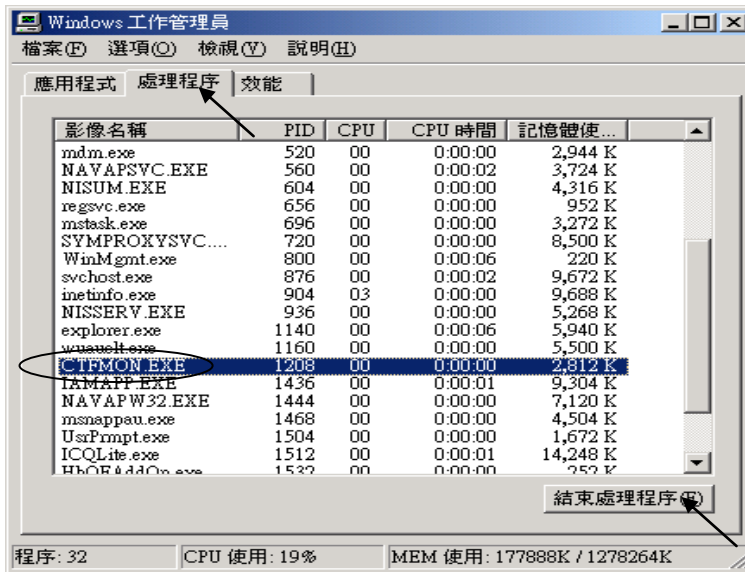
2.1.1: When closing my ISaGRAF window on windows 2000, it holds. Why ?

This problem usually happens on the windows 2000. When you close some ISaGRAF windows by clicking on the "X", it holds about 20 to 40 seconds (No response).



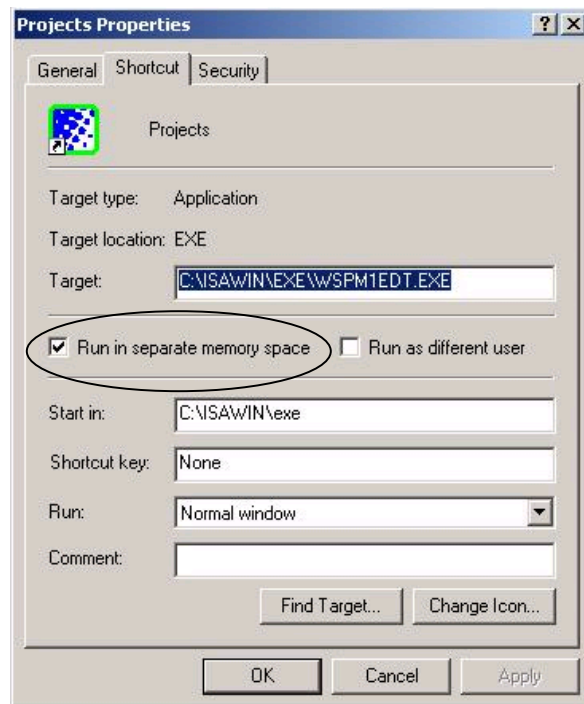
This "hold" behavior is caused by the "CTFMON.EXE" process. We still don't know the reason yet. You may stop this process by click on the "Ctrl" & "Alt" & "Del" at the same time to open the window Task Manager, and then stop it as next page.

However you will find the “CTFMON.EXE” still load to run when you reboot your PC or run Microsoft Office. So you need to stop it every time when your windows 2000 is rebooted. If you want to know more about the “CTFMON.EXE”, please visit www.microsoft.com & search “CTFMON.EXE”.



2.1.2: One Quick way to avoid the “hold” problem on windows 2000.

You may create a short cut for the “ISaGRAF project manager. And then check on "run in separate memory space" option in the shortcut property.



2.2: Step 2 - Installing The ICP DAS Utilities For ISaGRAF

The “ICP DAS Utilities For ISaGRAF” consists of 3 major items.

I/O libraries for all ICP DAS ISaGRAF controllers.

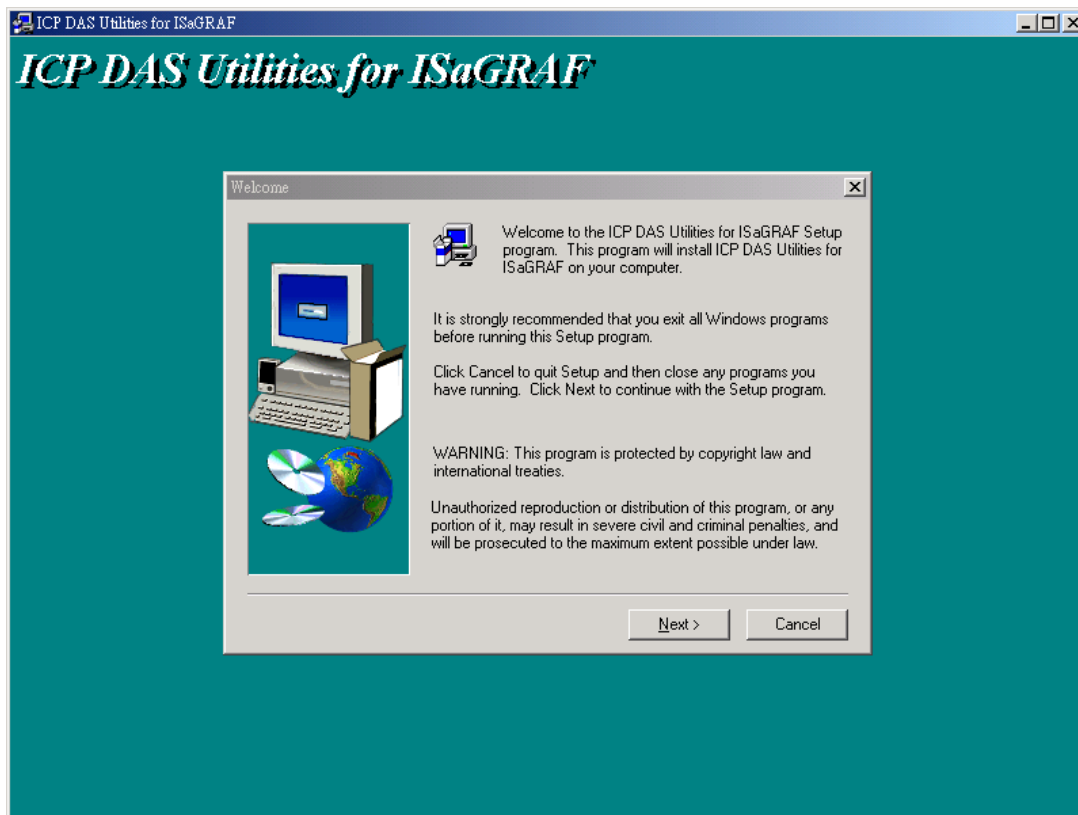
Modem_Link utility

Auto-scan I/O utility

Note:

The ISaGRAF Workbench software program must be installed before attempting to install the “ICP DAS Utilities for ISaGRAF”. If you have not already installed the ISaGRAF Workbench program, please refer to **step 1** before continuing.

There is a CD-ROM supplied with each of the WinPAC-8xx7 controllers with the “ICP DAS Utilities for ISaGRAF”. Please insert the CD-ROM into your CD-ROM drive. Then run **CD-ROM: \napdos\isagraf\setup.exe** . Follow the steps to install it.



Note:

If “ICP DAS Utilities for ISaGRAF” is not in your CD-ROM, please download “**ICP DAS Utilities For ISaGRAF.zip**” from <http://www.icpdas.com/products/PAC/i-8000/isagraf.htm> > Driver.

2.3: Step 3 - Installing The Web Page Editor

This is an option. You may not need it if you are very familiar with the HTML design. It is also possible to use any text editor to build web pages, for example, “Notepad” on the windows 2000 or XP.

We will use “Microsoft Office FrontPage 2003” (or higher version) to build web pages in this manual.

User may choose your prefer web page editor to do the same thing.

Chapter 3 : Setting Up A Web HMI Demo

The WinPAC-8xx7 (or WP-8xx7) is the abbreviation of the WinPAC-8147 / 8447 / 8847.

Important:

1. **WP-8xx7 / 8xx6 supports only High profile I-8K and I-87K I/O cards in its slot 0 to 7. Refer to WP-8xx7 CD: \napdos\isagraf\wp-8xx7\english_manu\ wp-8xx7_datasheet.pdf**
2. Please always set a **fixed IP** address to the WinPAC-8xx7. (No DHCP)
Recommend to use the NS-205 / NS-208 Industrial Ethernet Switch for WinPAC-8xx6/8xx7.
- 3 The leftmost I/O slot number of the WinPAC is 0.

3.1: Web Demo List

The Web page location:

WinPAC-8xx7 CD-ROM: \napdos\isagraf\wp-8xx7\wp_webhmi_demo\

The respective ISaGRAF project location:

WinPAC-8xx7 CD-ROM: \napdos\isagraf\wp-8xx7\demo\

Demo list:

Name	Description	IO board
sample	A Web HMI sample	No I/O board
example1	A simple example listed in Chapter 4	slot 0: I-87055W
wphmi_01	Display controller's date & time	No I/O board
wphmi_02	DI & DO demo	slot 0: I-87055W
wphmi_03	Read / Write Long, float & Timer value	No I/O board
wphmi_04	Read / Write controller's String	No I/O board
wphmi_05	Multi-Pages demo Page menu is on the Left	slot 0: I-87055W
wphmi_05a	Multi-Pages demo Page menu is on the Top	slot 0: I-87055W
wphmi_06	AIO demo, scaling is in ISaGRAF	slot 2: I-87024W slot 3: I-8017HW
wphmi_07	AIO demo, scaling is in PC	slot 2: I-87024W slot 3: I-8017HW
wphmi_08	download controller's file to PC	slot 0: I-87055W
wphmi_09	pop up an alarm window on PC	slot 0: I-87055W
wphmi_11	Trend curve.	slot 2: I-87024W slot 3: I-8017hW
wphmi_12	Record 1 to 8 Ch. i8017HW 's volt every 50ms and draw trend curve by M.S.Excel	slot 3: I-8017hW slot 2: I-8024W
wphmi_13	Record 1 to 4-Ch. i8017HW's voltage every 10ms and draw trend curve by M.S.Excel	slot 3: I-8017hW slot 2: I-8024W

3.2: Steps To Set Up A Web HMI Demo

3.2.1: Step 1 - Setup The Hardware

A. Please have one WP-8147/8447/8847 and then plug one I-87055W board in its slot 0.

If you don't have the I-87055W (8 IN & 8 OUT board), please follow the same steps as below however your Web HMI demo may be replaced to "wphmi_01" not "wphmi_05"

B. Prepare one VGA monitor, one USB mouse and one Ethernet cable and then connect them to the WinPAC-8xx7. (Keyboard is using the software keyboard on the bottom-right of the VGA screen)

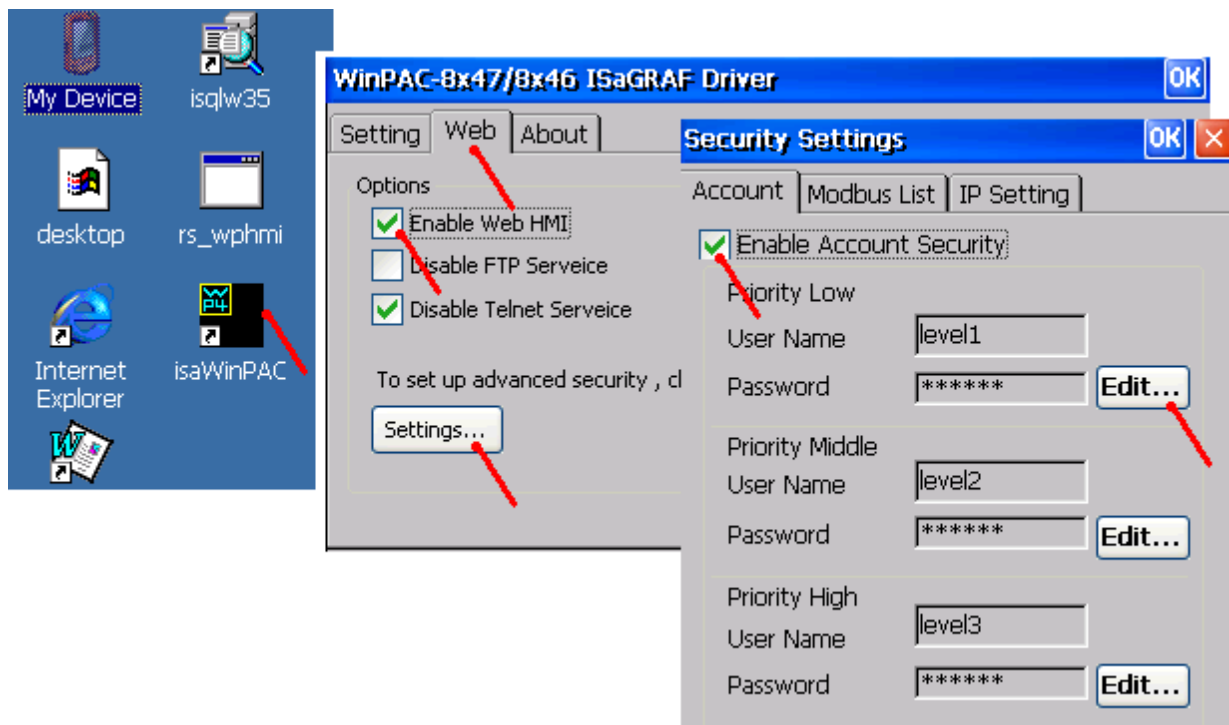
C. Power the WinPAC-8xx7 up.

3.2.2: Step 2 - Setting The Web Options

A. Please refer to the Appendix A.3 to set a **fixed IP** address to the WinPAC. (No DHCP)

B. Check on "Enable Web HMI" and then click on "Setting", Please check the "Enable Account Security" and then click on "Edit" to set (username , password). **Then remember to click on "OK"**

Note: If "Enable Account Security" is not checked, any user can easily get access to your WinPAC through the Internet Explorer.



3.2.3: Step 3 - Download ISaGRAF Project

Please download ISaGRAF project "wphmi_05" to the WinPAC-8XX7. This project is in the WP-8xx7 CD-ROM:\napdos\isagraf\wp-8xx7\demo\ "wphmi_05.pia"

wphmi_05 demo need one I-87055W. If you don't have the I-87055W (8 IN & 8 OUT board), you may download "wphmi_01" (CD-ROM:\napdos\isagraf\wp-8xx7\demo\ "wphmi_01.pia")

If you know how to restore "wphmi_05.pia" to your ISaGRAF Workbench and download it to the controller, please go ahead to the section 3.2.4.

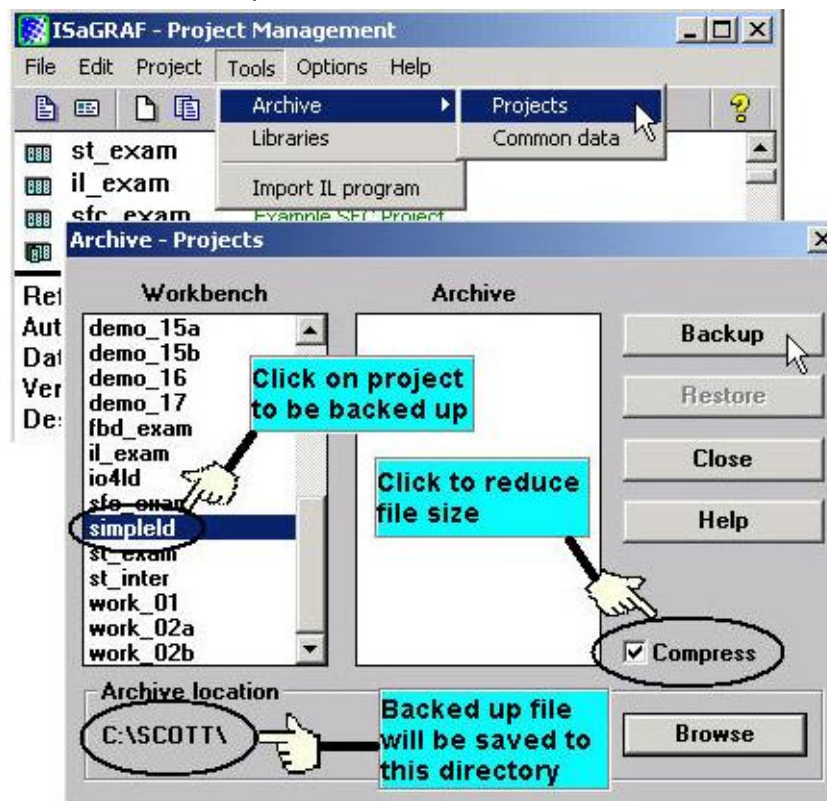
However if you don't know it, please refer to the below steps. Please make sure the ISaGRAF Workbench is already installed to your PC. (refer to the section 2.1 & 2.2)

Steps To Backing Up & Restoring An ISaGRAF Project:

For archiving purposes you can "Back Up" and "Restore" an ISaGRAF project. For example, you may want someone to test your program or email to service@icpdas.com for ICP DAS's ISaGRAF technical service.

Backing Up An ISaGRAF Project

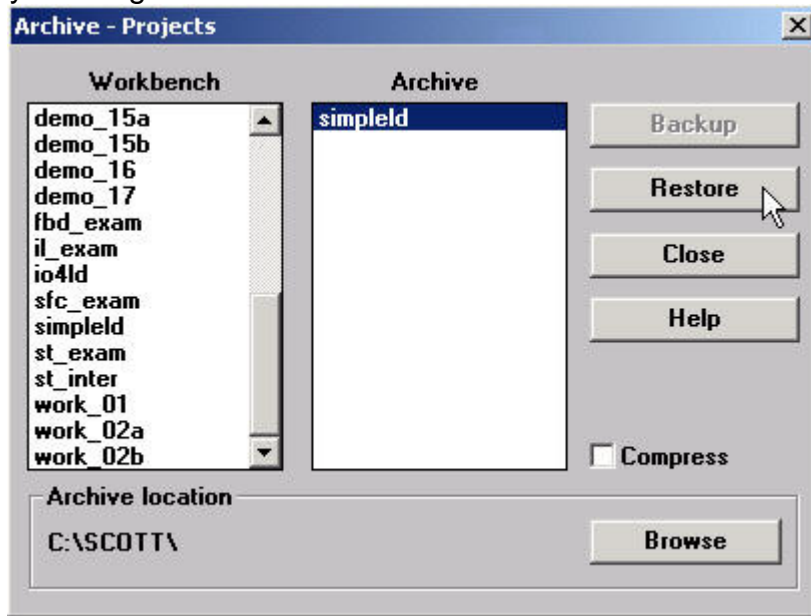
Open the "ISaGRAF Project Management", select "Tools" from the menu bar, click on "Archive", and then click on "Projects". An "Archive Projects" window will open which allows you to designate where you want to save the ISaGRAF project to. Click on the name of the ISaGRAF project you want to backup, and then click on the "Backup" button. You can compress the size of the file you have backed up by clicking on the "Compress" checkbox BEFORE you click on the "Backup" button.



Then you will now find the backed up ISaGRAF project file in the "Archive" location you have designated. In the example above, the name of the backed up file is "simpleld.pia".

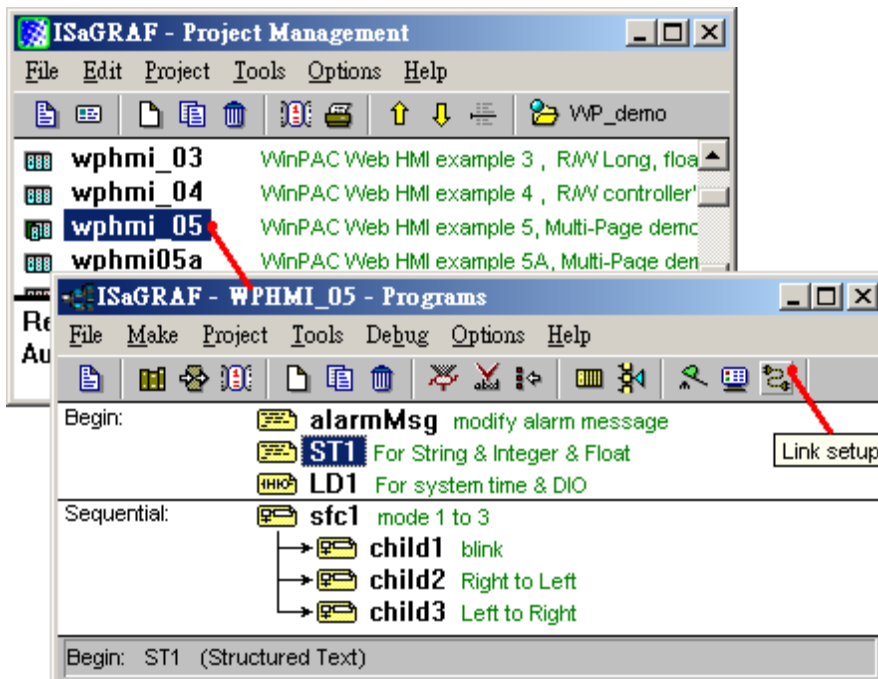
Restoring An ISaGRAF Project

To restore an ISaGRAF project from a backed up file(*.pia), use the same method as above to access the "Archive Projects" window, click on the name of the project you want to restore from the "Workbench" window, then click on the name of the backed up file from the "Archive" window, then click on the "Restore" button. The ISaGRAF project will now be restored to the sub-directory you designated.

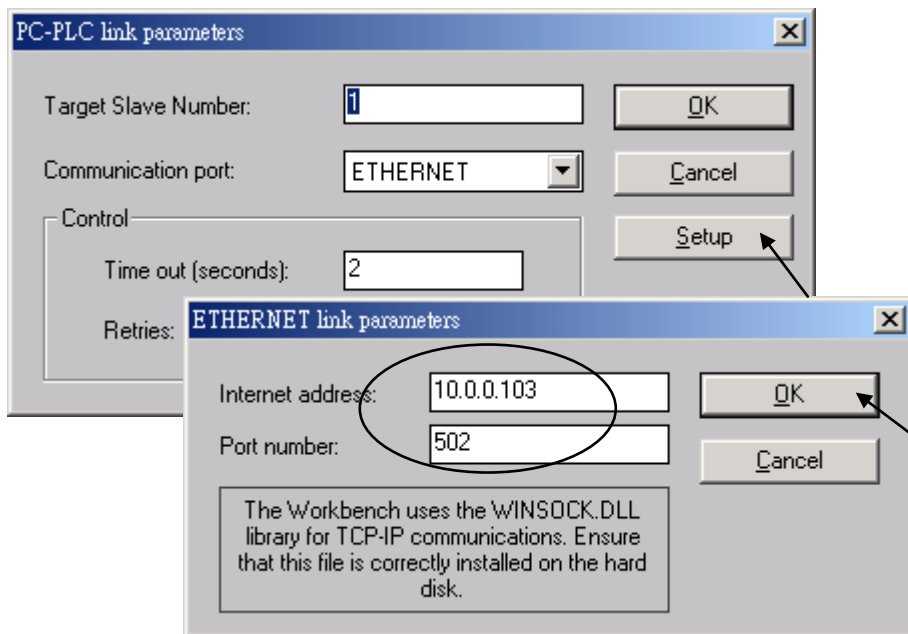


3.2.3.1: Steps To Download a ISaGRAF Project To The Controller:

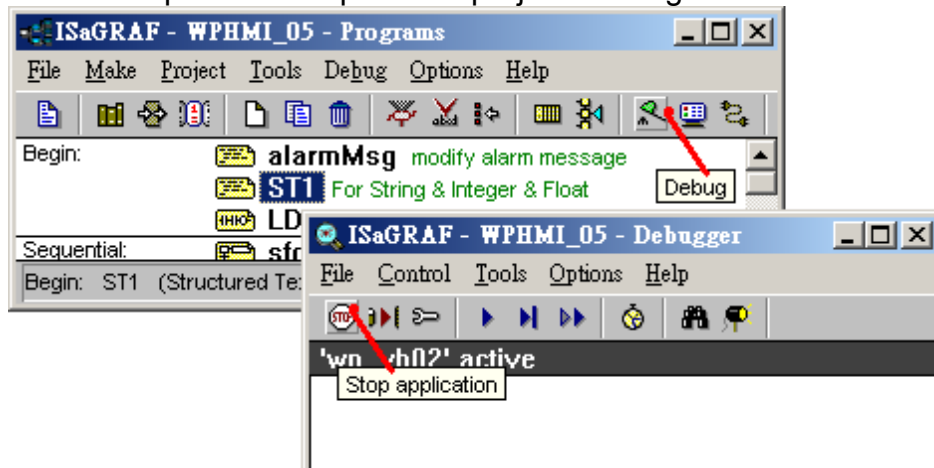
Double click on the "wphmi_05" to get into the project. Then click on "Link setup".



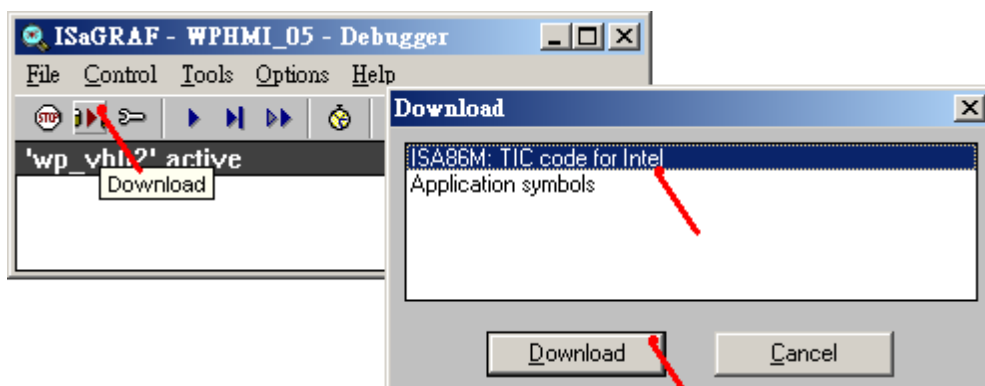
Click on “Setup” first and then entering the IP address of your controller. The port number should be 502.



To download “wphmi_05” project to the WinPAC-8xx7, Click on “Debug” . If communication is established, click on “stop” first to stop the old project running in the WP-8xx7.



Then click on “Download” to download it to the controller.



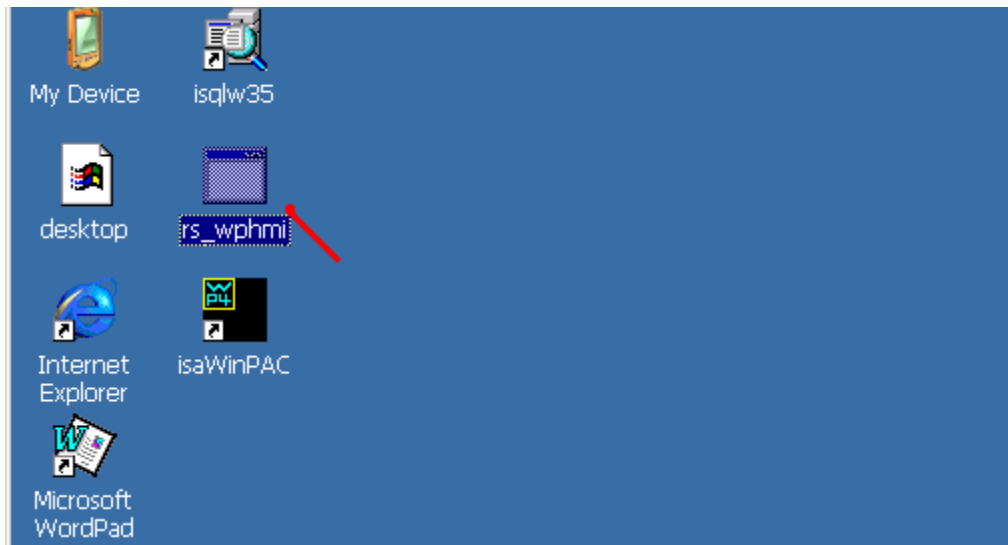
3.2.4: Step 4 - Download Web Pages To The Wincon

A. Please copy all files in the CD-ROM:

WinPAC-8xx7 CD: \napdos\isagraf\wp-8xx7\wp_webhmi_demo\wphmi_05\ *.*
to the WinPAC-8xx7 's \Micro_SD\Temp\HTTP\WebHMI\

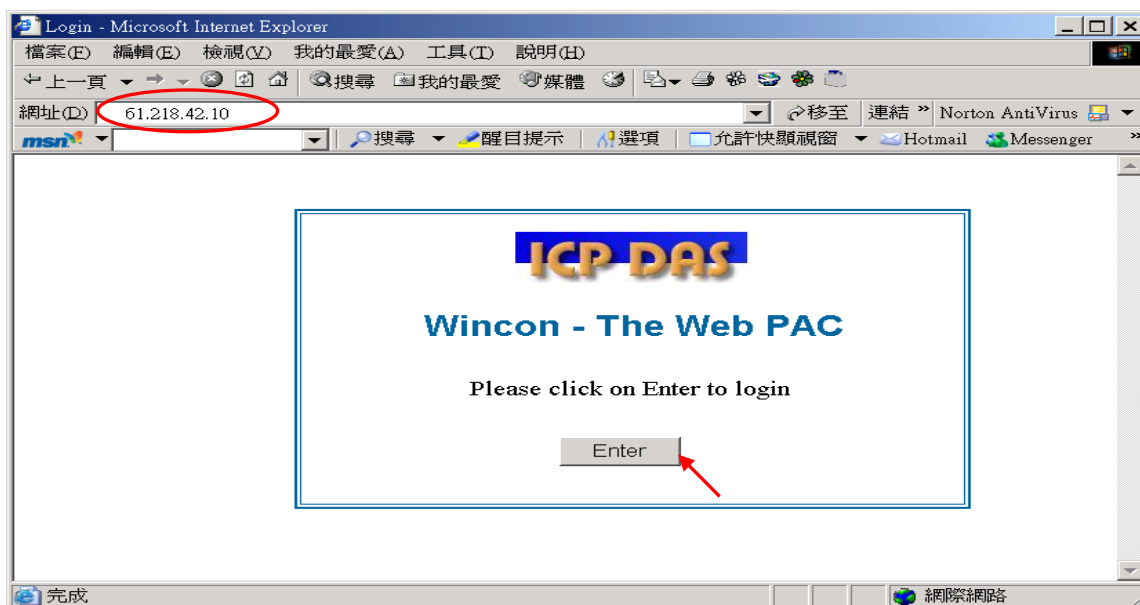
wphmi_05 demo need one I-87055W in its slot 0. If you don't have the I-87055W (8 IN & 8 OUT board), you may download "wphmi_01"

B. Since the Web Pages are modified or new copied, please run "rs_wphmi.exe" to reset the Web server. **The "rs_wphmi.exe" must be run every time when user has modified any file in the WP-8xx7 's \Micro_SD\Temp\HTTP\WebHMI**



3.2.5: Step 5 - Show Time

Please run Internet Explorer (Rev. 6.0 or higher), key in the IP address of your WinPAC-8xx7.
For example: 61.218.42.10 or http://61.218.42.10



Chapter 4 : Programming A Web HMI Example

The WinPAC-8xx7 (or WP-8xx7) is the abbreviation of the WinPAC-8147 / 8447 / 8847.

Important:

1. WP-8xx7 / 8xx6 supports only High profile I-8K and I-87K I/O cards in its slot 0 to 7. Refer to WP-8xx7 CD: \napdos\isagraf\wp-8xx7\english_manu\ wp-8xx7_datasheet.pdf
2. Please always set a **fixed IP** address to the WinPAC-8xx7. (No DHCP)
3. Recommend to use NS-205 or NS-208 Industrial Ethernet Switch for WinPAC.

This chapter show you how to build a simple ISaGRAF project and its Web HMI pages.

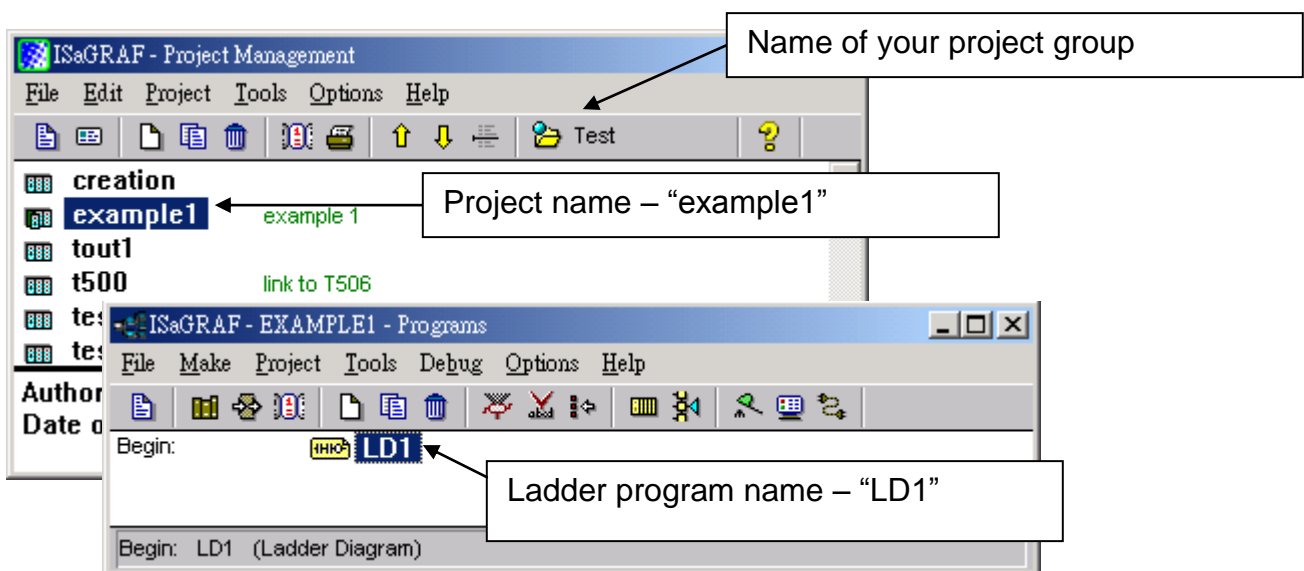
Please refer to CD-ROM: \napdos\isagraf\wp-8xx7\english_manu\ "user_manual_i_8xx7.pdf" - Section 2.1 for detailed ISaGRAF programming basics.

If user would like to program WinPAC-8xx7 by using both ISaGRAF & (EVC++ or VS.net), it is also possible. Please refer to Chapter 6 or Chapter 7.

4.1: Writing A Simple ISaGRAF Program

We are going to use ISaGRAF Workbench to write a simple ISaGRAF example program, then download it to the WinPAC-8xx7 controller (with one **I-87055W** I/O board in its slot 0) to make it work. If you haven't installed "ISaGRAF" & "ICP DAS Utilities for ISaGRAF", please go back to read chapter 2.

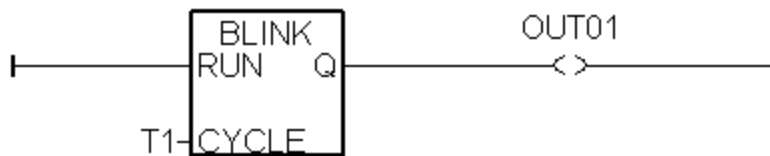
This example contains one Ladder program. (This demo program resides at the WinPAC-8xx7 CD-ROM: \napdos\isagraf\wp-8xx7\demo\ "example1.pia")



Variables declaration:

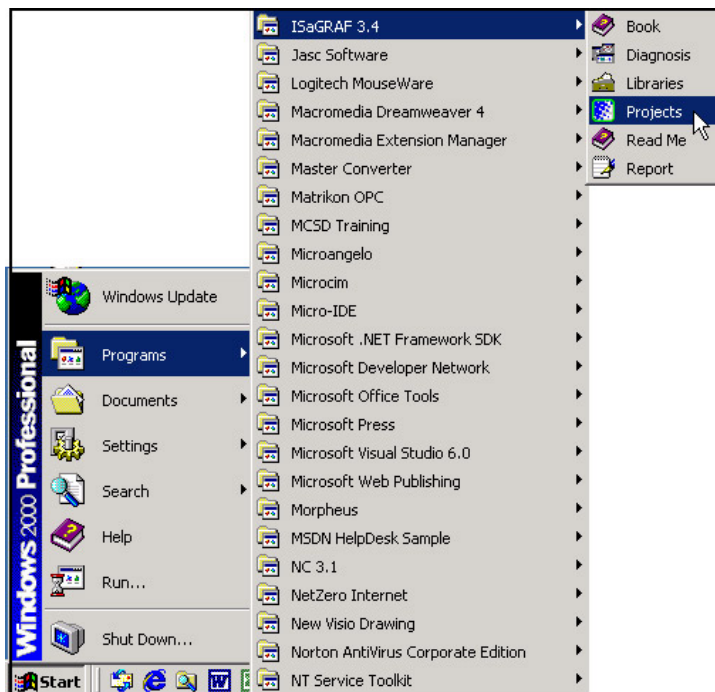
Name	Type	Attribute	Description
OUT01	Boolean	Output	Output 1 in the I-87055W, Modbus network addr = 1
OUT02	Boolean	Output	Output 2 in the I-87055W, Modbus network addr = 2
K1	Boolean	Input	Input 1 in the I-87055W, Modbus network addr = 11
K2	Boolean	Input	Input 2 in the I-87055W, Modbus network addr = 12
T1	Timer	Internal	Time Period of blinking, initial value set as T#8s Modbus network addr = 21

Ladder Logic Program Outline:



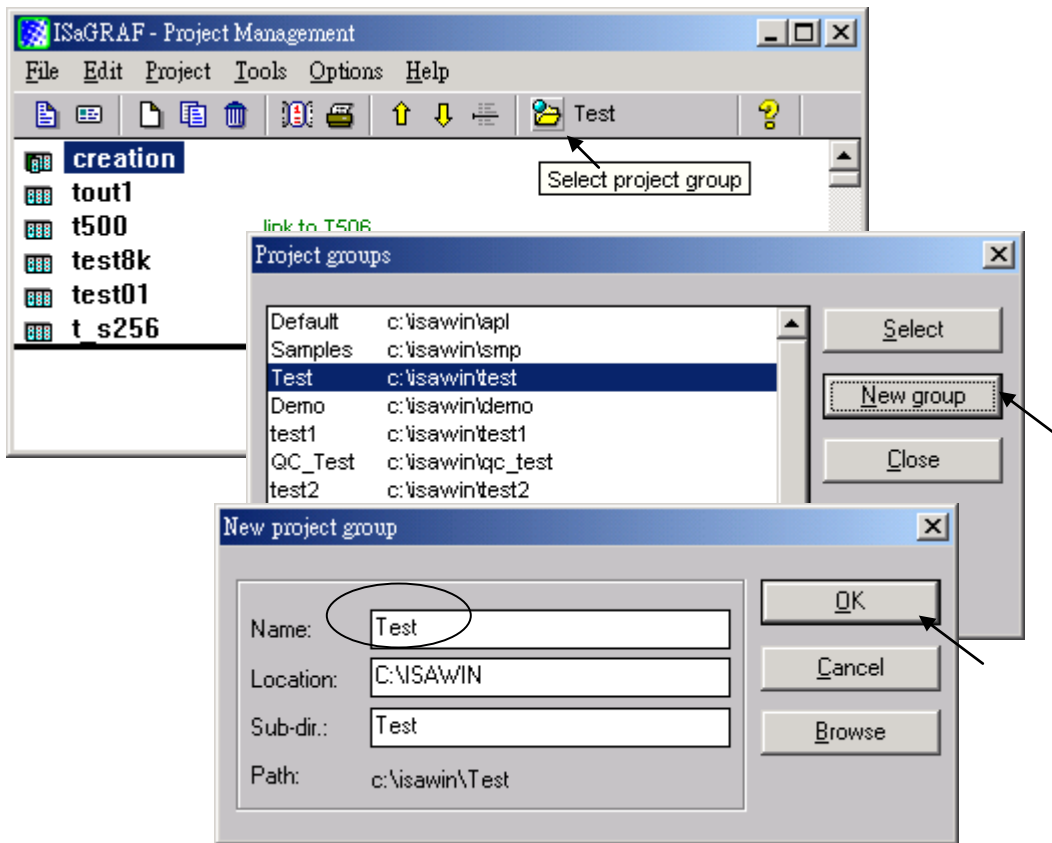
4.1.1: Open ISaGRAF-Project Management

Click on the Windows "Start" button, then click on "Programs" > "ISaGRAF 3.4", (or ISaGRAF 3.5) and then click on "Projects" as shown below.



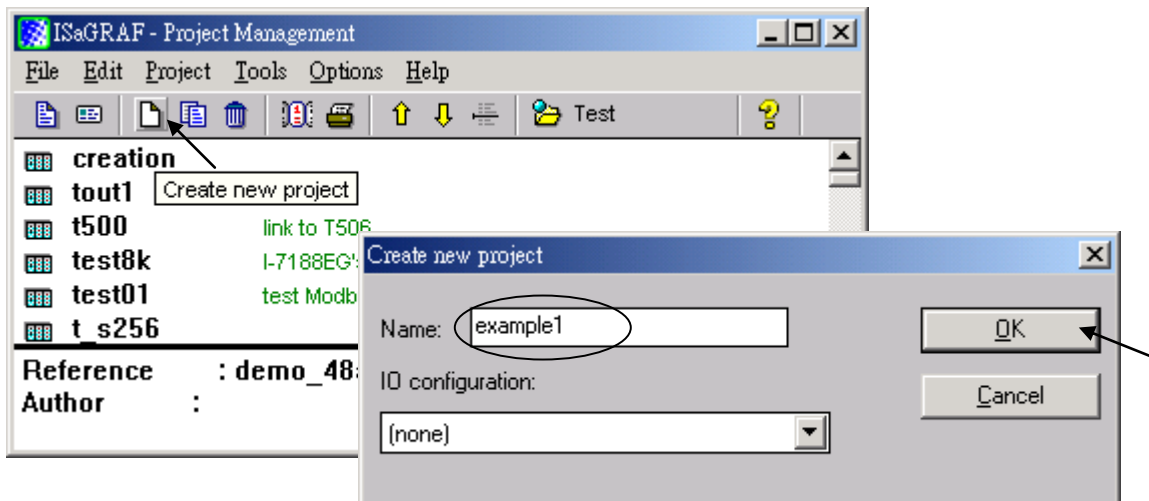
4.1.2: Creating An ISaGRAF User's Group

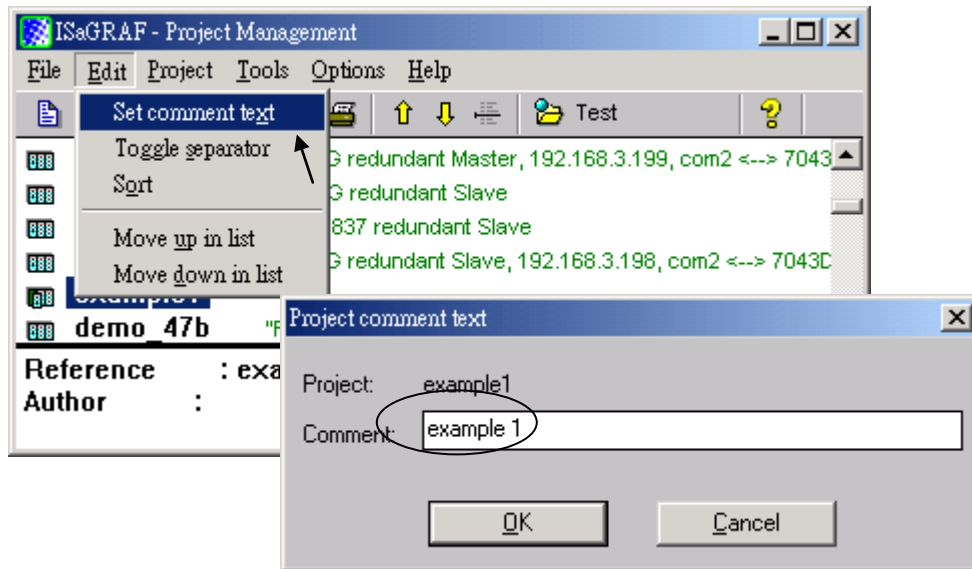
Click on the "Select Project Group", and then click on "New Group", then type in the name for the new user's group you wish to create, and last click on "OK".



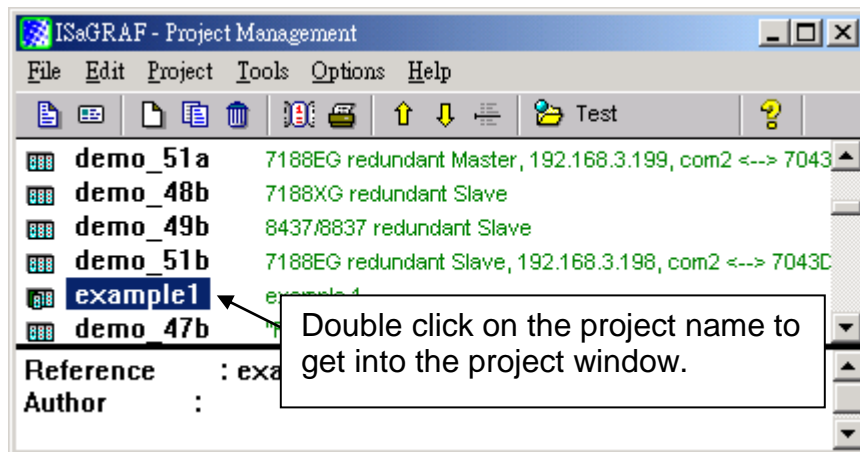
4.1.3: Creating A New ISaGRAF Project

To start a new ISaGRAF project, click on the "Create New Project" icon and then enter in the name for the new project. You can then enter additional information for your project by clicking on the "Edit" and then "Set Comment Text" menu as illustrated below.



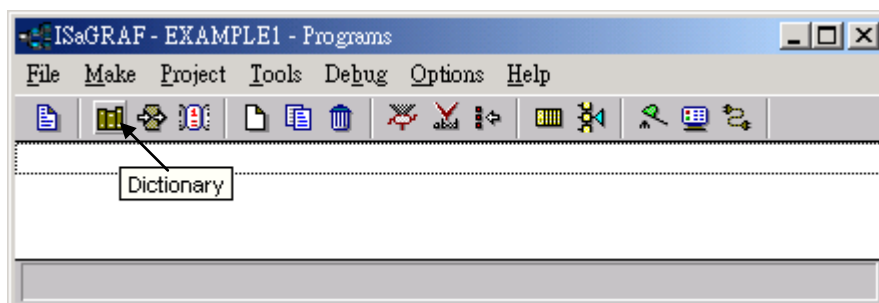


You will now see the name of the new project in the "Project Management" window. Double click on the name of the new project to open the new project.



4.1.4: Declaring The ISaGRAF Project Variables

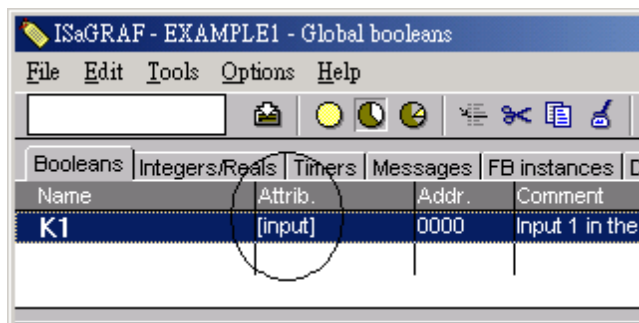
Before you can start creating an ISaGRAF program, you must first declare the variables that will be used in the ISaGRAF program. To begin this process, first click on the "Dictionary" icon and then click on the "Boolean" tab to declare the Boolean variables that will be used in our example program.



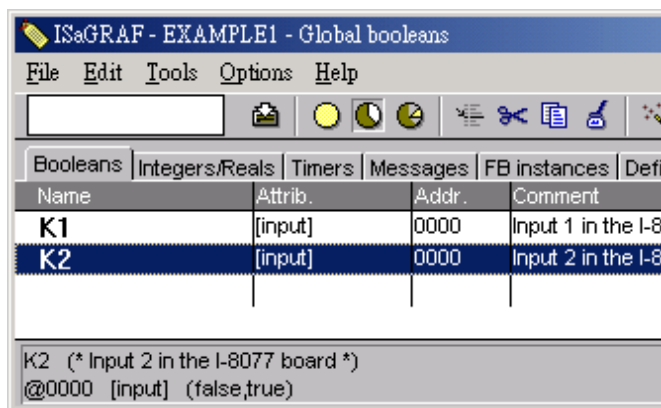
To declare the program variables for the ISaGRAF project, double click on the colored area below the "Boolean" tab, and a "Boolean Variable" window will open. Enter in the name of the

variable to be used in the project. For the purpose of this example program the variable "Boolean Variable Name" is "K1", and "Input 1 in the I-87055W board" is added to the "Comment Section". The next item that must be declared is what type of "Attribute" the variable will possess. In this example program, K1's attribute will be an "Input". Then press the "Store" button to save it.

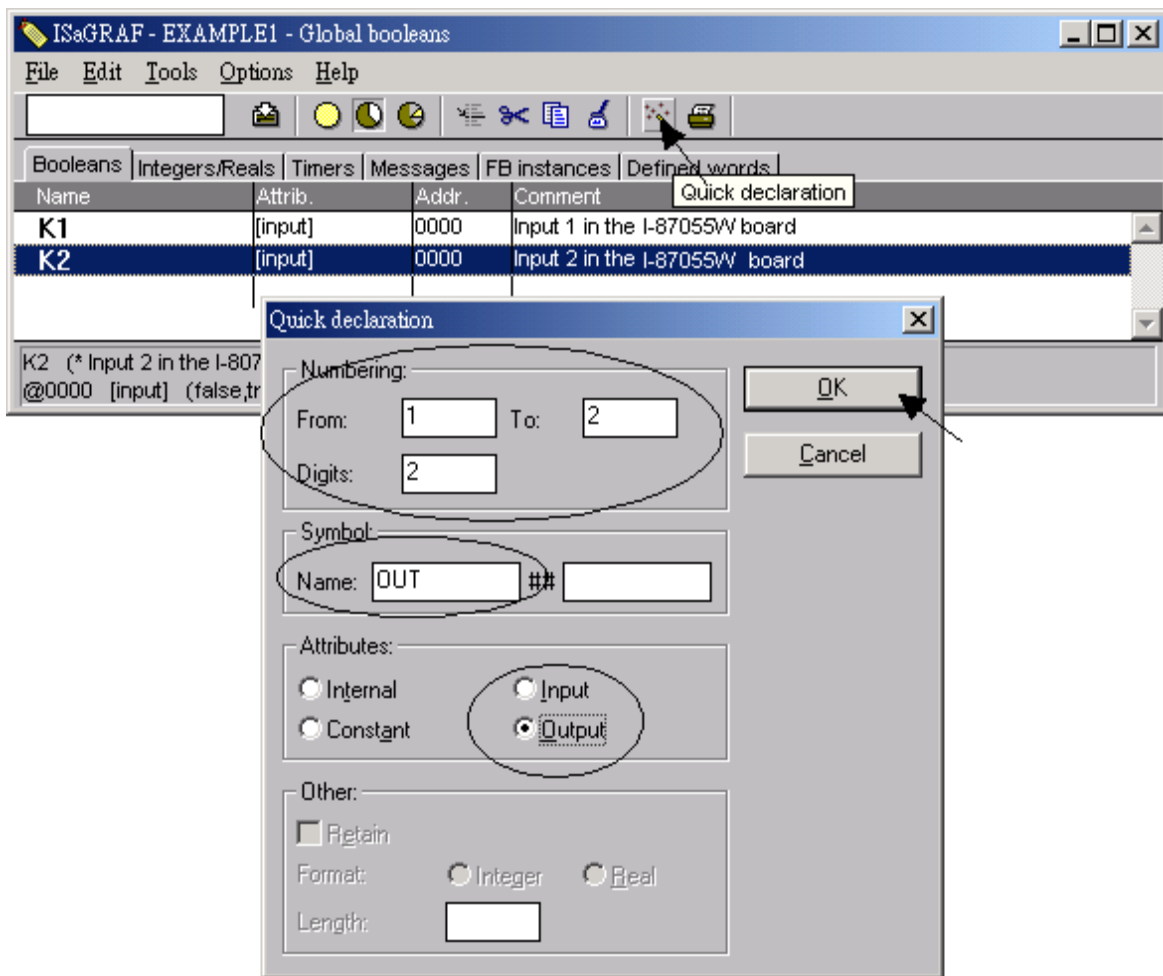
The new Boolean variable has now been declared.



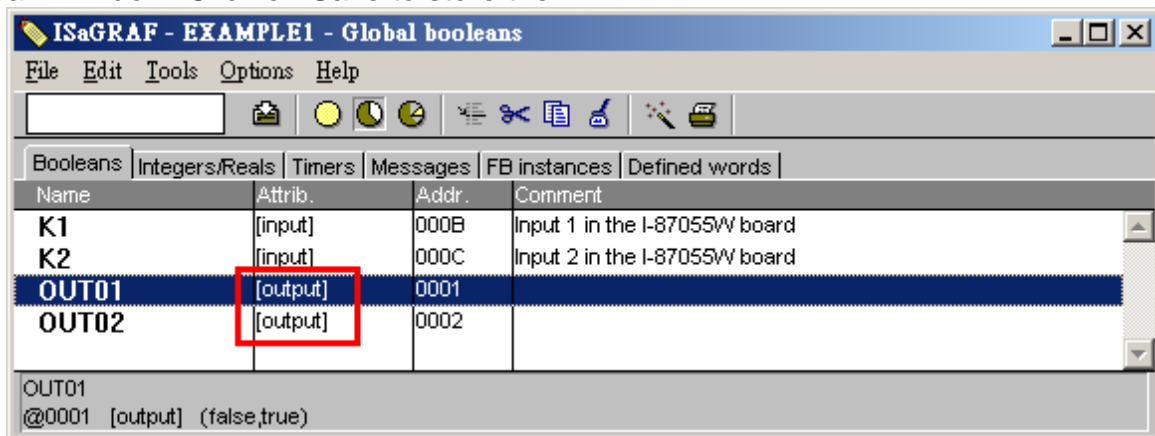
NOTE: You MUST make sure that the variable you have declared has the desired **Attribute** assigned. If you decide that you want to change a project variable's attribute, just double click on the variable name and you can reassign the attribute for the variable. Please follow the above same step to declare one another boolean variable – "K2". Then you will have as below.



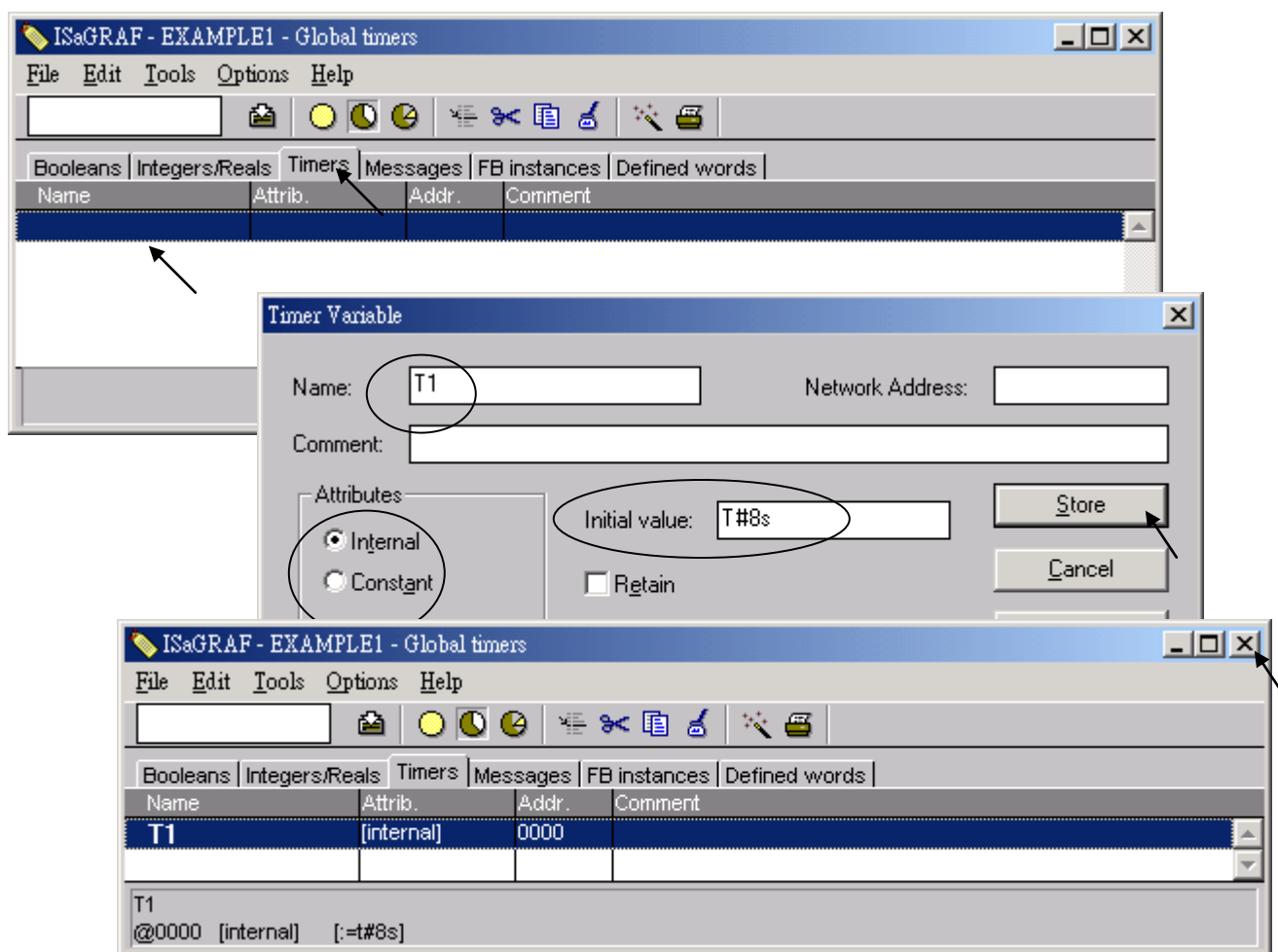
There are two outputs used in this example program named "OUT01 and OUT02". ISaGRAF provides a quick and easy way to declare like variables that are sequentially ordered. To begin this process, click on the "Quick Declaration" icon, and enter in the output number that you will start with the "Numbering" in "from" and "To" fields (this example uses from 1 to 2). Enter the "Symbol" name for the output variables being declared, and lastly, set the attribute to "Output"



When you click on the "OK" button, all two outputs will be immediately added to the "Global Boolean" window. Click on Save to store them.



To declare the timer (T1) variable used in this example program, click on the "Timers" tab in the setup screen. Double click on the colored area and enter the Name as "T1", set the "Attributes" to "Internal", the "Initial Value" to "T#8s", then click on the "Store" button. Then please click on "X" to close the "dictionary" window.



4.1.5: Assign Modbus Network Address No to Variables

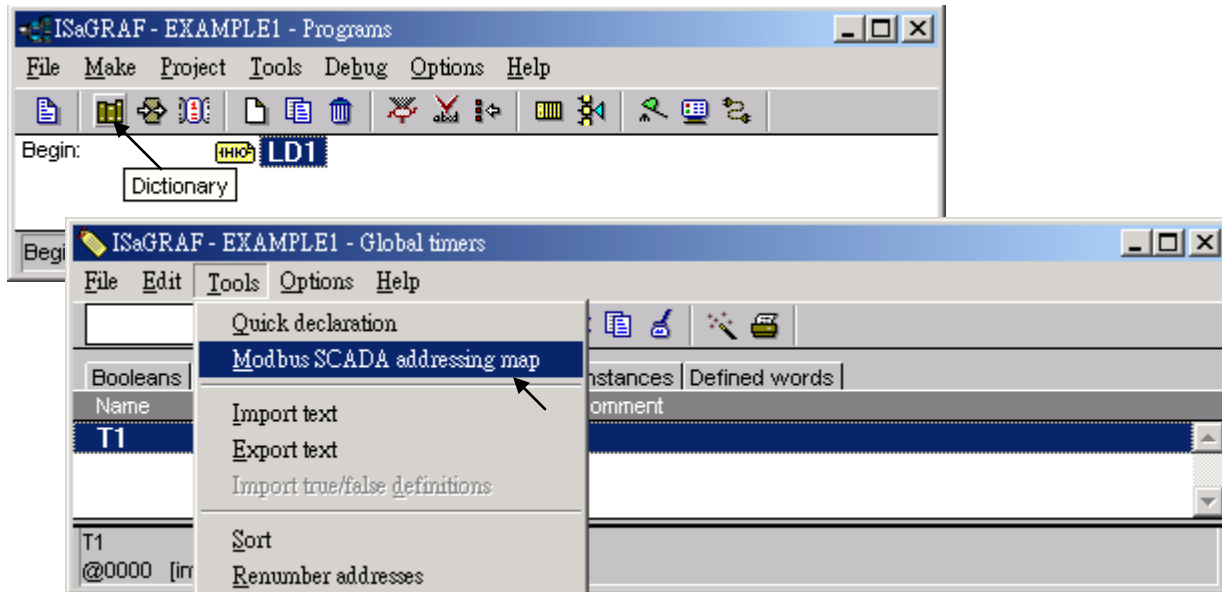
The Web HMI will exchange the variable value with the ISaGRAF project if they have assigned the proper "Modbus network address". The Web HMI only recognize Modbus No. from 1 to 1024. However other SCADA software may R/W the Modbus No. from 1 to 8191 in the WinPAC-8xx7.

Variables without assigning Modbus No. will not be available by Web HMI and other SCADA software or HMI devices.

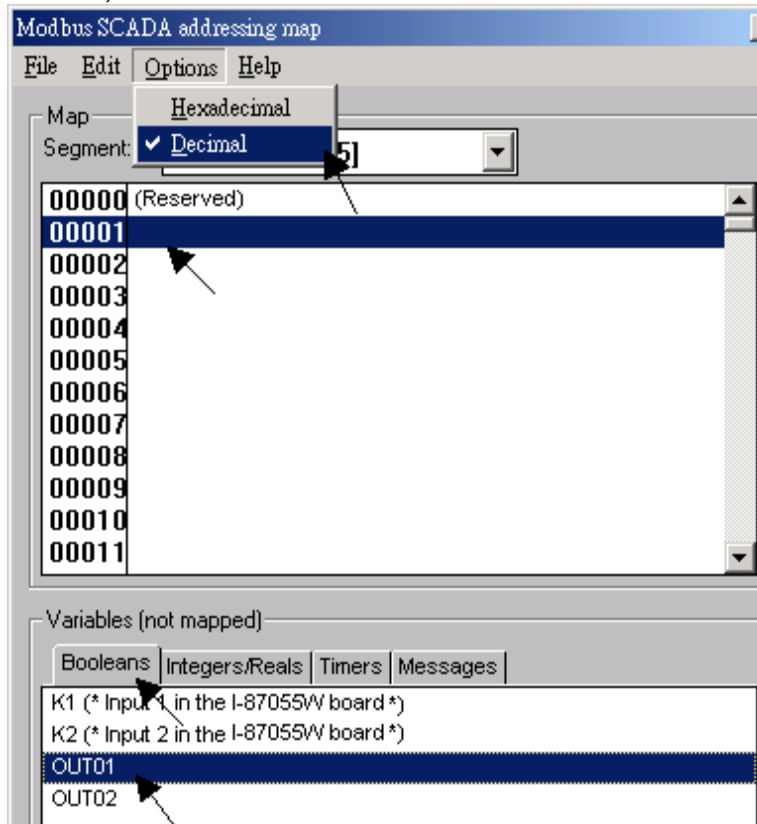
Please refer to WinPAC-8xx7 CD-ROM:

\napdos\isagraf\wp-8xx7\english_manu\ "user_manual_i_8xx7.pdf"
for section 4.1 & 4.2 for detailed information about assigning Modbus network address.

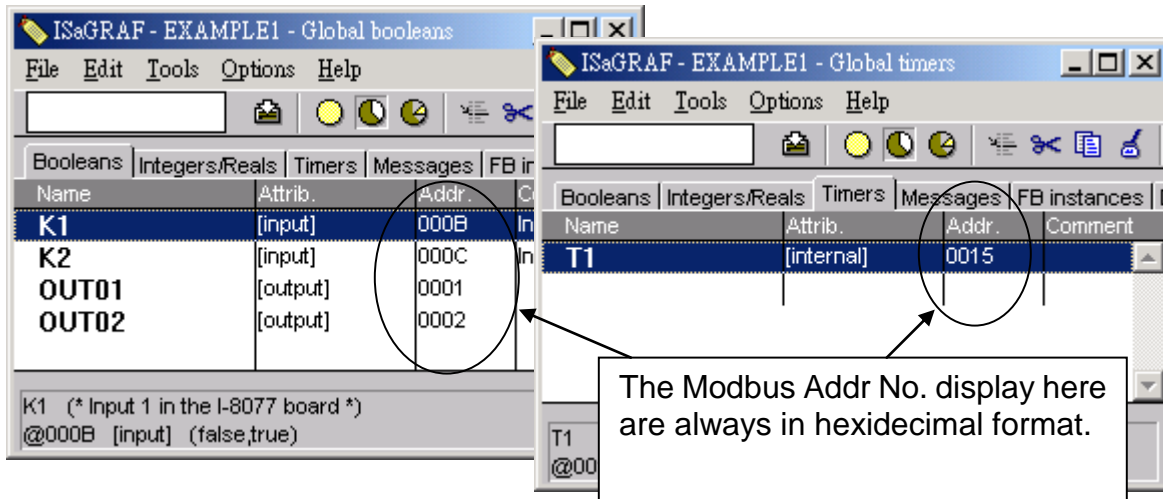
Please get into the dictionary, then click on “Tools – Modbus SCADA addressing map”



Please click on “Options – Decimal”, or it will use Hexadecimal format as default. First click on “00001” on the top window, and then double click on “OUT01” to attach it to the Modbus No. 1.



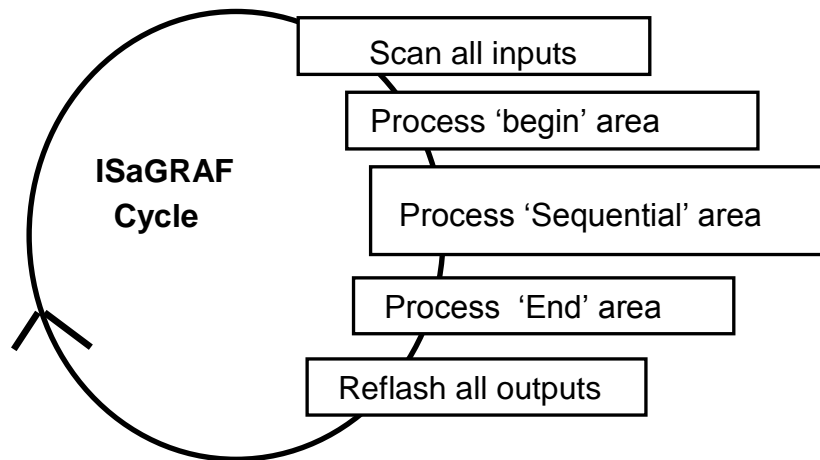
Please follow the same way to assign OUT01 to No.2, K1 to No.11, K2 to No.12 and then Timer variable T1 to No.21. Then we have below window.



Very Important: If assign Modbus No. to Long integer or Float or Timer variables, they should occupy two Modbus No. Please refer to WinPAC-8xx7 CD-ROM:
 \napdos\isagraf\wp-8xx7\english_manu\ "user_manual_i_8xx7.pdf" - Section 4.2 for detailed information.

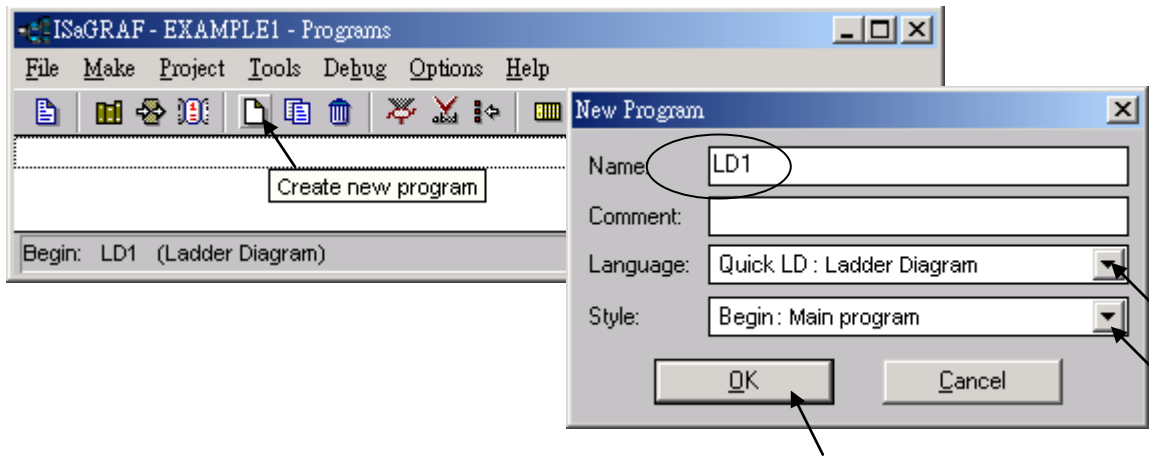
4.1.6: Create The LD - "LD1" Program

ISaGRAF will run every program one time in each PLC scan cycle. Programs in the "begin" area will run first, then the "Sequential" area, and last the "End" area. An ISaGRAF cycle runs in the way as the below scheme.

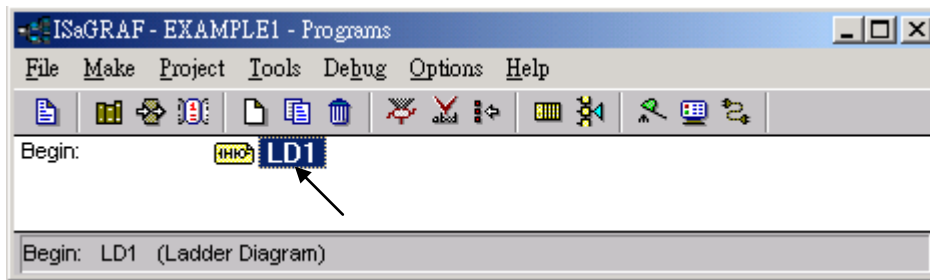


Click on the "Create New Program" icon and the "New Program" window will appear.

Enter the "Name" as "LD1", next, click on the "Language" scroll button and select "Quick LD: Ladder Diagram", and make sure the "Style" is set to "Begin: Main Program". You can add any desired text to the "Comment" section for the LD program, but it isn't required.

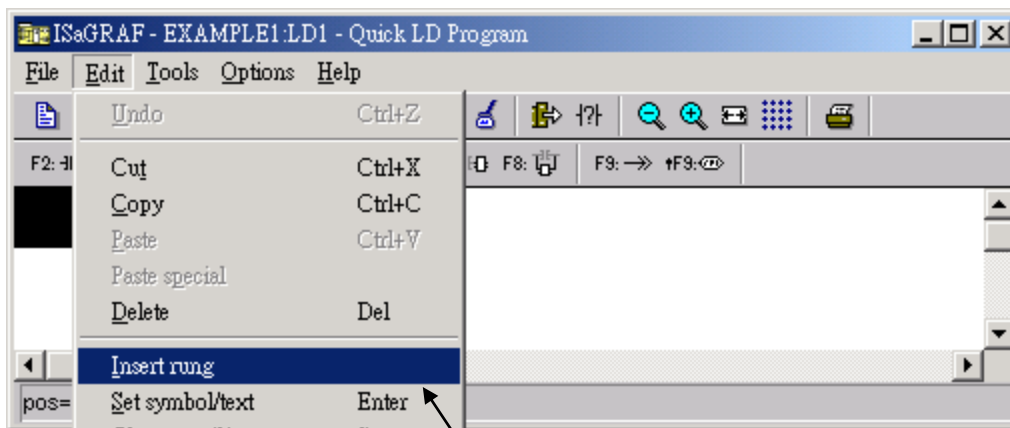


Now we have one program inside this project. Please double click on the "LD1" to get into it.

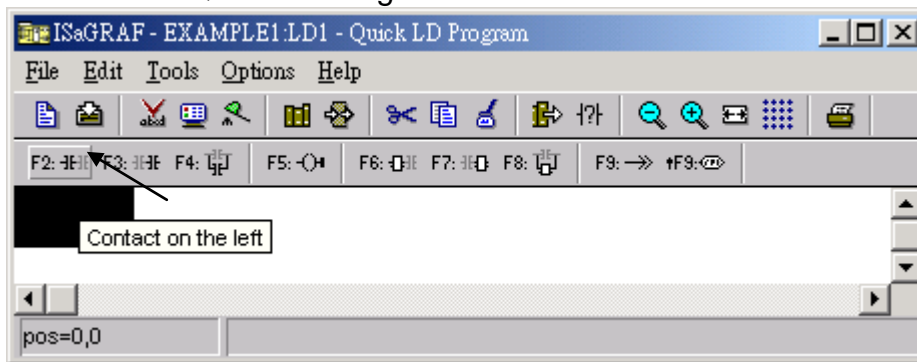


4.1.7: Edit The "LD1" Program

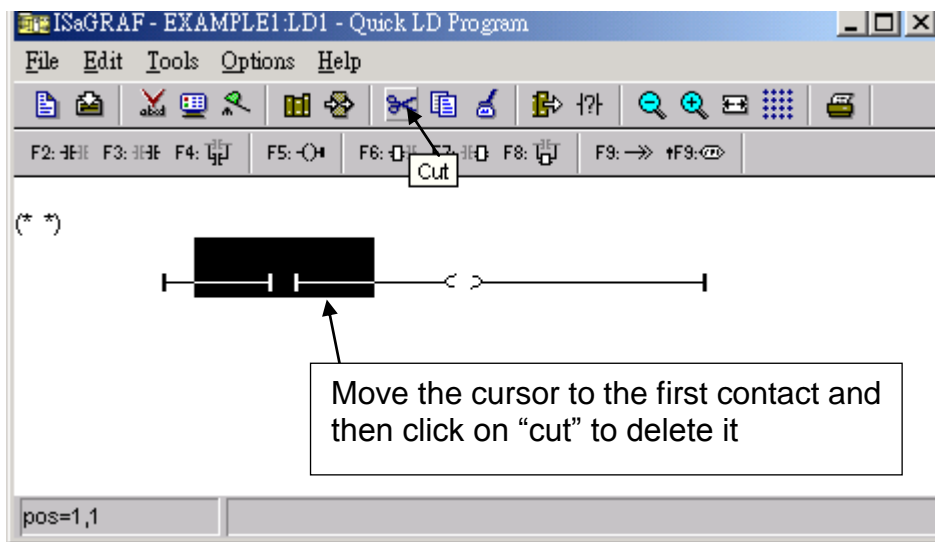
When you double click on the "LD1" name the "Quick LD Program" window will appear. To start programming our LD program, click on "Edit" from the main menu bar, then click on "Insert Rung" . "Insert Rung" means to insert a basic LD rung just above the current position.



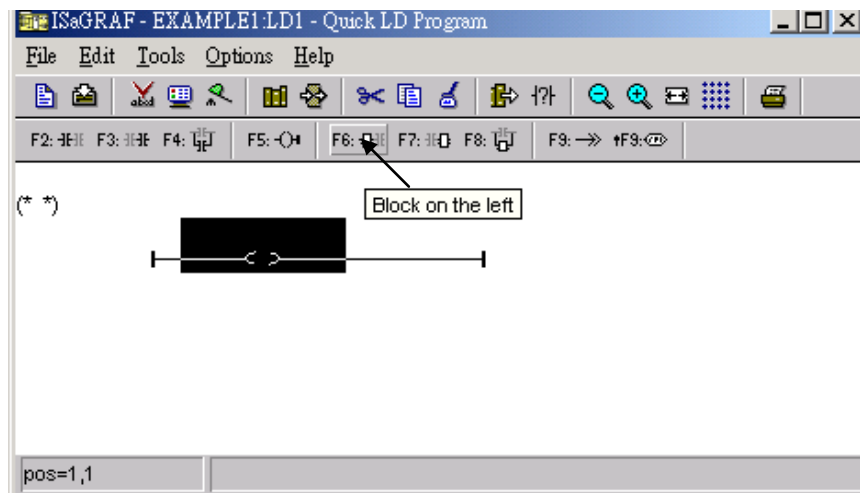
Or, you may just simply click on the "F2 (Contact On The Left)" icon, and the following will appear within the Quick LD Program window.



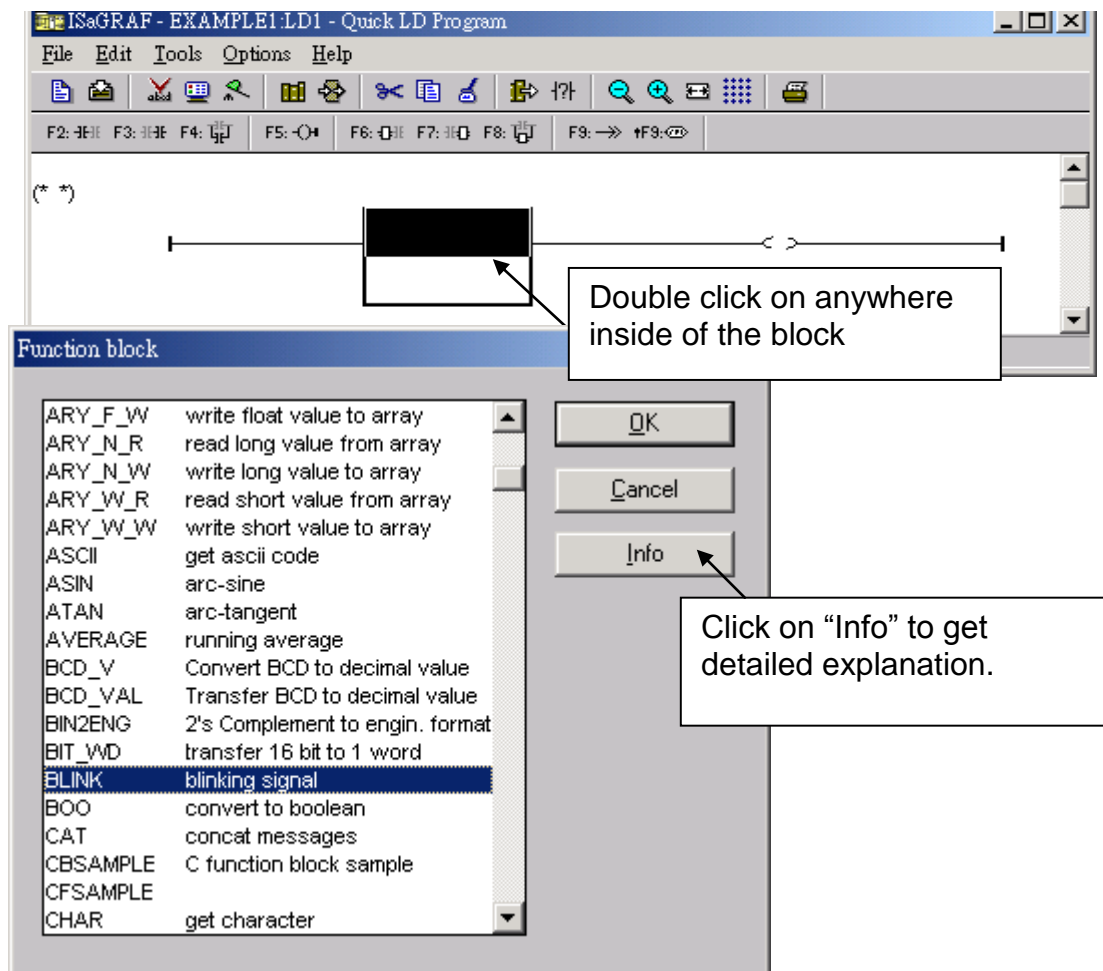
We are going to write the first line of the LD1 program. Move the cursor to the first "contact" and then click on "cut" to delete it.



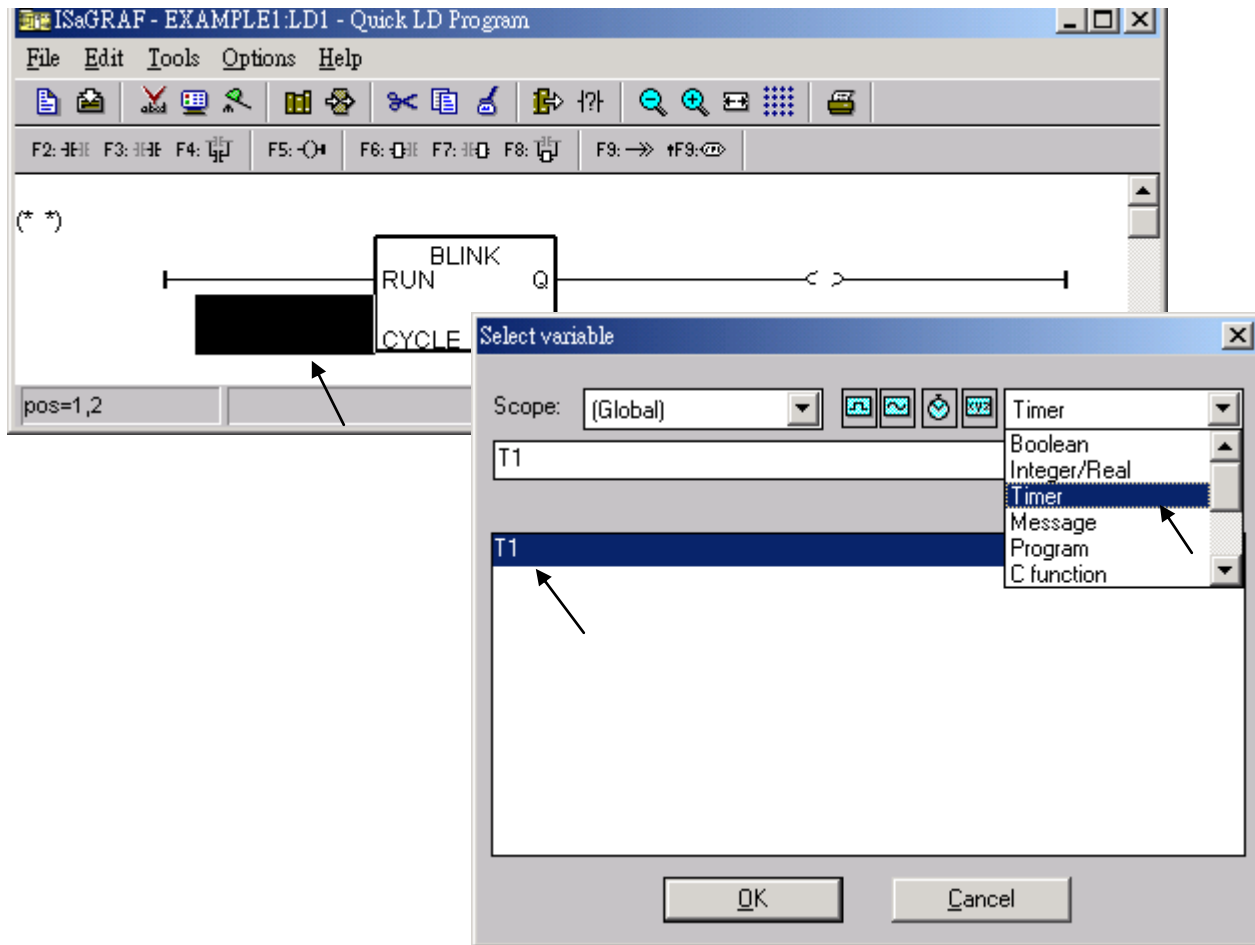
Click on the "F6 (Block on the left)" icon and you will create a block on the left of the "coil".



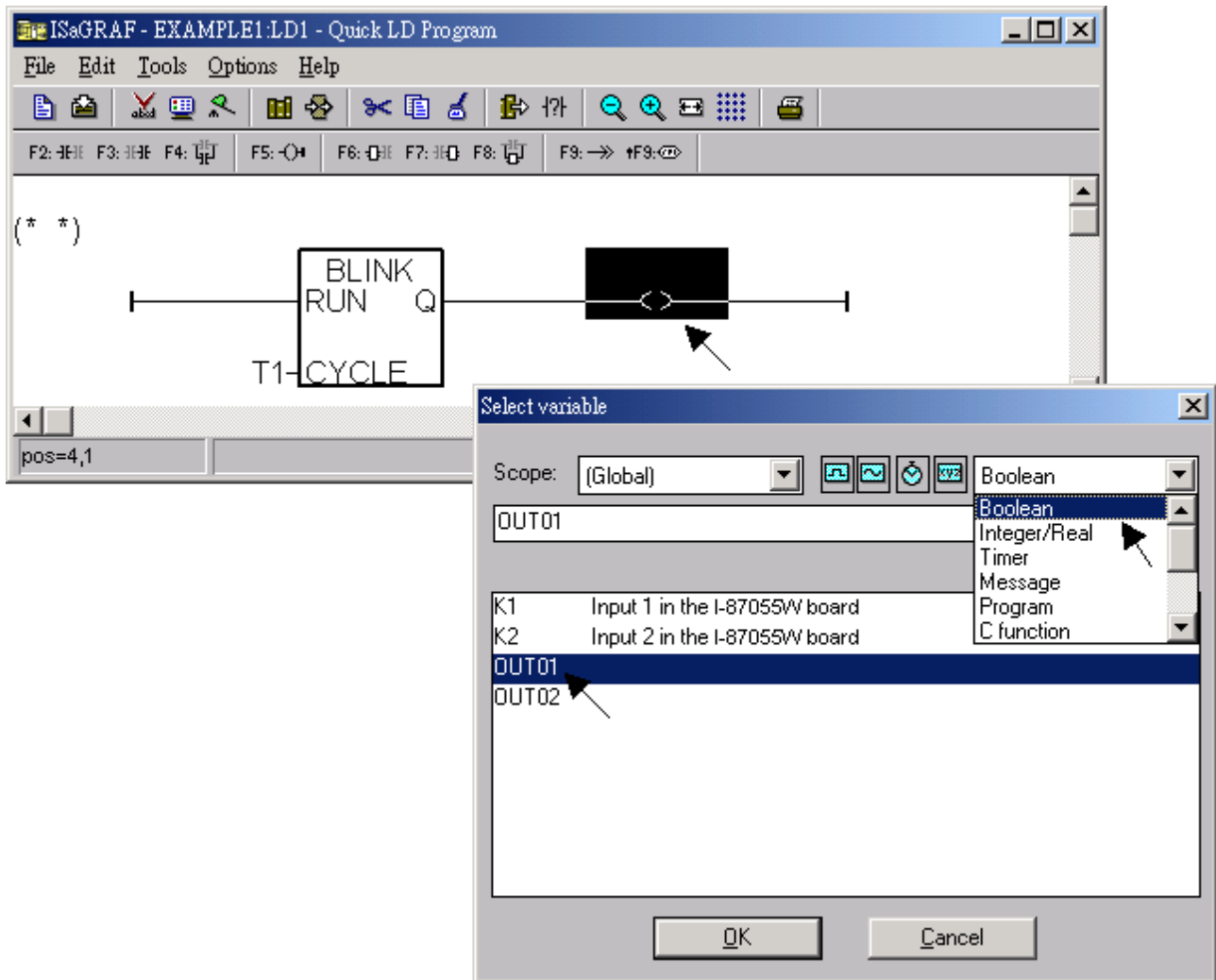
Now we are going to assign the associated variable & constant to each item. Double click anywhere inside of the block and the "Function Block" assignment window appears. Select the "BLINK" type function block. To learn how the "BLINK" function operates you can click on the "Info" button for a detailed explanation of its functionality



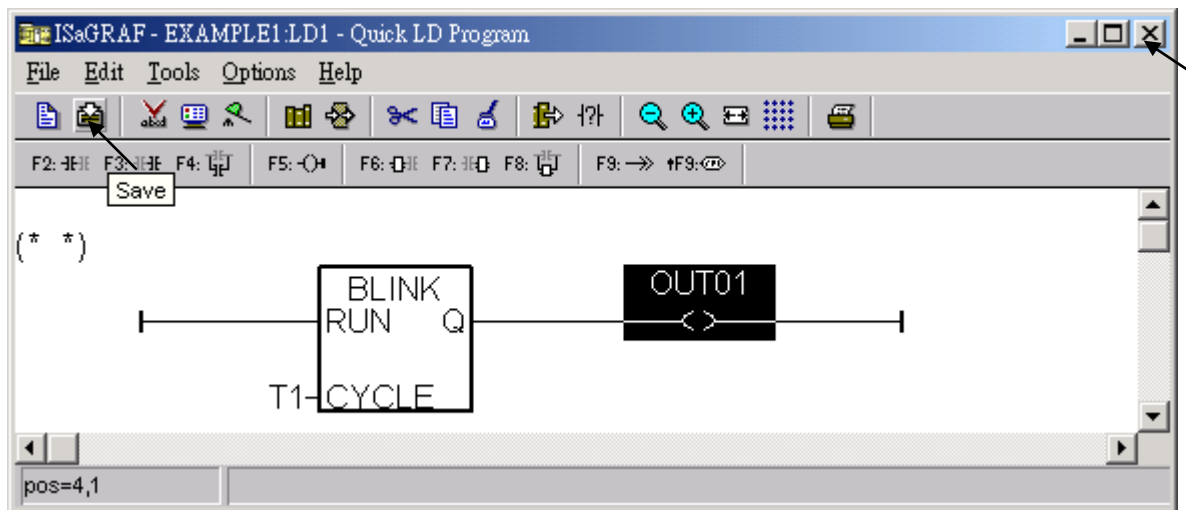
Now move your cursor to the left of the parameter “CYCLE” of the “BLINK” block. Double click on it, select “Timer” and then double click on variable name - “T1”



Move your cursor to the “coil”. Double click on it, select “Boolean” and then double click on variable name – “OUT01”.



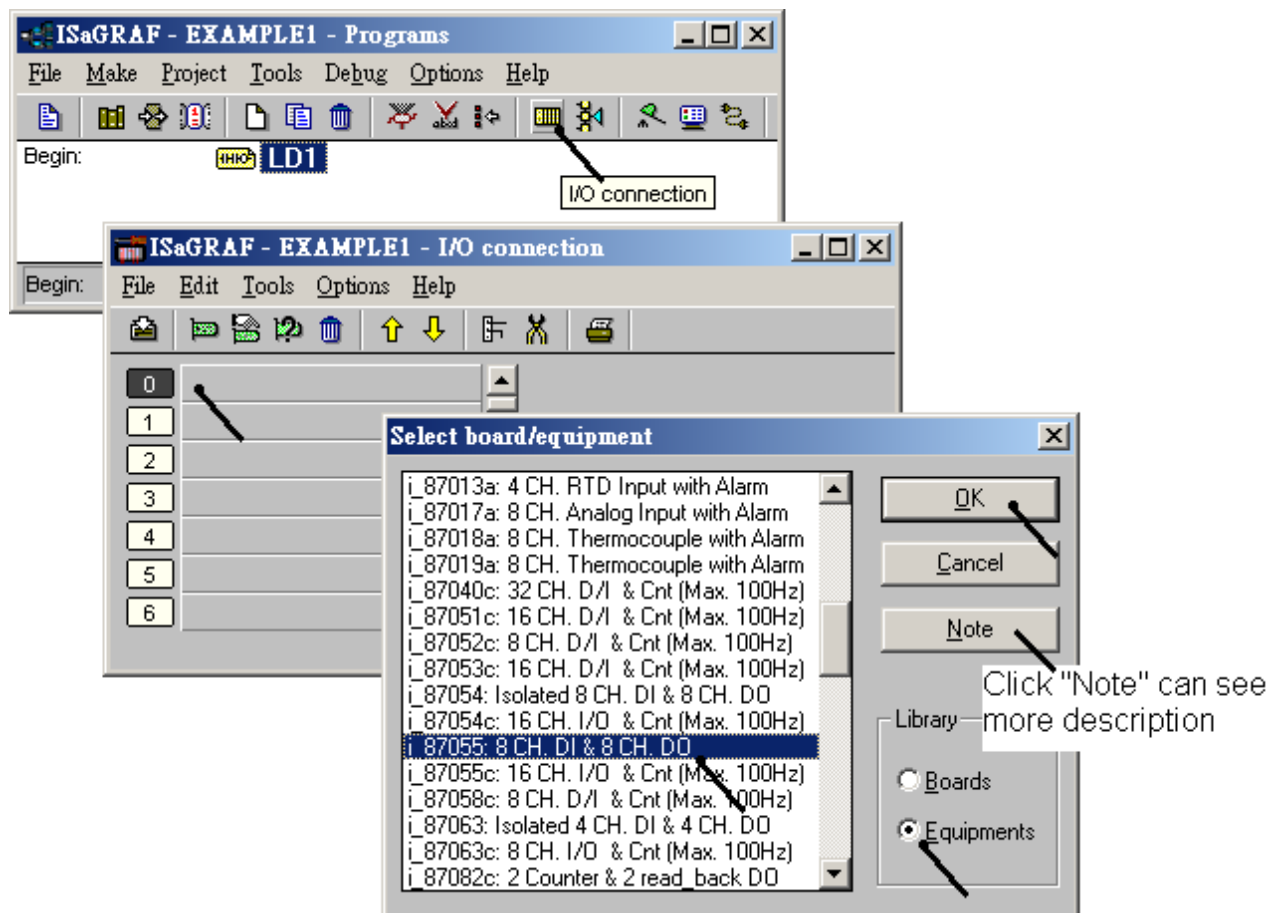
Now we have finished our Ladder code, click on “Save” and then click on “X” to exit.



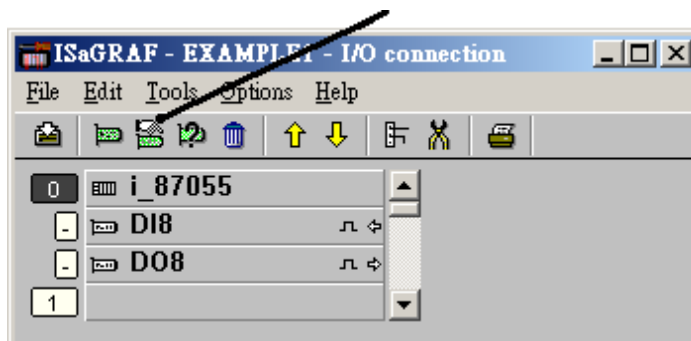
4.1.8: Connecting The I/O

We have defined variables name of “OUT01”, “OUT02” as “output” attribution, while “K1” & “K2” as “input” attribution in step 4.1.4. These “input” & “output” variables should be map to

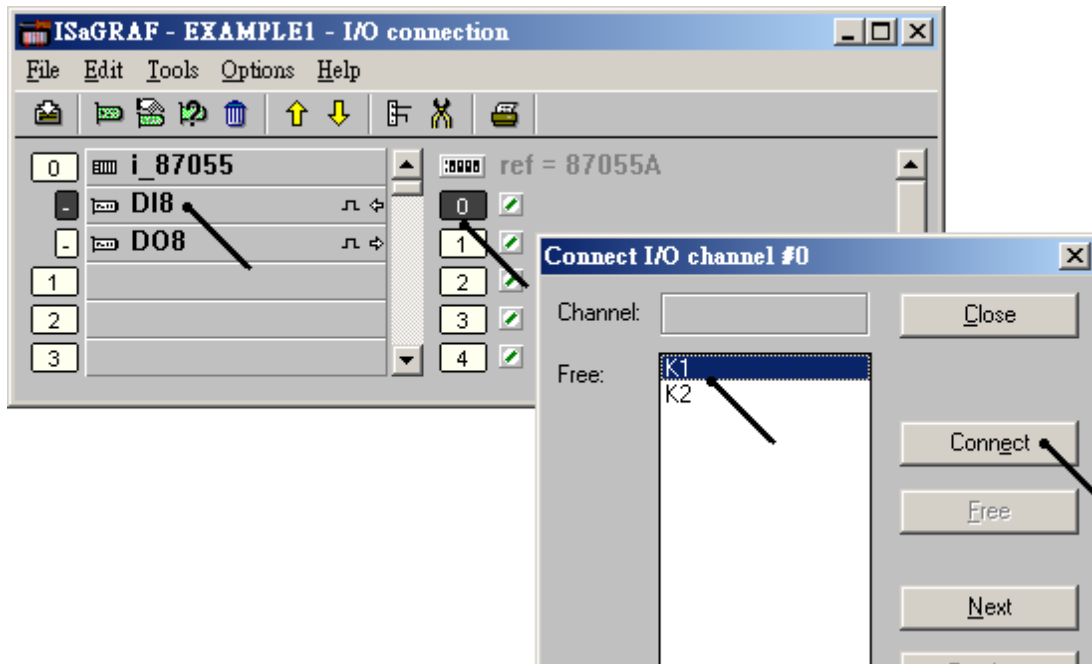
physical I/O in the controller before they can work. To do that, click on “I/O connection” to get into the I/O connection window. Double click on the No. 1 slot (Please make sure your I-87055W I/O board is plug in slot 0 of the WP-8xx7) & then check on the “Equipments” & double click on the “I_87055: 8 CH. DI & 8 CH. DO”.



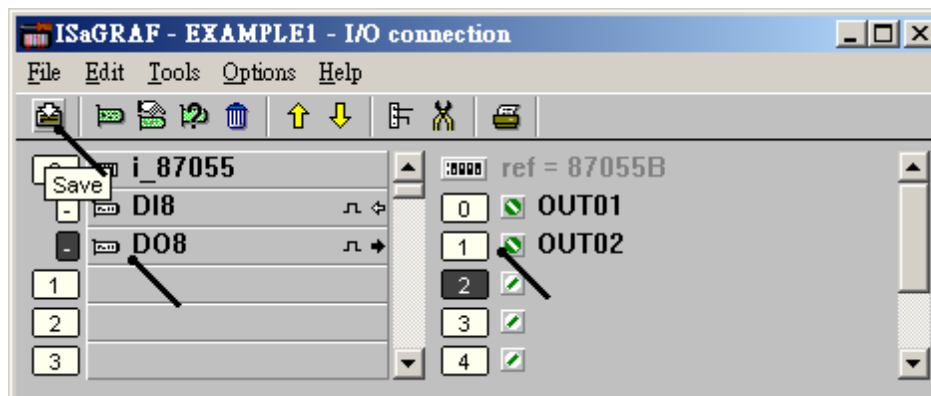
Then we have. (If you don't have the I-87055W, you may click the “Real / Virtual board” to make it become virtual board.)



To map input variables “K1” & “K2” to the input channel No. 1 & 2 of the “I-87055”, double click on the channel 1 and then click on “Connect”. Then click on “Connect” again to connect channel 2.



By the same way, please connect “OUT01” , “OUTPUT02” to output channel 1 to 2. Then we have below window. Click on “Save” and then exit.



IMPORTANT NOTICE:

1. I/O Slots 0 through 7 are reserved for REAL I/O boards that will be used in the WP-8xx7. You can use slot No. 8 and above for additional functionality.
2. All of the variables with “Input” and “Output” attribute MUST be connected through the I/O connection as described above for any program to be successfully compiled. Only the Input and Output attributed variables will appear in the "I/O Connections" window. In this example we have only 2 boolean output variables - OUT01, OUT02 and 2 boolean input variables – K1 & K2.

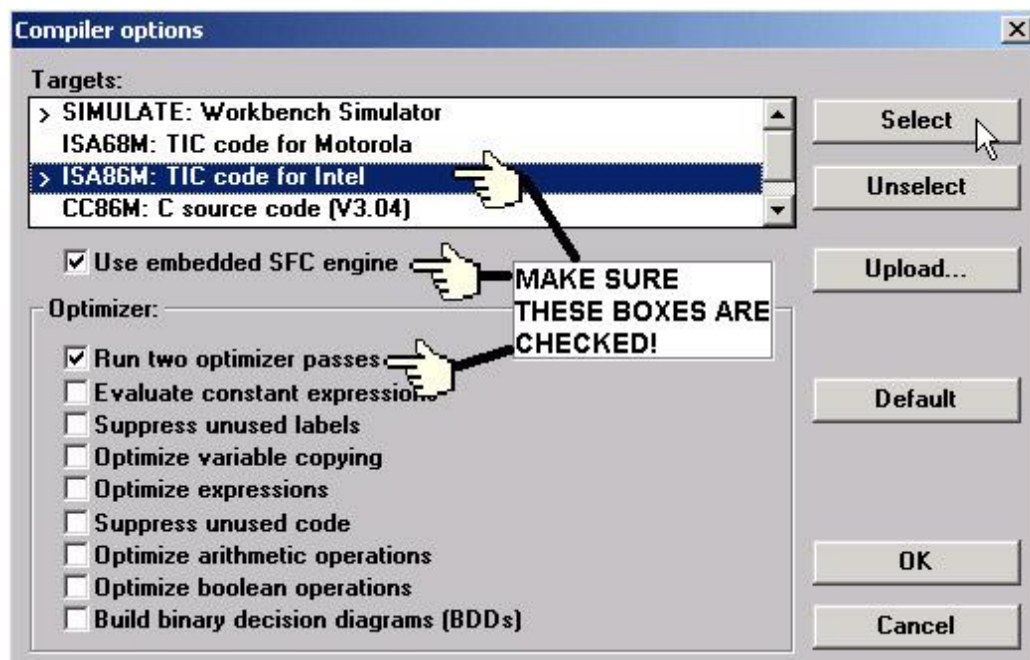
4.2: Compiling & Simulating The Example Project

For ANY AND EVERY ISaGRAF program to work properly with any of the I-7188EG, 7188XG, uPAC-7186EG, I-8417/8817/8437/8837 , iPAC-8xx7, W-8xx7 and WinPAC-8xx7 controller systems, it is the responsibility of the programmer to properly select the correct "Compiler Options". You MUST select the "ISA86M: TIC Code For Intel" option as described below.

To begin the compilation process, first click on the "MAKE" option from the main menu bar, and then click on "Compiler Options" as shown below.



The "Compiler Options" window will now appear. Make sure to select the options as shown below then press the "OK" button to complete the compiler option selections.

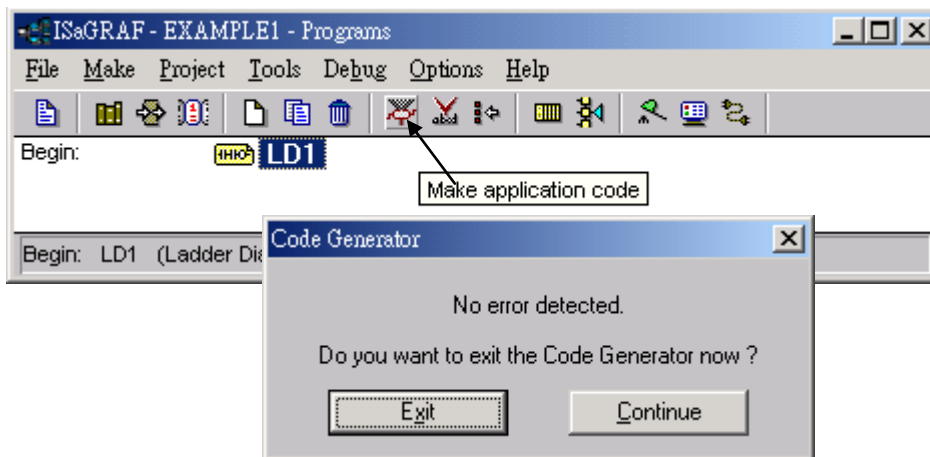


4.2.1: Compiling Error Result In Different ISaGRAF Version

Please refer to appendix H of this manual.

TIME TO COMPILE THE PROJECT!

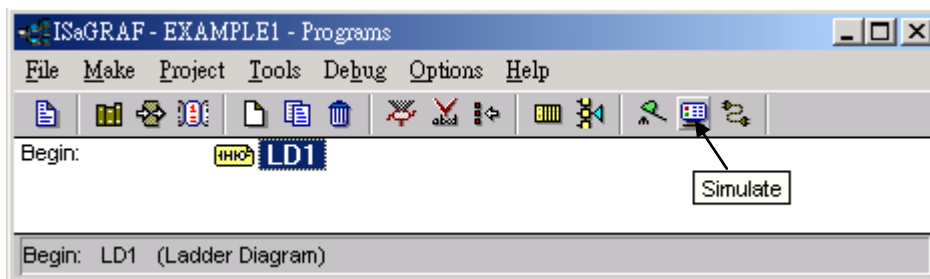
Now that you have selected the proper compiler options, click on the "Make Application Code" icon to compile the example project. If there are no compiler errors detected during the compilation process, CONGRATULATIONS, you have successfully created our example program.



If errors are detected during the compilation process, just click on the "CONTINUE" button to review the error messages. Return to the Project Editor and correct the errors as outlined in the error message window.

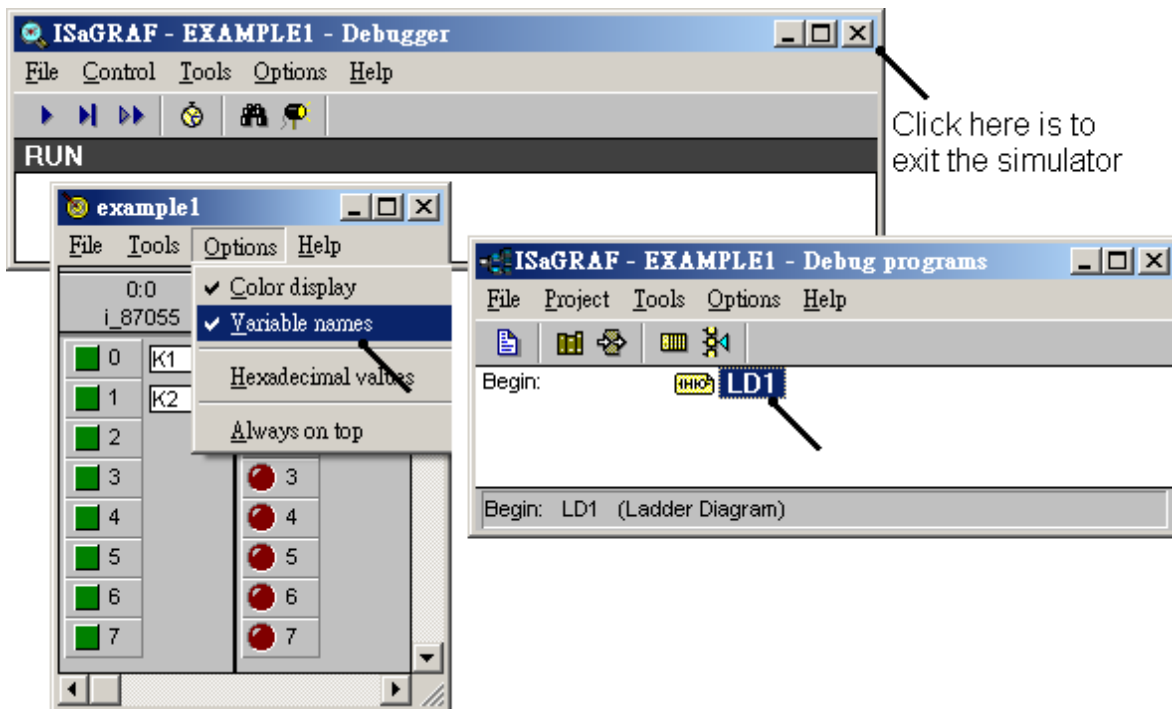
TIME TO SIMULATE THE PROJECT!

If the compilation is Ok, you may simulate the project on the PC to see how the program works without the controller. To do that, click on the "Simulate" icon.



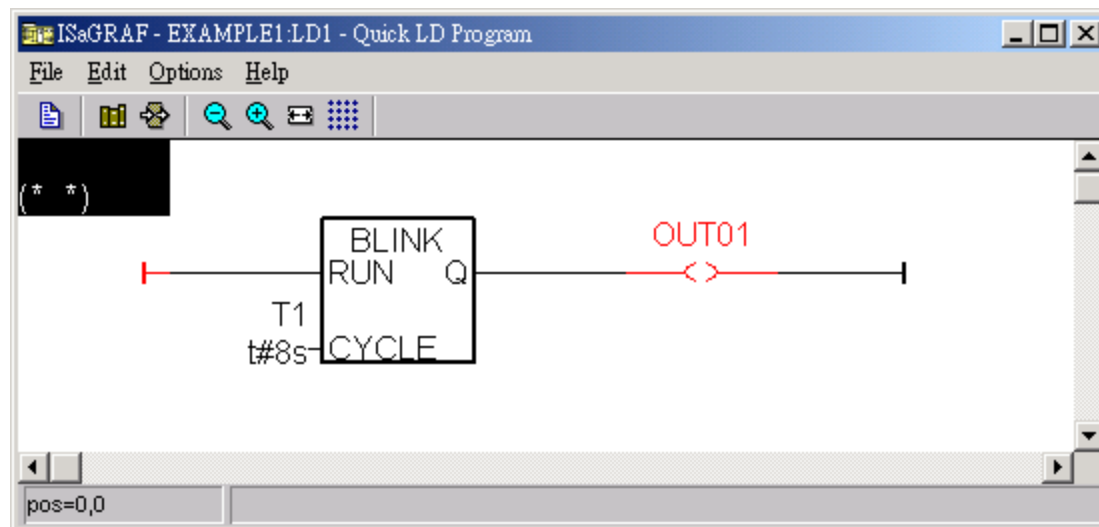
When you click on the "Simulate" icon three windows will appear. The windows are the "ISaGRAF Debugger", the "ISaGRAF Debug Programs", and the "I/O Simulator" windows. If the I/O variable names you have created DO NOT appear in the I/O simulator window, just click on the "Options" and "Variable Names" selection and the variable names you have created will now appear next to each of the I/O's in the simulator window.

In the "ISaGRAF Debug Program" window, double click on the "LD1" where the cursor below is positioned. This will open up the ISaGRAF Quick LD Program window and you can see the LD program you have created.



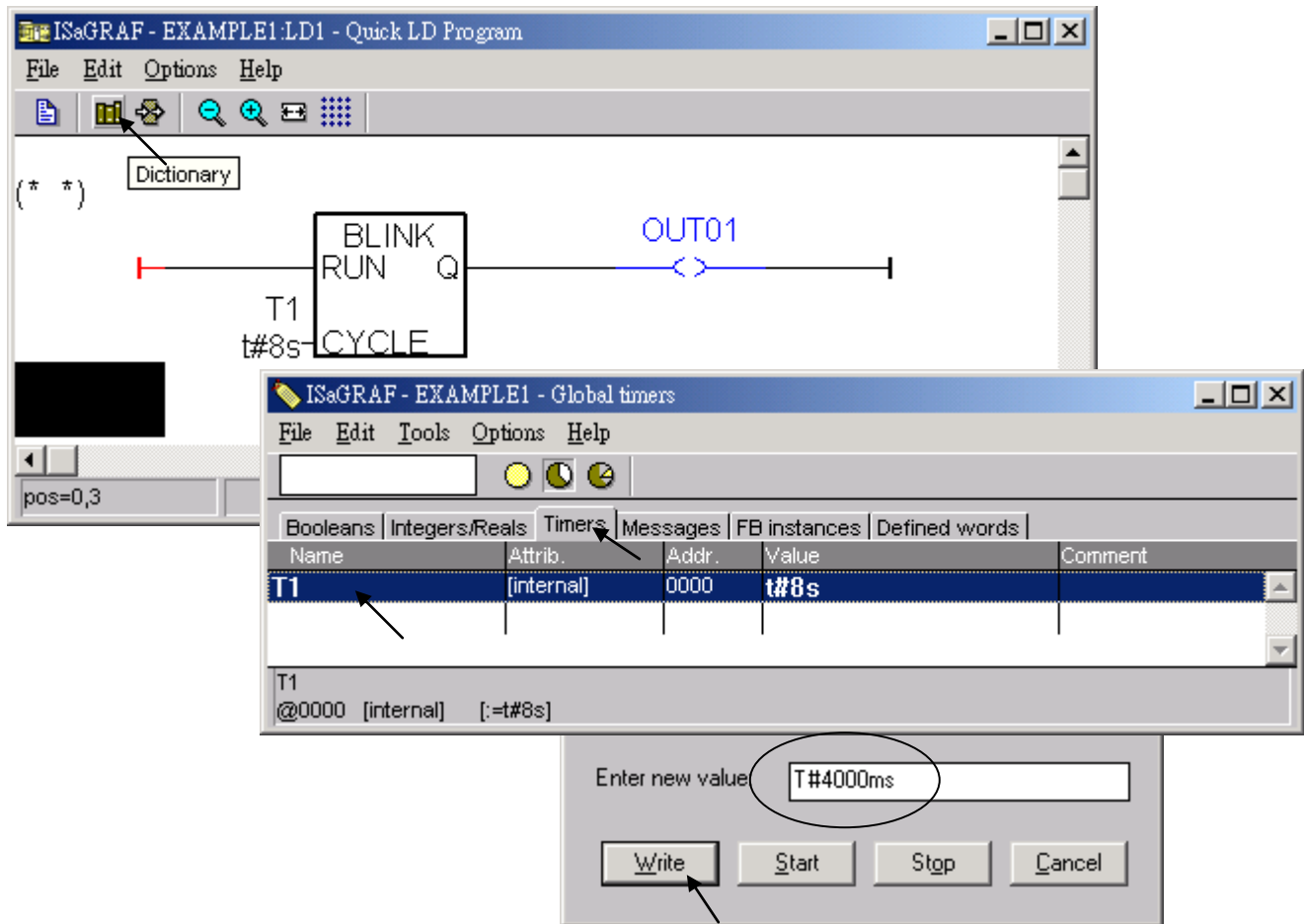
Running The Simulation Program

When you double click on "LD1" in the "ISaGRAF Debug Programs" window, the follow window should appear.

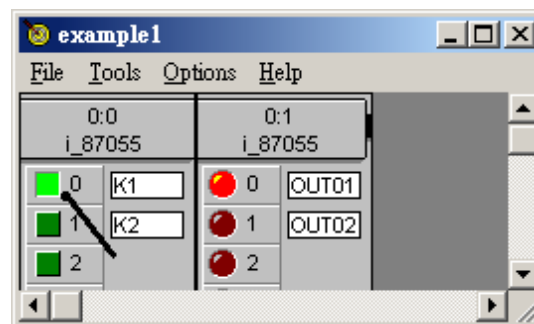


You can see outputs "OUT01" will blink in the period of 8 seconds.

You can adjust the "T1" variable while the program is running. To accomplish this, click on the "Dictionary" icon which will open the "ISaGRAF Global Variables" window as shown in the first two pictures below. Click on "Timer" tab and then double click on "T1" to change the timer value to "T#4000ms" (this means 4000 ms). Then click on "Write".



Now we are going to simulate the "K1" & "K2" input. Click on "K1" using the left button of the mouse.



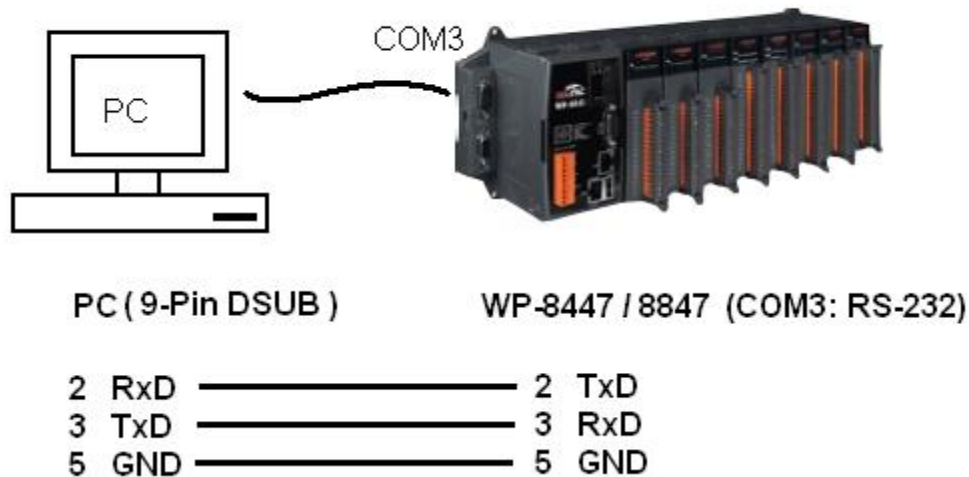
To exit simulation, please close the debugger window.



4.3: Download & Debug The Example Project

We have two ways to download the project to the WinPAC-8xx7. One is using Ethernet cable, the other one is using RS-232 cable. Here will show you the RS-232 way. **(Please refer to section 3.2.3.1 if you would like to download the project via Ethernet)**

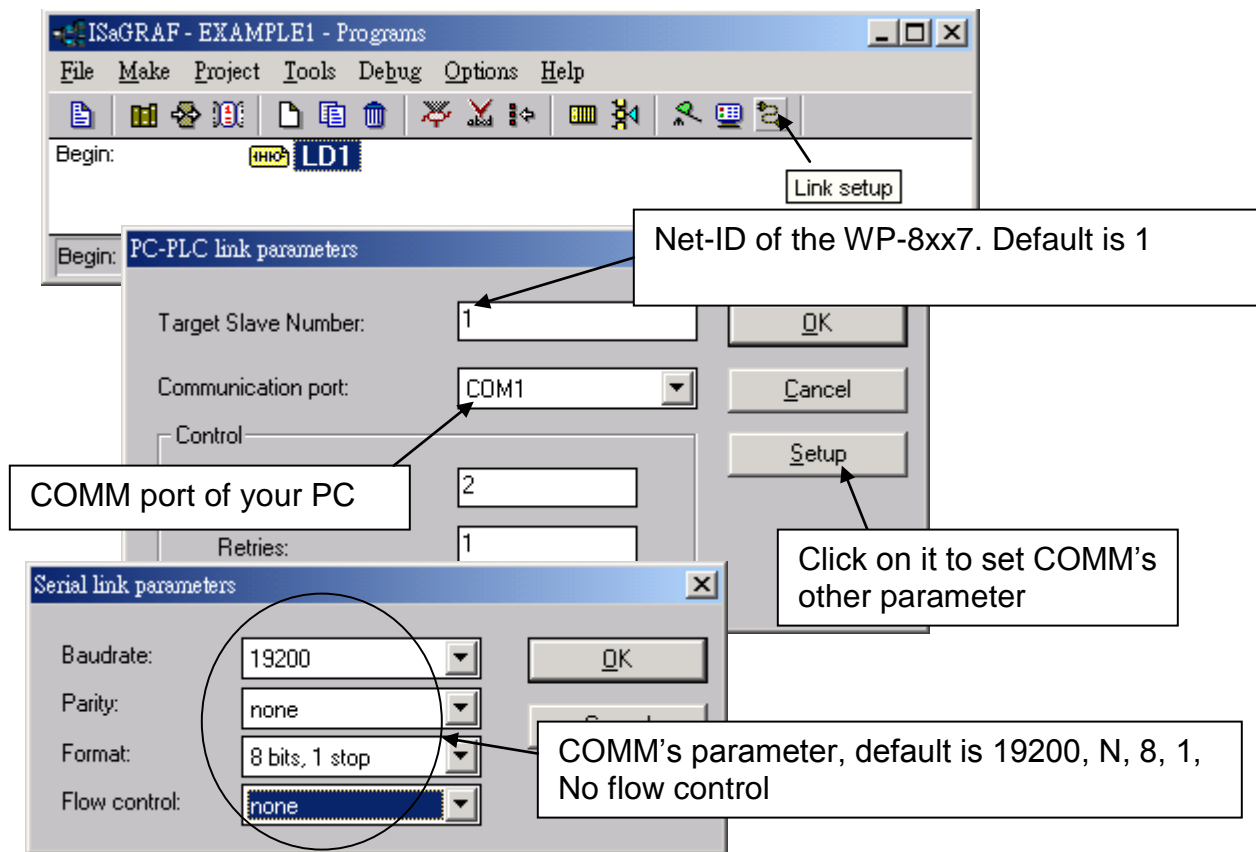
To begin this process, please install the hardware as below. The RS-232 cable wiring should be as below figure. **(Please make sure the “Modbus RTU Slave Port” is set as COM3 (refer to Appendix A.2, or it can only be download via Ethernet)**



Note: The WP-8147 doesn't have COM3. Only WP-8447 / 8847 have.

This section lists how to download the ISaGRAF program via RS-232 cable. However user may also use Ethernet cable to download program to the WinPAC-8xx7 (please refer to section 3.2.3.1)

Click on the "Link Setup" icon in the "ISaGRAF Programs" window. When you click on the "Link Setup" icon, the following window will appear. Please set the proper value.

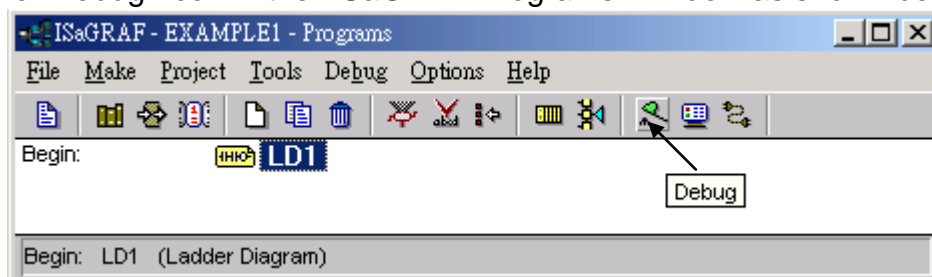


The RS-232 communication parameters for the target WP-8xx7 controller MUST be set to the same serial communication parameters for the development PC. For WP-8xx7 controllers (serial port communications), the default parameters for COM3 (RS-232) port are:

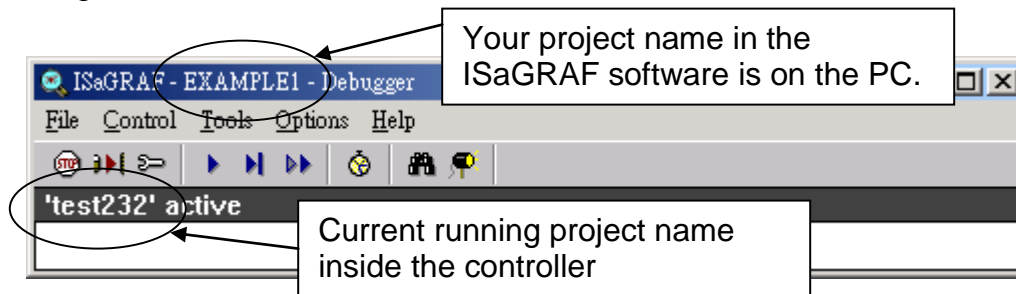
Baudrate: 19200
 Parity: none
 Format: 8 bits, 1 stop
 Flow control: none
 (Please refer to Appendix A.2 to setup COM3 as Modbus RTU slave port)

DOWNLOADING THE EXAMPLE PROJECT

Before you can download the project to the controller, you must first verify that your PC and the controller system are communicating with each other. To verify proper communication, click on the "Debug" icon in the "ISaGRAF Programs" window as shown below.



If the development PC and the WinPAC-8xx7 controller system are communicating properly with each other, the following window displayed below will appear (or if a program is already loaded in the controller system, the name of the project will be displayed with the word "active" following it).

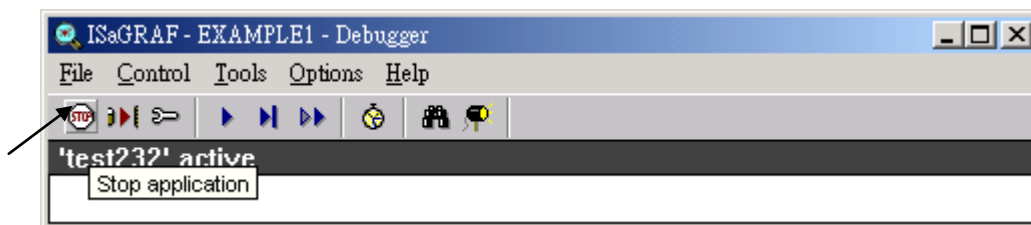


If the message in the "ISaGRAF Debugger" says "Disconnected", it means that the development PC and the controller system have not established communications with each other.

The most common causes for this problem is either the serial port cable not being properly configured, or the development PC's serial port communications DO NOT match that of the WP-8xx7 controller system.

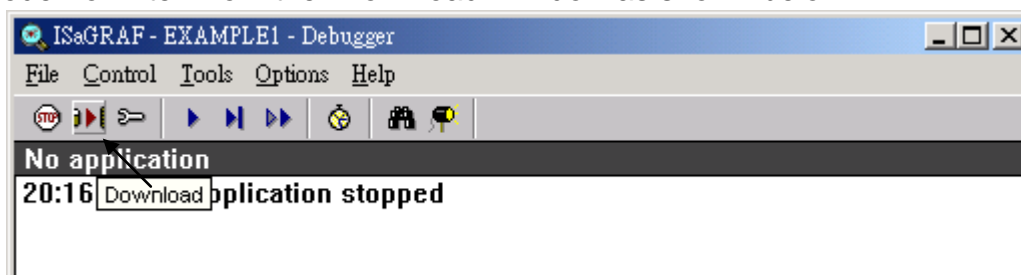
You may have to either change the serial port communication settings for the development PC (which may require changing a BIOS setting) or change the "Serial Link Parameters" in the ISaGRAF program.

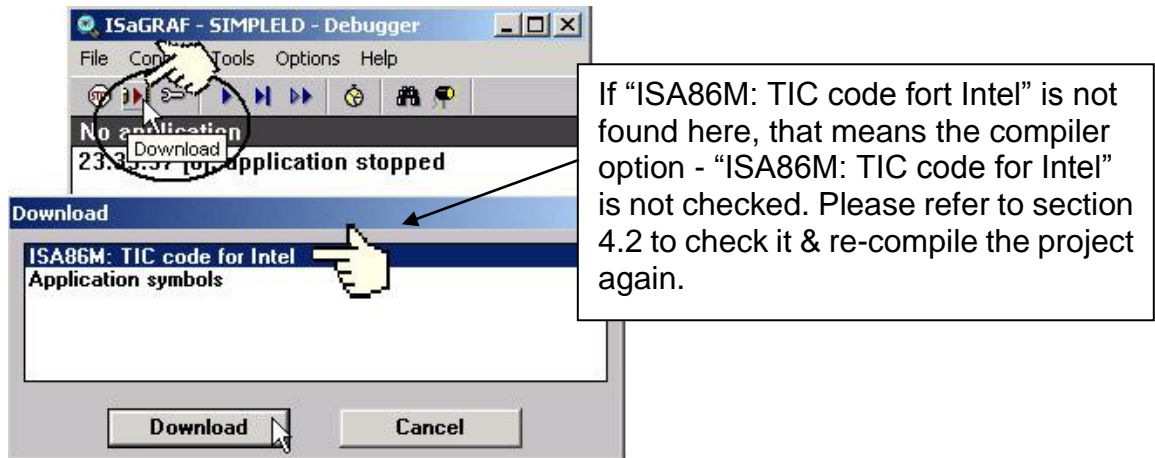
If there is a project already loaded in the controller system you will need to stop that project before you can download the example project. Click on the "STOP" icon as illustrated above to halt any applications that may be running.



STARTING THE DOWNLOADING PROCESS

From the "ISaGRAF Debugger" window click on the "Download" icon, then click on "ISA86M: TIC Code For Intel" from the "Download" window as shown below.

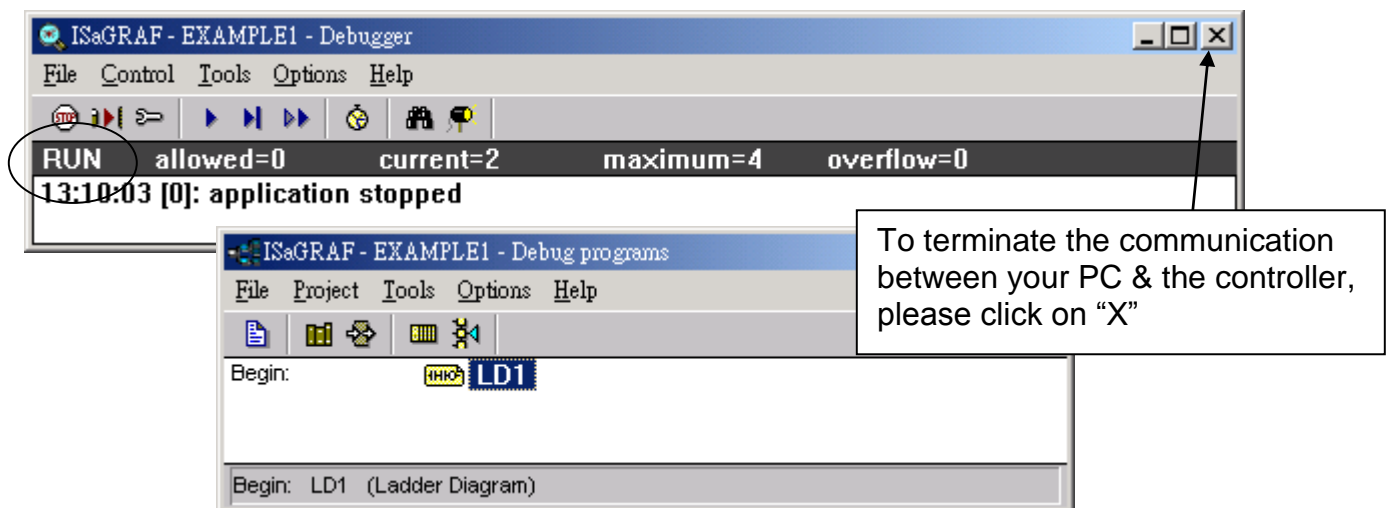




The example project will now start downloading to the WP-8xx7 controller system. A progress bar will appear in the "ISaGRAF Debugger" window showing the project downloading progress.

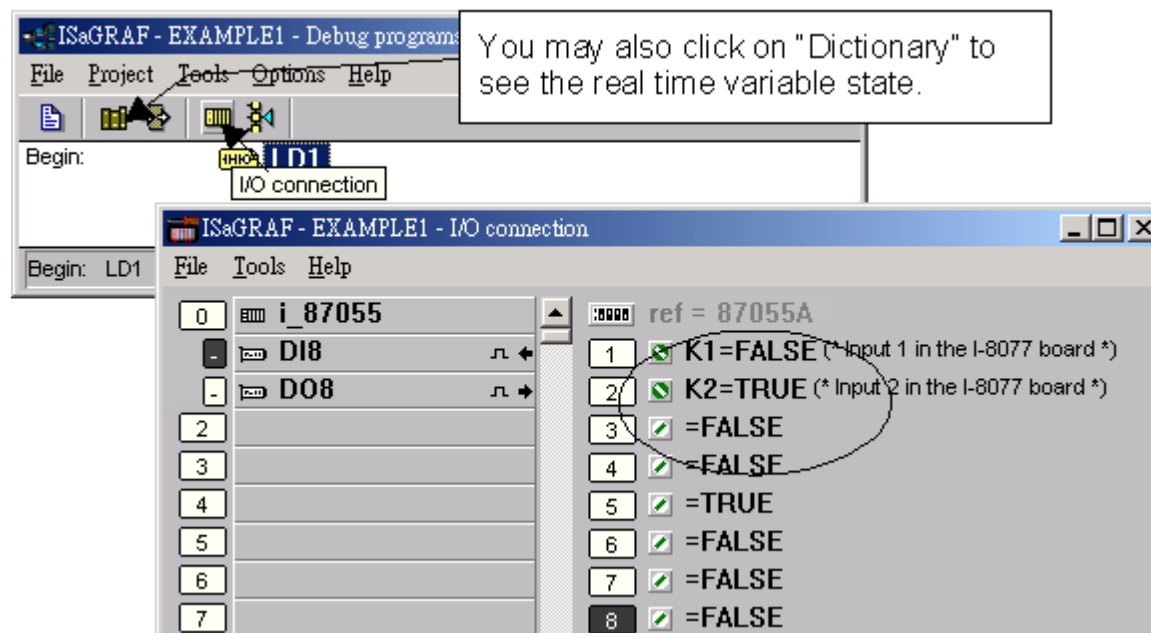


When the example project has successfully completed the downloading process to the W-8xx7 controller system the following two windows will appear.

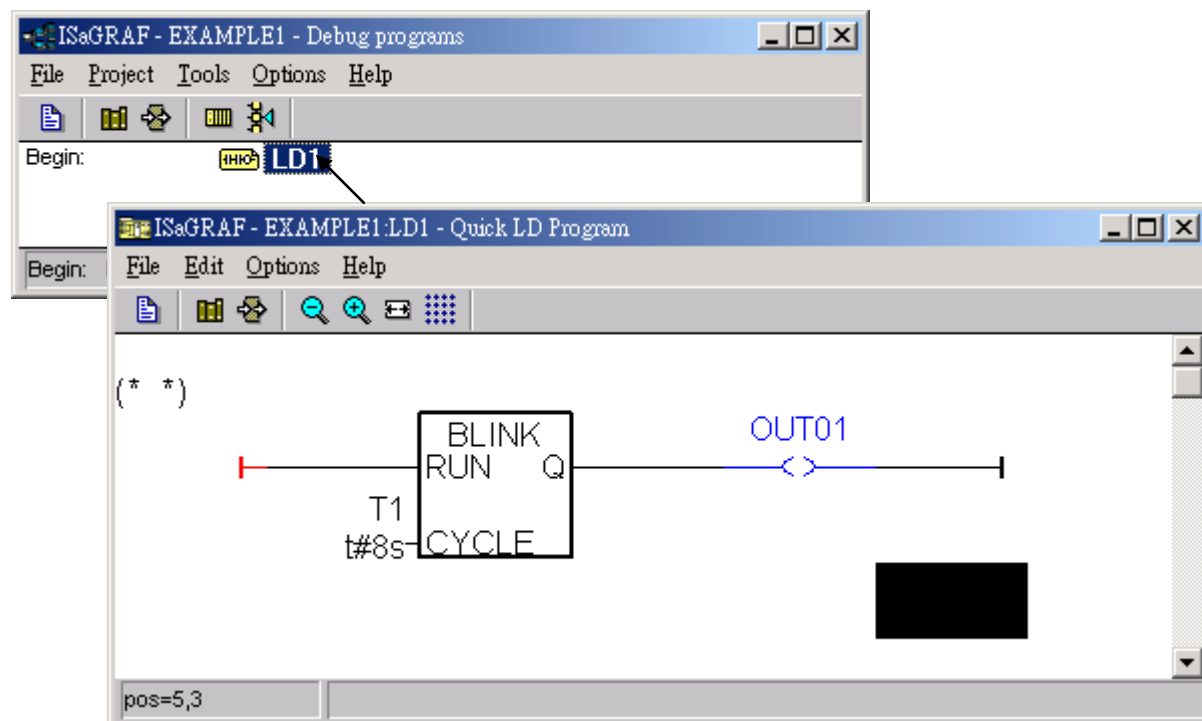


RUNNING THE EXAMPLE LD PROGRAM

You can observe the real time I/O status from several ISaGRAF windows while you are running the example project. One of the windows is the "I/O Connections" window, which shows each of the inputs and outputs as assigned. Click on the "I/O Connections" icon in the ISaGRAF Debugger window to open the "I/O Connections" screen. You may switch ON/OFF the D/I on the front panel of the I-87055W I/O board to see what happens about "K1" & "K2"



Another VERY helpful window you can open is the "Quick LD Program" window. From this window you can observe the LD program being executed in real time.



4.4: Design The Web Page

After finishing the ISaGRAF project & download it to the WinPAC-8xx7, we are going to design the Web Page for this ISaGRAF project.

If you haven't practiced "Setting Up A Web HMI Demo" listed in the Chapter 3, it's better to do it once to get familiar with it.

We will use "**Microsoft Office FrontPage 2003**" (or advanced version) to build web pages in this manual. User may choose your prefer web page editor to do the same thing.

You may refer to the finished web pages of this example in the WP-8xx7 CD-ROM at design time. However it is better to do it one time by yourself to get more understanding.

WinPAC-8xx7 CD: \napdos\isagraf\wp-8xx7\wp_webhmi_demo\example1\

4.4.1: Step 1 – Copy The Sample Web HMI pages

This is a sample Web HMI pages in the WinPAC-8xx7 CD-ROM:

\napdos\isagraf\wp-8xx7\wp_webhmi_demo\sample\

Please copy this "sample" folder to your drive and rename it, for example, "**example1**".

The basic Web HMI files includes 2 folders and 3 DLL files and 4 htm files as below.

./img/	(default image files - *.jpg , *.bmp , *.gif)
./msg/	(default message files – wincon.js & xxerror.htm)

whmi_filter.dll	(three DLL files)
login.dll	
main.dll	

index.htm	(first default page)
login.htm	(the Web HMI welcome page)
menu.htm	(the page-menu page, normally on the left on the Internet Explorer)
main.htm	(first page when successfully login)

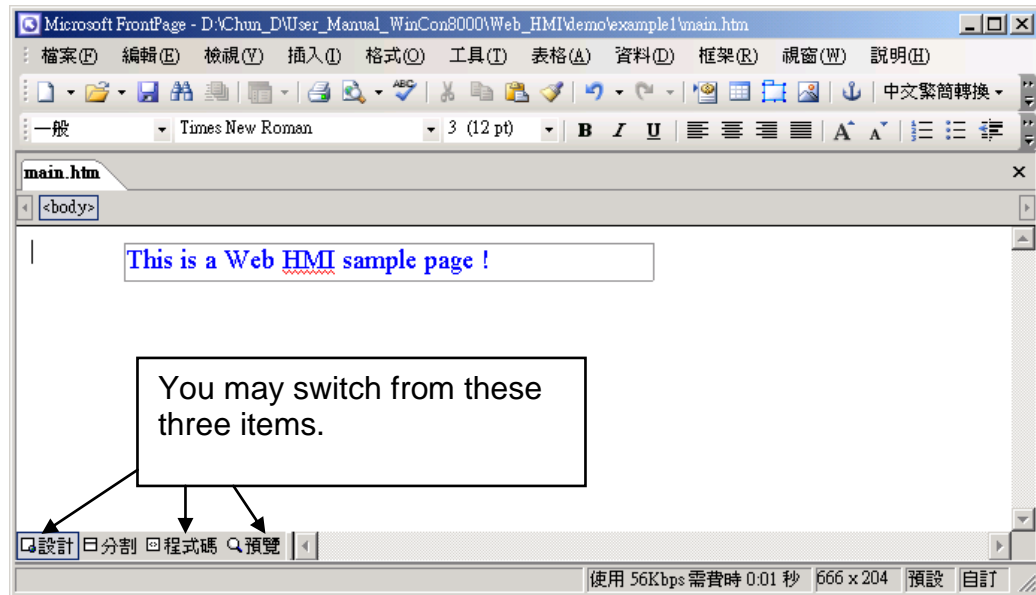
User may put his own image files into the folder named as "user_img". And put user-defined java script file or css file into the folder named as "user_msg". Other folder name is not acceptable by the Wincon Web HMI.

The "index.htm" file is the default entry page of the web server. User should not modify it. The "index.htm" re-directs to the "login.htm" file in 1 to 2 second when someone visits the WinPAC-8xx7 via the Internet Explorer.

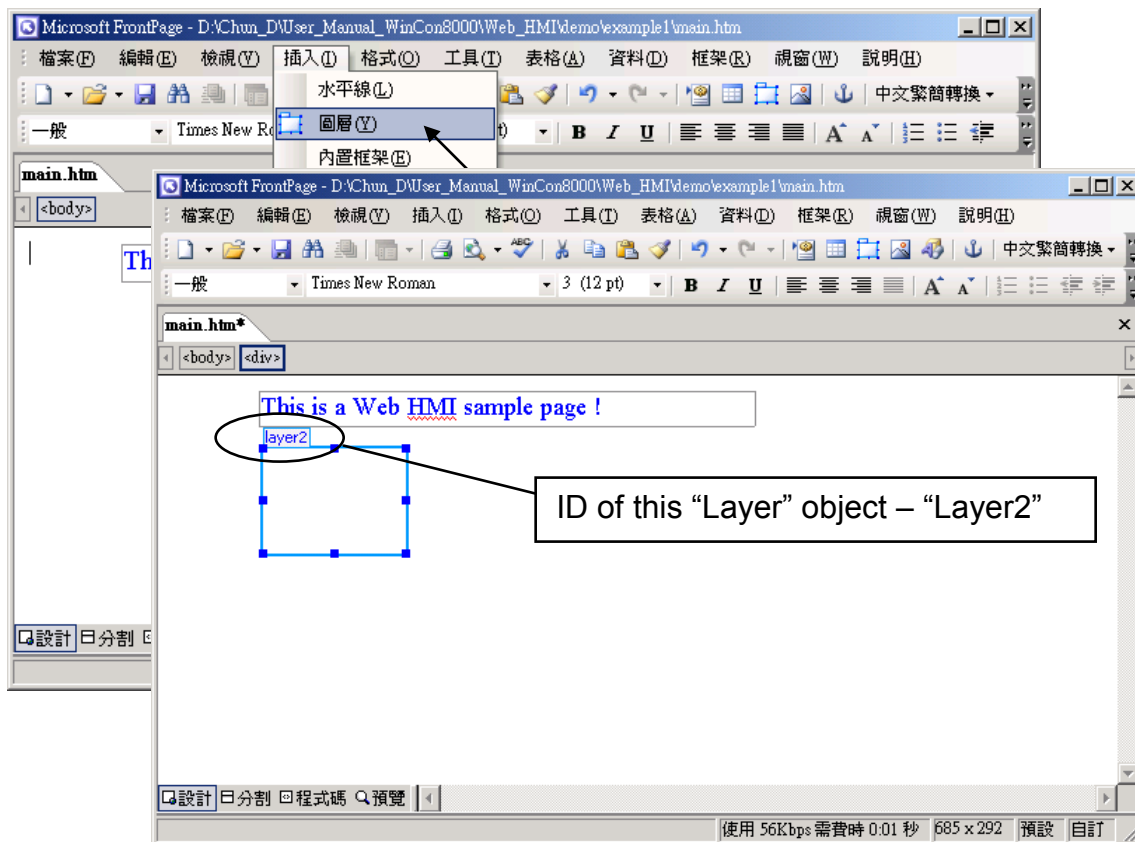
User may modify the "login.htm" , "menu.htm" & "main.htm" to fit his own need. We will only modify the "main.htm" in this example.

4.4.2: Step 2 – Building The Main.htm

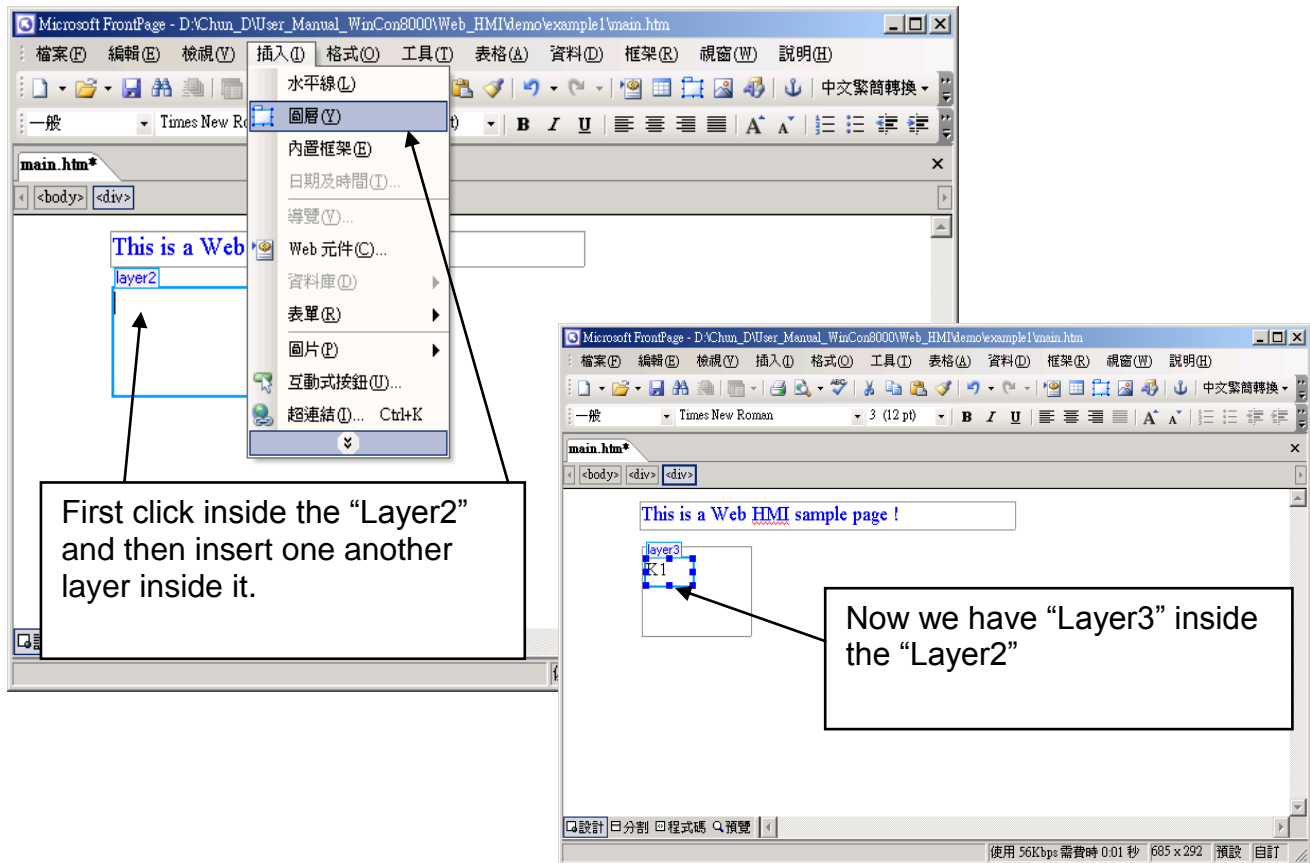
Please run the Microsoft Office FrontPage 2003 (or advanced version) and open the “main.htm”



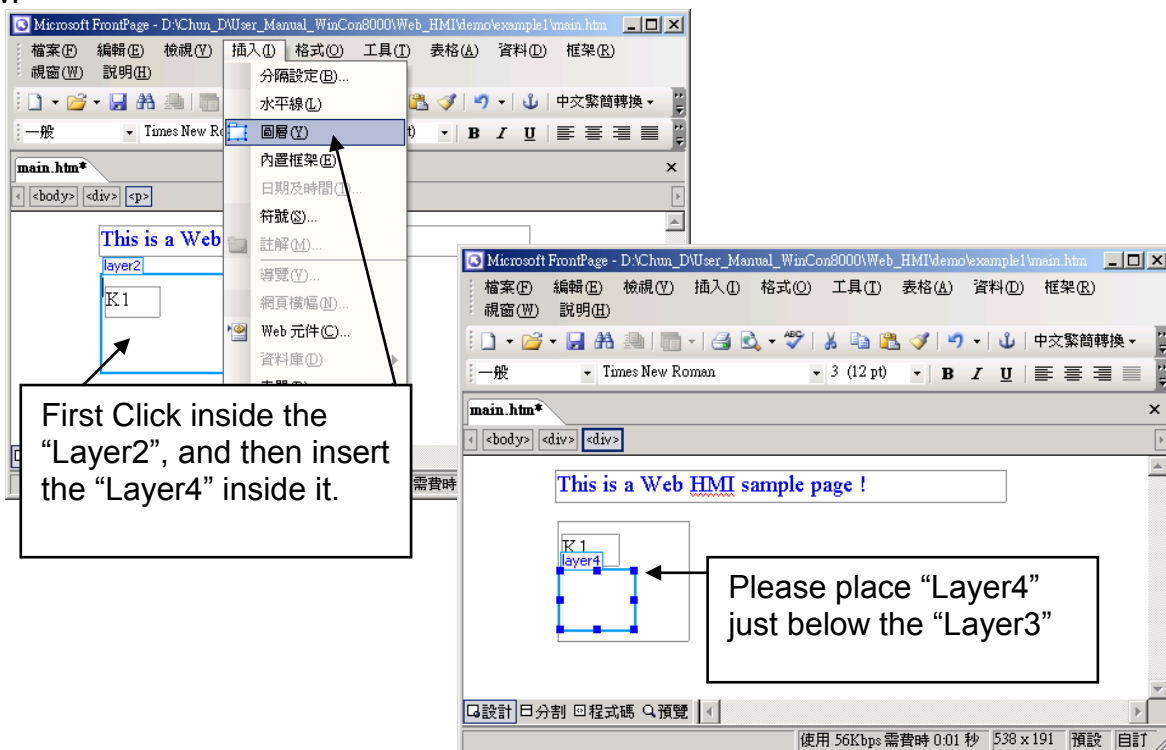
Please switch the window to design the page.
Please insert a layout object – “Layer” as below.

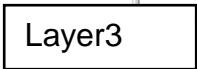
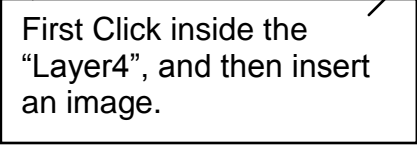


Click inside this “Layer” and then insert one another layer inside it as below. Please enter “K1” into the new created “Layer”.

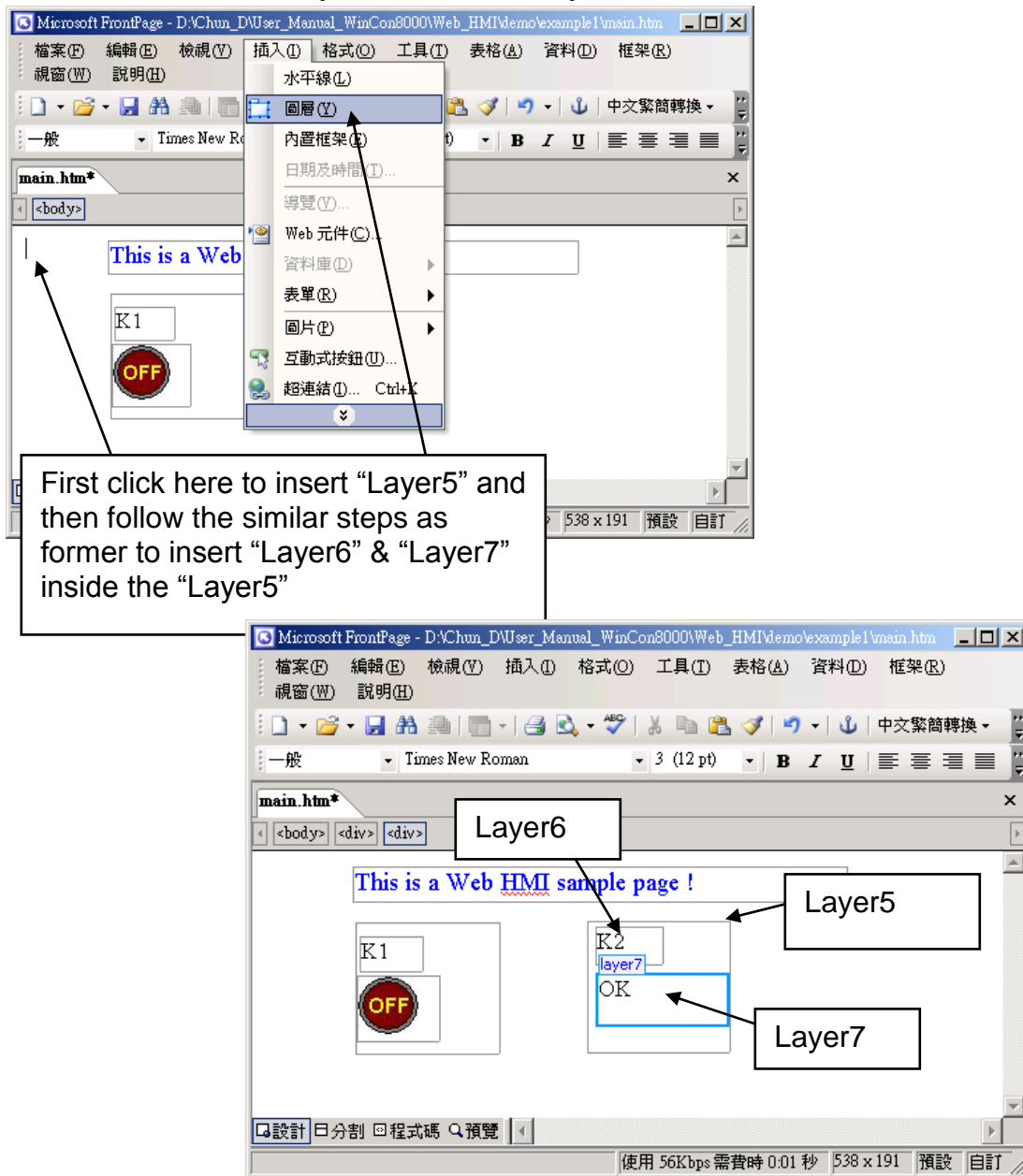


Follow the same former steps to insert one another “Layer” to be in just below the “Layer3” as below.



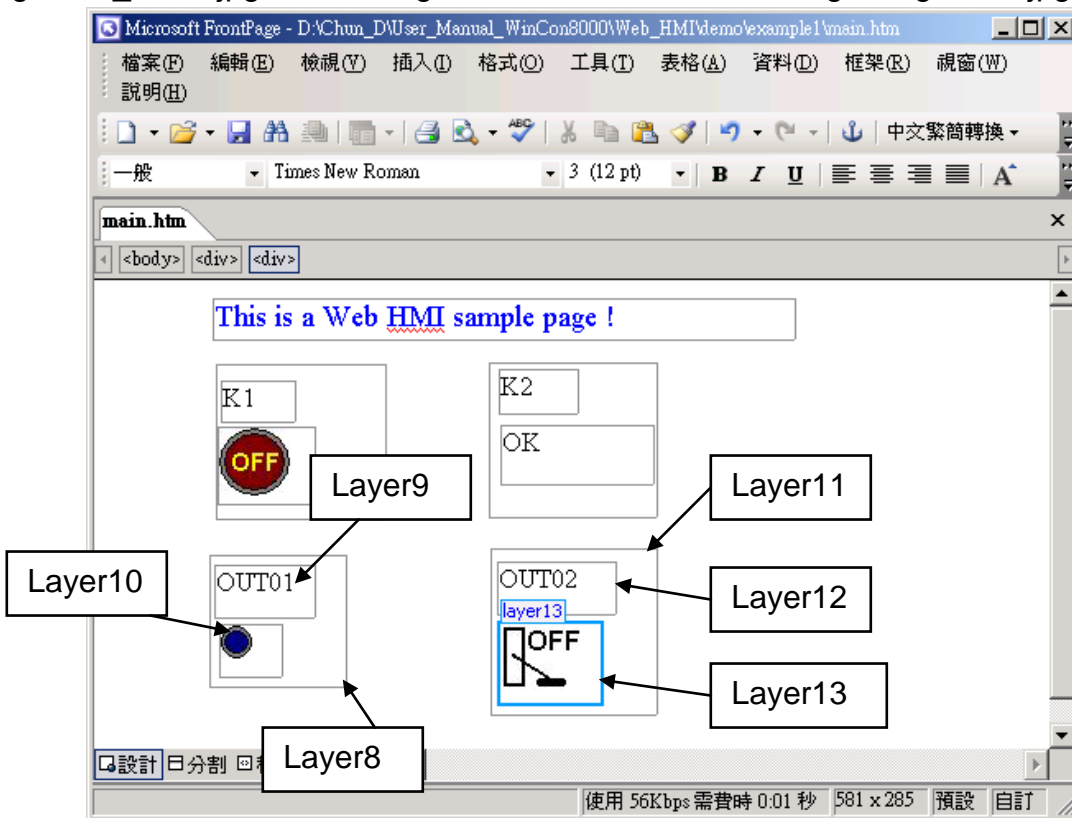


Please follow the similar steps to insert one another “Layer5” and one “Layer6” with a “K2” symbol inside it, and also a “Layer7” with a “OK” symbol inside it as below.



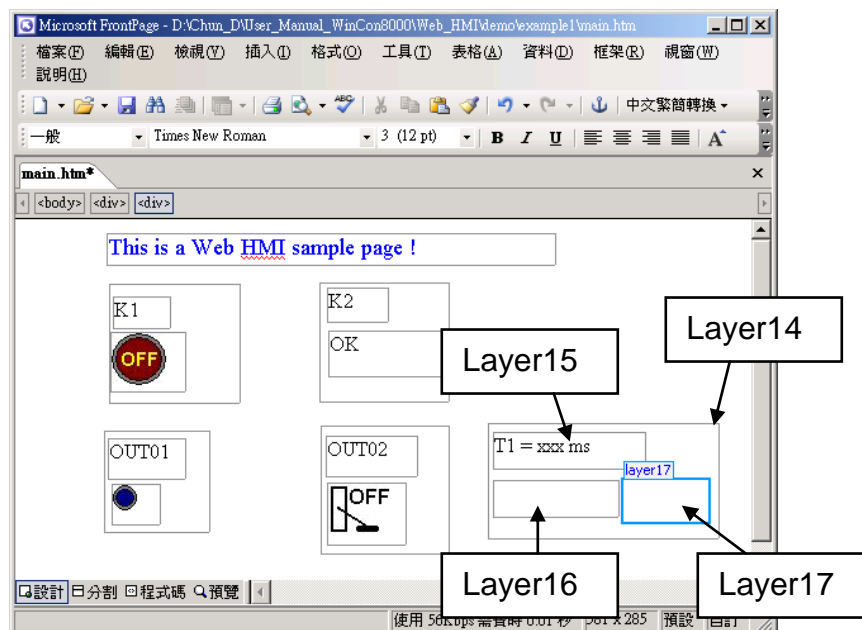
We will use “K1” to display the state of the first input of the I-87055W board, and “K2” for its second input.

Please follow the similar steps to insert “OUT01” & “OUT02” as below. The OUT01 uses “./img/circle_blue0.jpg” as its image source, while OUT02 using “./img/cmd0.jpg”.

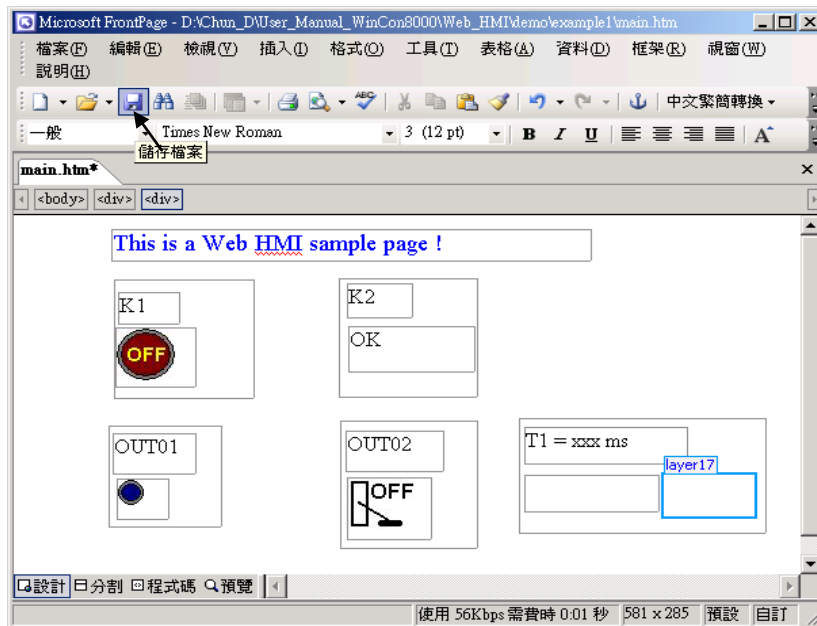


We will use OUT01 to display the state of the first output of the I-87055W board, while “OUT02” is for controlling and displaying the second output of the I-87055W.

Now please insert one another “Layer14”. Inside the “Layer14” please insert one “Layer15” with a “T1 = xxx ms” symbol. And two empty Layers – “Layer16” & “Layer17” just below the “Layer15”. We will use T1 to display the Timer value “T1” in the ISaGRAF project.



Click on “Save” to save this page.



4.4.3: Step 3 – Adding Control Code To The Main.htm

Please switch the window to the source code. A valid HTML document will contain the basic objects as below.

If you want to know more about the Web HMI's source code, please refer to Chapter 5.

```
<html>
<title>Your Title here</title>
<head>
<SCRIPT LANGUAGE="JavaScript">
</SCRIPT>
</head>
<body>
</body>
</html>
```

JavaScript code is normally placed inside the “head” area.

The “body” area describes the behavior of this page.

Please go to the body area and then modify the code as below.

Caption Area: Layer1
A Layer is starting with "<div " & ending with "</div>" tag

```
<!-- Caption -->
<font color="blue" size="4">
<div style="position: absolute; width: 353px; height: 24px; z-index: 1; left: 73px; top: 12px" id="layer1">
This is a Web HMI sample page !</div>
</font>
```

K1 Area: Layer2 to Layer4

```
<div style="position: absolute; width: 102px; height: 93px; z-index: 2; left: 75px; top: 52px" id="layer2">
<div style="position: absolute; width: 44px; height: 24px; z-index: 1; left: 3px; top: 10px" id="layer3">
K1</div>
<div style="position: absolute; width: 58px; height: 46px; z-index: 2; left: 1px; top: 38px" id="layer4">
</div>
<p>&nbsp;</div>
```

Please insert name="B11" just after the "<img "

K2 Area: Layer5 to Layer7

```
<div style="position: absolute; width: 101px; height: 93px; z-index: 3; left: 241px; top: 51px" id="layer5">
<div style="position: absolute; width: 47px; height: 26px; z-index: 1; left: 6px; top: 4px" id="layer6">
K2</div>
<div style="position: absolute; width: 92px; height: 35px; z-index: 2; left: 7px; top: 38px" id="layer7">
```

```
<font id="font_B12" color="blue" size="3">
<b id="B12"> OK </b>
</font> </div>
```

```
<p>&nbsp;</div>
```

Please modify "OK <div>" to become

```
<font id="font_B12" color="blue" size="3">
<b id="B12"> OK </b>
</font> </div>
```

OUT01 Area: Layer8 to Layer10

```
<div style="position: absolute; width: 82px; height: 79px; z-index: 4; left: 71px; top: 168px" id="layer8">
<div style="position: absolute; width: 60px; height: 31px; z-index: 1; left: 3px; top: 6px" id="layer9">
OUT01</div>
<div style="position: absolute; width: 37px; height: 31px; z-index: 2; left: 6px; top: 42px" id="layer10">
</div>
<p>&nbsp;</div>
```

Please insert name="B1" just after the "<img "

OUT02 Area: Layer11 to Layer13

```
<div style="position: absolute; width: 100px; height: 100px; z-index: 5; left: 242px; top: 164px" id="layer11">
```

```
<div style="position: absolute; width: 71px; height: 31px; z-index: 1; left: 4px; top: 8px" id="layer12">
OUT02</div>
<div style="position: absolute; width: 61px; height: 48px; z-index: 2; left: 5px; top: 45px" id="layer13">
</div>
```

```
<form name="form_B2" method="post" action="./main.dll">
  <input name="BEGIN" type="hidden">
  <input name="B2" type="hidden" value="0">
  <input name="END" type="hidden">
</form>
```

```
<p>&nbsp;</div>
```

Please insert
Style="cursor:hand" name="B2" onclick="ON_OFF(form_B2, form_B2.B2,
boolean_val[2])"
just after the "<img " tag

Please insert

```
<form name="form_B2" method="post" action="./main.dll">
  <input name="BEGIN" type="hidden">
  <input name="B2" type="hidden" value="0">
  <input name="END" type="hidden">
</form>
```

T1 Area: Layer14 to Layer17

```
<div style="position: absolute; width: 181px; height: 90px; z-index: 6; left: 374px; top: 162px" id="layer14">
<div style="position: absolute; width: 119px; height: 28px; z-index: 1; left: 4px; top: 7px" id="layer15">
```

```
T1 = <b id="T1">xxx ms</b></div>
```

Please modify "T1 = xxx ms </div>" to become
T1 = <b id="T1">xxx ms</div>

```
<div style="position: absolute; width: 98px; height: 28px; z-index: 2; left: 4px; top: 45px" id="layer16">
```

```
<form name="form_L21" method="post" action="./main.dll">
  <input name="BEGIN" type="hidden">
  <input name="L21" type="text" size="8" value="xxx">
  <input name="END" type="hidden">
</form>
```

```
&nbsp;</div>
```

Please insert below code inside "Layer16"

```
<form name="form_L21" method="post" action="./main.dll">
  <input name="BEGIN" type="hidden">
  <input name="L21" type="text" size="8" value="xxx">
  <input name="END" type="hidden">
</form>
```

```
<div style="position: absolute; width: 67px; height: 33px; z-index: 3; left: 106px; top: 44px" id="layer17">
```

<input type="button" value="Enter" onclick="Check_L21()">

 </div>
<p> </div>

Inside the "Layser17", please insert
<input type="button" value="Enter" onclick="Check_L21()">

We have finished the code in the <body> </body> area.

Now please go to the "head" area

In the "head" area, please modify the sample code to be as below.

// variable to record object's blink state, 0:not blink, 1: blink, For example:

// *****

var B12_blink=0; // init as 0:not blink

// *****

// function to blink object

var blink_step=0;

function blink_obj()

{

if(blink_step==1)

{

blink_step=0;

// display your object here

// blink B12, For example:

// *****

if(B12_blink==1)

{

B12.innerText="Error !" ;

font_B12.color="red";

}

// *****

}

else

{

blink_step=1;

// un-display your object here

// blink B12, For example:

// *****

if(B12_blink==1)

{

B12.innerText="" ;

font_B12.color="red";

}

// *****

The "Error !" symbol will blink when the K2 =
True in this example. Please un-mask the
code inside these 3 areas.


```

}
setTimeout("blink_obj()", blink_period);
}

```

We need a function “Check_L21 to check the entered T1 value and post it to the Wincon. Please un-mask the sample code to be as below.

```

// form sample, to check value of L21 & then post val to controller
// For example:
// *****

```

```

function Check_L21()
{
  var val=form_L21.L21.value;
  if(val>12000 || val<4000)
  {
    alert("T1's value should be in the range of 4000 to 12000");
    return;
  }
  Check(form_L21); // post value to the controller
}

```

```

// *****

```

And also inside the “refresh_data() “ function, please insert below code.

// To refresh displayed data, this function is called by IE about every 1.5 sec later

```

function refresh_data()
{
  B1.src = "./img/circle_blue" + boolean_val[1] + ".jpg";
  B2.src = "./img/cmd" + boolean_val[2] + ".jpg";

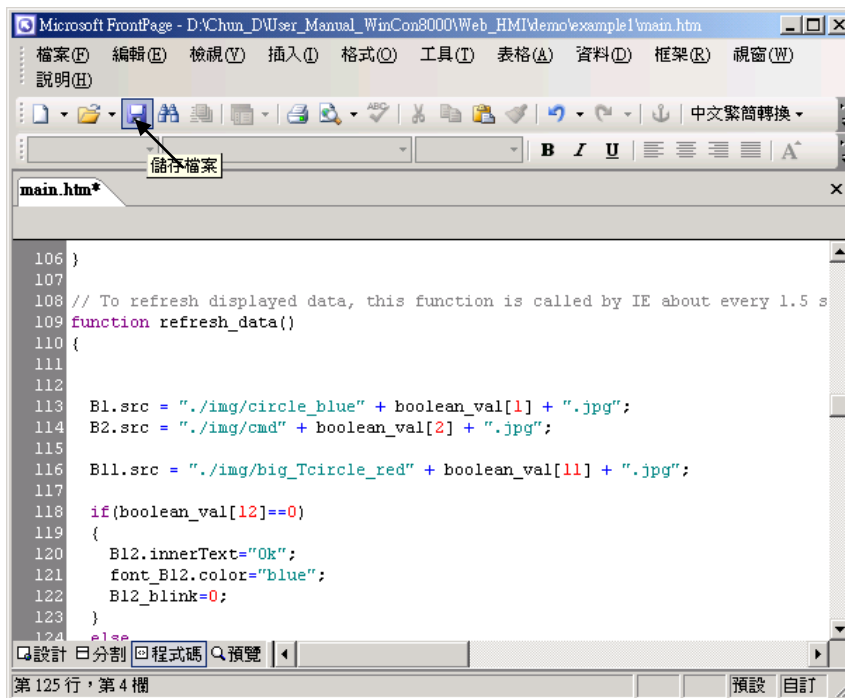
  B11.src = "./img/big_Tcircle_red" + boolean_val[11] + ".jpg";

  if(boolean_val[12]==0)
  {
    B12.innerText="Ok";
    font_B12.color="blue";
    B12_blink=0;
  }
  else
  {
    B12_blink=1;
  }

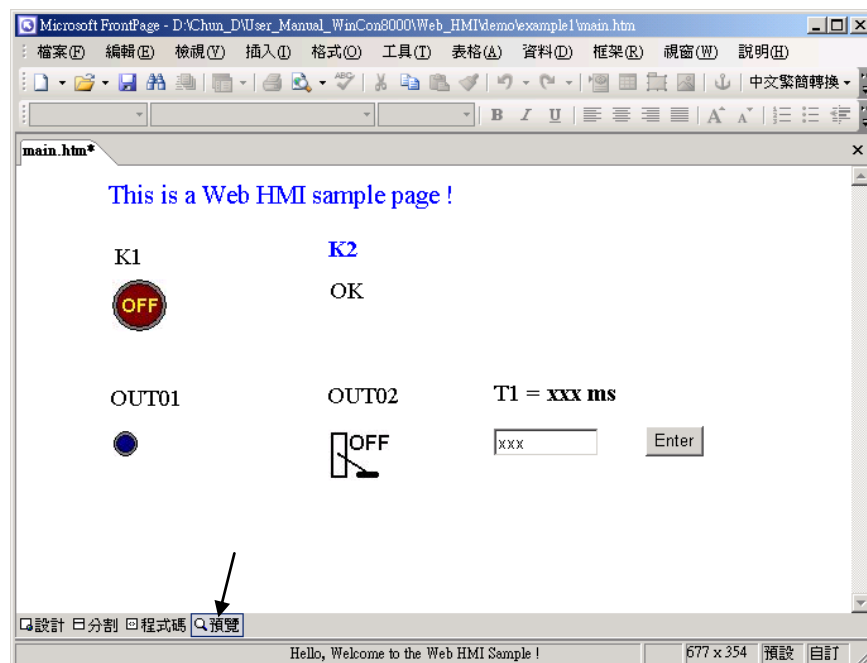
  T1.innerText=timer_val[21] + " ms";
}

```

Now we have finished all the code. Please save it.



You may click on "Preview" to simulate its run time behavior.



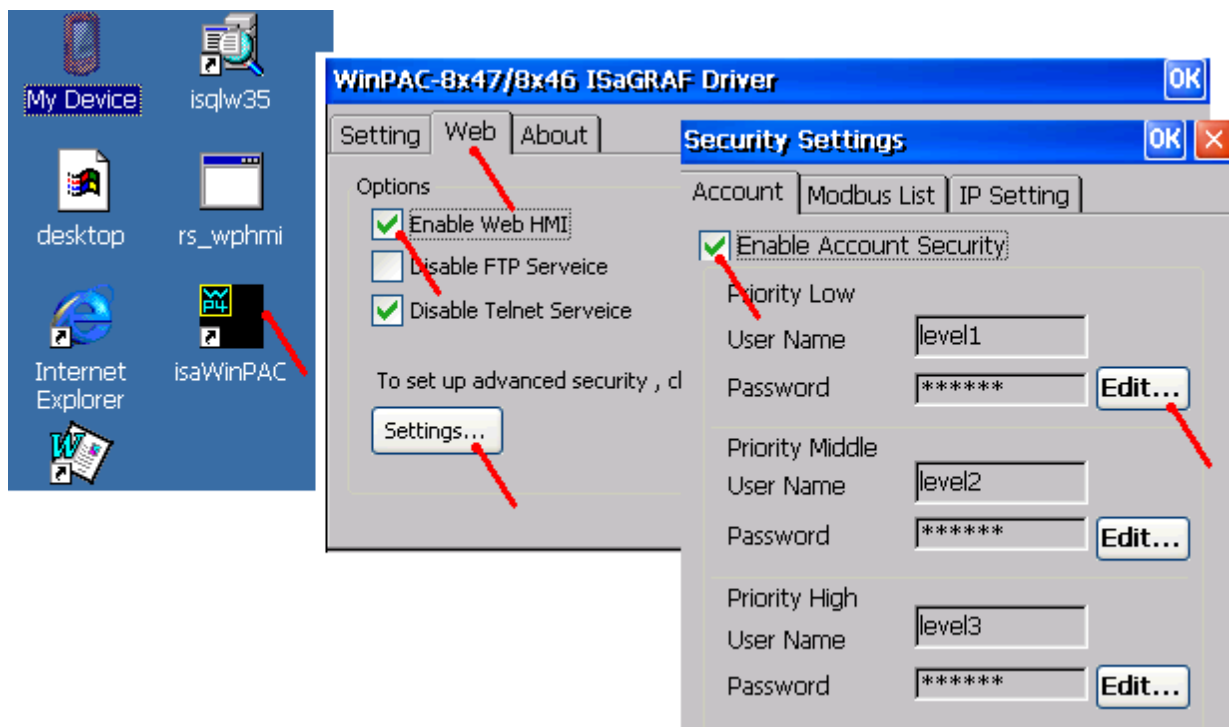
4.4.4: Step 4 – Download Web HMI Pages To The Controller

The steps are similar as listed in Section 3.2. If you haven't practiced "Setting Up A Web HMI Demo" listed in the Chapter 3, it's better to do it once to get familiar with it.

First set the web options.

Check on "Enable Web HMI" and then click on "Setting", Please check on "Enable Account Security" and then click on "Edit" to set (username , password). **Then remember to click on "OK"**

Note: If "Enable Account Security" is not check, any user can easily get access to your WinPAC-8xx7 through the Internet Explorer.



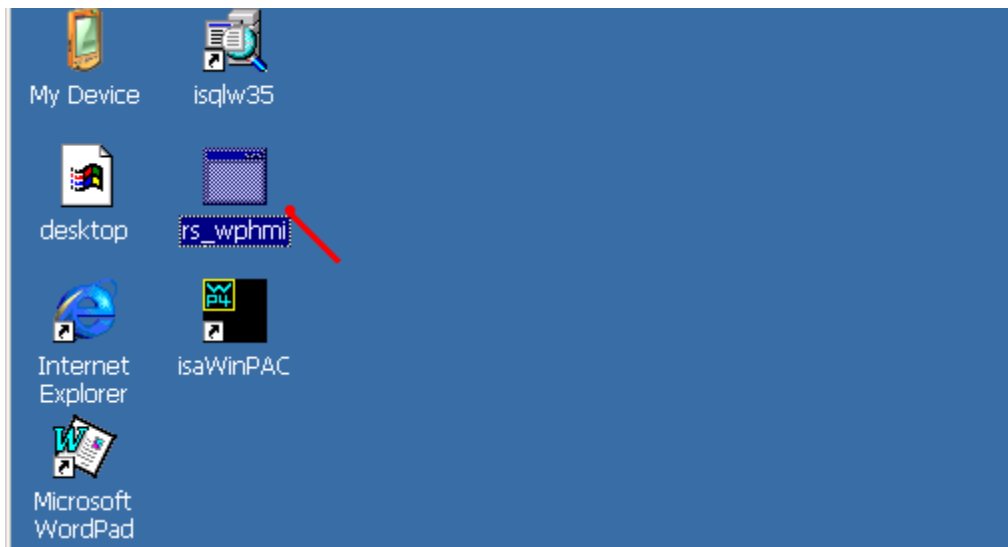
And then, please copy all files in this example1 to the controller

<your hard drive>:\example1\ *.*

to the WinPAC-8xx7's

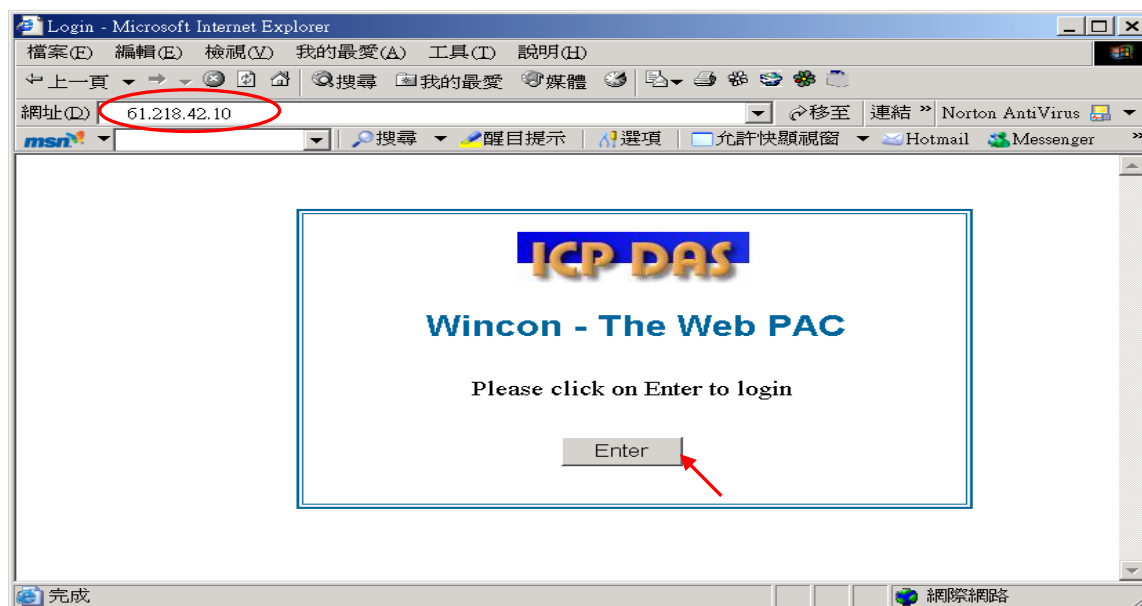
Micro_SD\Temp\HTTP\WebHMI\

Since the Web Pages are modified or new copied, please run "rs_wphmi.exe" to reset the Web server. **The "rs_wphmi.exe" must be run every time when user has modified any file in the WP-8xx7's \Micro_SD\Temp\HTTP\WebHMI**

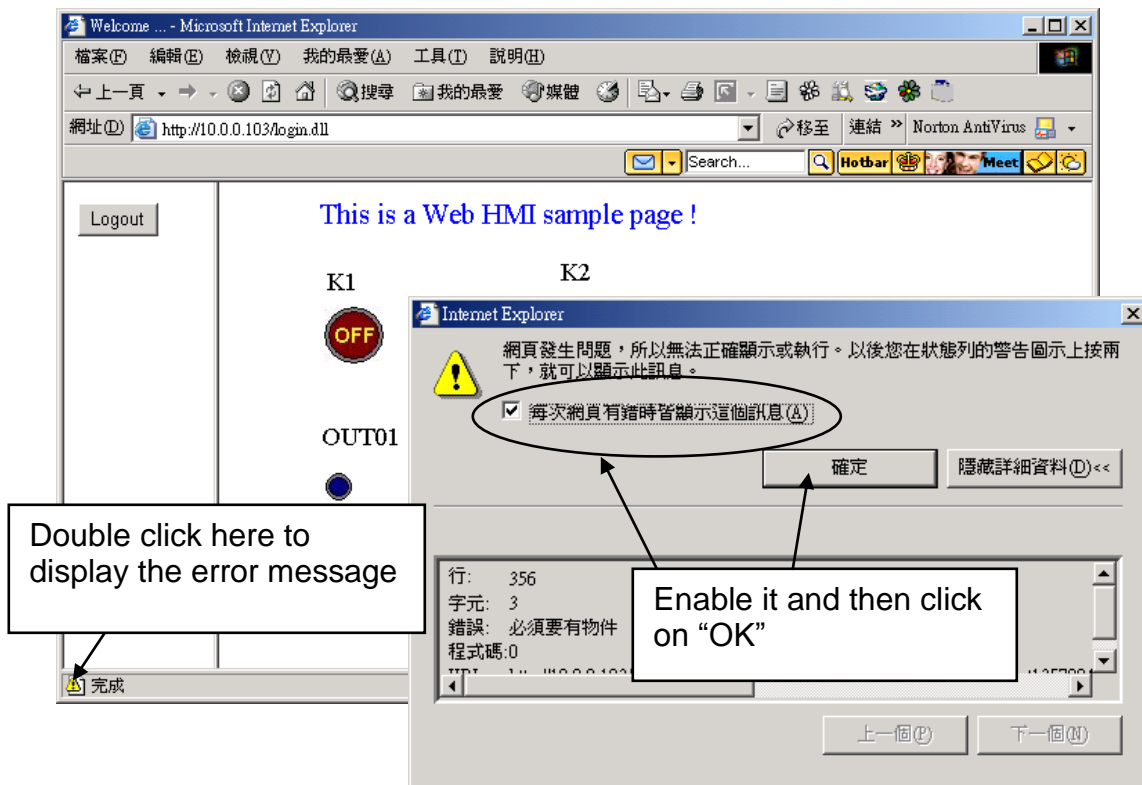


Show Time:

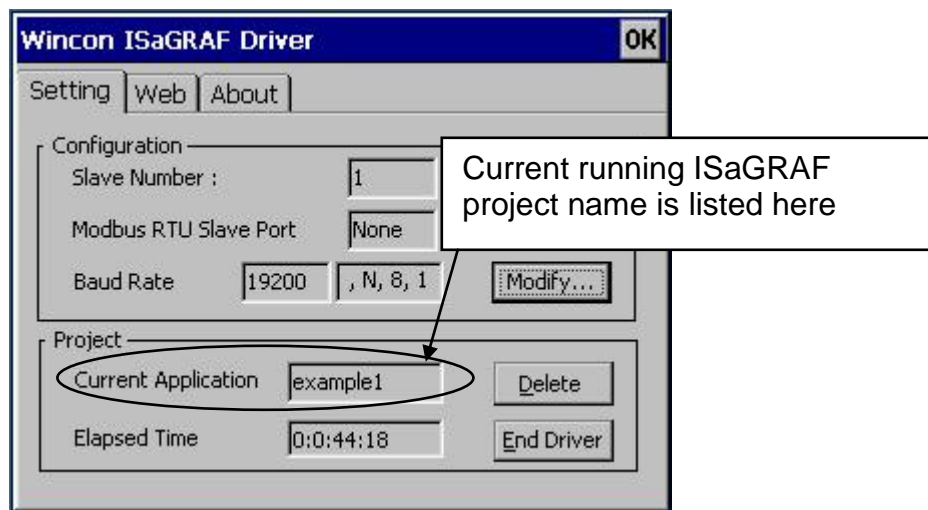
Please run Internet Explorer (Rev. 6.0 or higher), key in the IP address of your WP-8xx7.
For example: 61.218.42.10 or http://61.218.42.10



If there is something wrong with the web page. You may enable the below item to display the debug message every time it has error.



And also check if your ISaGRAF project already download to the controller (Section 4.3 or section 3.2.3).



And do you assign the correct Modbus Network address to the respective ISaGRAF variables ? (Section 4.1.5).

Chapter 5 : Web HMI Basics

The WinPAC-8xx7 (or WP-8xx7) is the abbreviation of the WinPAC-8147 / 8447 / 8847.

Important:

1. **WP-8xx7 / 8xx6 supports only High profile I-8K and I-87K I/O cards in its slot 0 to 7. Refer to WP-8xx7 CD: \napdos\isagraf\wp-8xx7\english_manu\ wp-8xx7_datasheet.pdf**
2. Please always set a fixed IP address to the WinPAC-8xx7. (No DHCP)

Note:

1. This chapter describes the programming basics for the Web HMI. We will not focus on the HTML basics. If you want to know more about the HTML programming, the best way is to “buy a HTML related book” from the bookstore. There are a lot of books doing this job.
2. The Web HMI only supports the basic HTML tags. It doesn't support ASP, PHP or JSP or other Page Server language.
3. Please do not use `<frameset>` `</frameset>` , `<frame>` `</frame>` in the Web HMI.
4. Note: The object name, object ID, code, variable name and function name is case sensitive. For example, `refresh_data()` and `Refresh_data()` is different.
5. There are more than ten Web HMI examples in the WinPAC-8xx7's CD-ROM. Please refer to section 3.1.

5.1: Basic Files For The Web HMI

The basic Web HMI files include 2 folders and 3 DLL files and 4 htm files as below.

./img/	(default image files - *.jpg , *.bmp , *.gif)
./msg/	(default message files – wincon.js & xxerror.htm)
whmi_filter.dll	(three DLL files)
login.dll	
main.dll	
index.htm	(first default page)
login.htm	(the Web HMI welcome page)
menu.htm	(the page-menu page, normally on the left on the Internet Explorer)
main.htm	(first page when successfully login)

User may put his own image files into the folder named as “user_img”. And put user-defined javascript file or css file into the folder named as “user_msg”. Other folder name is not acceptable by the Wincon Web HMI.

The “index.htm” file is the default entry page of the web server. User must not modify it. The “index.htm” re-directs to the “login.htm” file in 1 to 2 seconds when someone visits the WinPAC-8xx7 via the Internet Explorer.

User may modify the “login.htm” , “menu.htm” and “main.htm” to fit the requirement.

5.2: Login.htm

Login.htm is the first welcome page when a user visiting in. It can be modified. Below is the basic code for the login.htm

```
<html>
<head>

<title>Login</title>

<meta http-equiv=pragma content=no-cache>

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" >

<script language="JavaScript">
var random_val=123;
function get_random_val()
{
    var rightNow = new Date();
    random_val += 323456789*rightNow.getMinutes() +
                107654321*(rightNow.getTime()%1000);
    setTimeout("get_random_val()", 197); // repeat call
}

//check if username and password are empty
function validate(fm)
{
    setKey(fm);
    return true;
}

//Embed key while submitting
function setKey(fm)
{
    var rightNow = new Date();
    cookieVal = random_val+rightNow.getTime();
    fm.key_.value = cookieVal;
}
</script>
</head>

<body onload="get_random_val()">

<div style="position: absolute; width: 332px; height: 34px; z-index: 5; left: 147px; top: 27px"
id="layer1">
Welcome !</div>

<div style="position:absolute; width:122px; height:38px; z-index:4; left: 171px; top: 95px;"
id="layer2">
```

This line is only for the "Login.htm", please do not apply to other pages, for example, the "menu.htm" & "main.htm" & other .htm pages.

Please apply your charset here.
For example,
English: UTF-8
Chinese: gb2312
Traditional Chinese: big5
or other language

get_random_val () should be always called at the beginning of the Login.htm . It is the entry point of the Login.htm

Your caption is here.


```

<form name="form1" action="/login.dll" method="post">
  <input type="hidden" name="key_">
  <input type="submit" name="Submit" value=" Enter " style="cursor:hand" onClick="return
validate(this.form)">
</form>

```

"form1" is necessary

You may modify " Enter " to your own word. For example "請進". This may require user to modify the related "charset" at the beginning of this page.

</div>

</body>

<!-- To ensure no-cache work -->

<head>

<meta http-equiv=pragma content=no-cache>

</head>

This code is only for the "Login.htm", please do not apply to other pages, for example, the "menu.htm" & "main.htm" & other .htm pages.

</html>

That's all the login.htm need. You can insert more images or text to it. Only remember to keep its basic code.

5.3: Menu.htm

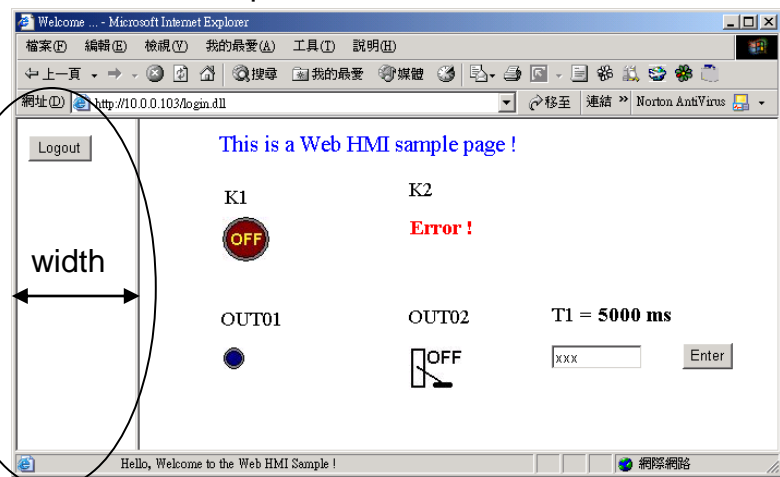
Note:

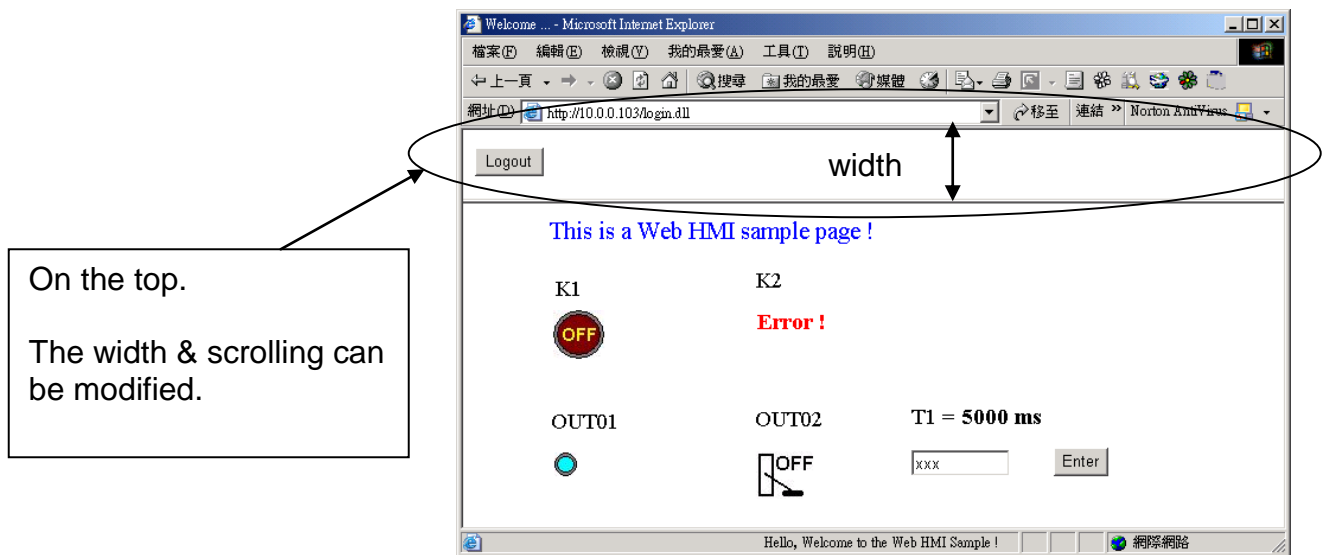
If you want to know more about the multi-page application, there are two demos in the WinPAC-8xx7 CD-ROM: \napdos\isagraf\wp-8xx7\wp_webhmi_demo\wphmi_05 & wphmi_05a. The "wphmi_05" place its page-menu on the left, while "wphmi_05a" on the top.

The "Menu.htm" defines the Page-menu of the Web HMI especially for the multi-page application. The page-menu can place only on the left or on the top.

On the left.

The width & scrolling can be modified.





Below is the basic code for the menu.htm

```
<!-- top_or_left=1 , scrolling=0 , width=60 , resize=1 -->
```

The first row is not a comment, it defines the Page-Menu behavior
top_or_left: 1:Top , 0:Left
scrolling: 1:Yes , 0:No
width: width of the Menu Frame, 0 – 999 (unit is pixel)
resize 1:Yes , 0:No

```
<html>
<head>
<title>Title1</title>
```

```
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
```

```
<SCRIPT LANGUAGE="JavaScript" src="./msg/wincon.js"></SCRIPT>
```

```
<SCRIPT LANGUAGE="JavaScript">
```

```
function start1()
{
  A_11();
}
function refresh_data()
{
  if(run_at_pc==1) return;
}
</SCRIPT>
```

This row is necessary
for menu.htm ,
main.htm & other
multi-pages

Please apply your charset here.
For example,
English: UTF-8
Chinese: gb2312
Traditional Chinese: big5
or other language

```
</head>
```

```
<body onload="start1()">
```

start1 () is the entry point of the menu.htm

```
<!-- Logout button -->
```

```

<form name="form_logout" method="post" action="./login.dll">
  <input style="cursor:hand" name="CMD" type="submit" value="Logout" onClick="return
logout(this.form)">
</form>

</body>
</html>

```

form_logout is for the logout button.

Note:

If you want to know more about the multi-page application, there are two demos in the WinPAC-8xx7 CD-ROM: \napdos\isagraf\wp-8xx7\wp_webhmi_demo\wphmi_05 & wphmi_05a . The “wphmi_05” place its page-menu on the left, while “wphmi_05a” on the top.

5.4: Main.htm

5.4.1: A Simple Main.htm Example

Before going further in the main.htm, first take a look at a simple main.htm example. This example only display a “Hello !” message when successfully login, nothing else.

```

<html>
<head>
<title>Title1</title>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" >

<SCRIPT LANGUAGE="JavaScript" src="./msg/wincon.js"></SCRIPT>

<SCRIPT LANGUAGE="JavaScript">
show_scroll_word(200,"Hello, Welcome to the Web HMI Sample !");

function refresh_data()
{
}

</SCRIPT>
</head>

<body onLoad="init()">

<font color="blue" size="4">
<div style="position: absolute; width: 353px; height: 24px; z-index: 1; left: 73px; top: 12px"
id="layer1"> Hello !</div>
</font>

</body>
</html>

```

Please apply your charset here. For example, English: UTF-8
Chinese: gb2312, Traditional Chinese: big5 , or other language

This line is necessary for menu.htm , main.htm & other multi-pages

Calling show_scroll_world() will display a moving word at the bottom of the Internet Explorer. Here 200 means 200 ms. You may make it slower, for example, using 500.

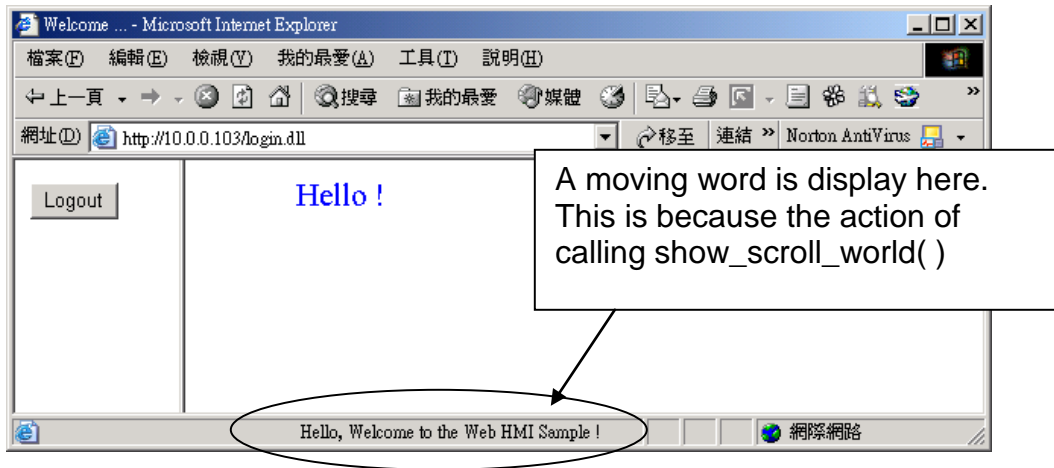
refresh_data() is called when the Internet Explorer has received the requested data from the controller. It is called in the period about 1.25 to 5 seconds depends on the communication quality.

init() is the entry pint of the main.htm & other multi-pages.

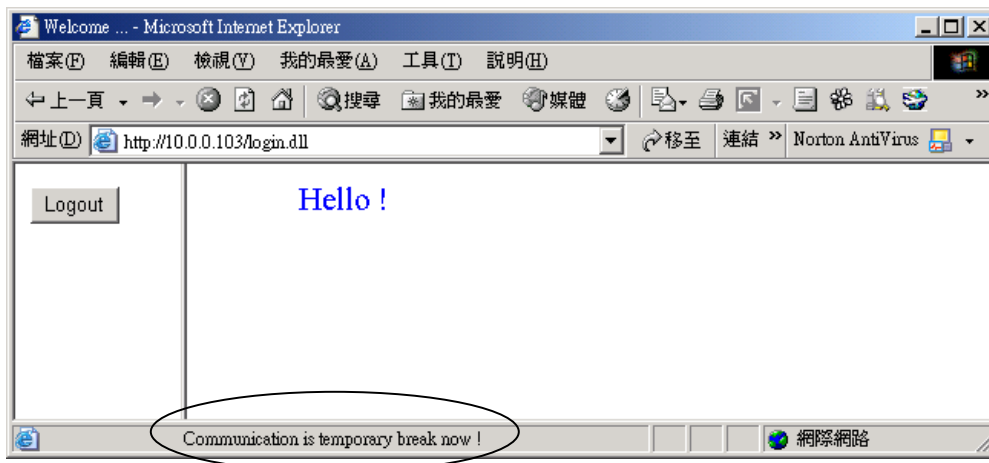
A layout object is starting with “<div” & ending at “</div>” tags. Here only show a message “Hello !”

You may replace the main.htm in the WinPAC-8xx7 CD-ROM:
\napdos\isagraf\wp-8xx7\wp_webhmi_demo\sample

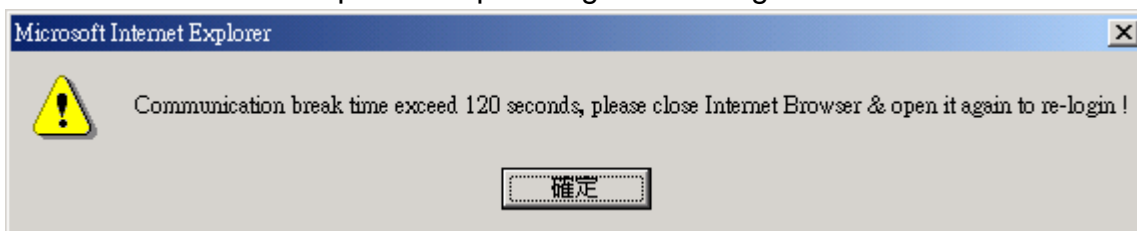
to the above main.htm & download it to the controller (refer to section 4.4.4). You will see the below window when you login successfully.



User may try to plug out the Ethernet cable of the WinPAC or of your PC. You will see it show “Communication is temporary break now !” in about 10 seconds. When you plug the cable back, the communication will be recovered in about 10 to 45 seconds.



If the communication broken time exceeds 120 seconds, it will show the below message. You have to close the Internet Explorer & open it again to re-login.



5.4.2: More About The refresh_data() Function And Dynamic Data

Note: The code, variable name and function name is case sensitive. For example, refresh_data() is correct, however Refresh_data() is not correct.

The refresh_data() function must always apply in the main.htm and other multi-pages. It is called when the Internet Explorer has received the requested data from the controller. The calling period is about 1.25 to 5 seconds depends on the communication quality

The refresh_data() is often used for refreshing the dynamic data. For example, the boolean value , integer value, timer value or float value of the variables in the ISaGRAF project.

The Internet Explorer can access to the data in the ISaGRAF project only when they are assigned a unique Modbus Network Address No (refer to section 4.1.5). The Web HMI only accepts Network Address No in the range of 1 to 1024. The data without a Network Address No (No. = 0) or not in the range of (1 to 1024) is not accessible by the Internet Explorer.

The main.htm and other multi-pages can use the below variable array to access to the ISaGRAF's data (case sensitive). The identifier appeared in the [] is the related Network Address No. For example boolean_val[2] means the boolean value of the ISaGRAF boolean data which is assigned with the Network Address No. = 2.

boolean_val	boolean value in the ISaGRAF
word_val	word value in the ISaGRAF, -32768 to +32767
float_val	real value in the ISaGRAF, for ex, 1.234 , -0.456E-02
timer_val	timer value in the ISaGRAF, unit is ms, max = 86399999 (< 1 day)
string_val	message value in the ISaGRAF, max string length is 255

To access to long integer value (32-bit integer) please use get_long_val() function. For example, get_long_val(11) , get_long_val(13) , get_long_val(15).

get_long_val()	long integer value in the ISaGRAF, -2147483648 to +2147483647
-----------------	---

Note:

The long integer, timer and float variable's Network Address No. must occupy 2 No. in the ISaGRAF project (refer to section 4.2 of "User's Manual of ISaGRAF Embedded Controllers" or in the CD-ROM:\napdos\isagraf\wp-8xx7\english_manu\ "User_Manual_I_8xx7.pdf").

That means if you assign a Network Address No.= 11 to a Real type variable(or Timer or integer will have 32-bit value – larger than 32767 or smaller than -32768), the next No. 12 should not assigned to any other variable in the ISaGRAF project. However you may assign No.=13 to one another variable.

5.4.2.1: Displaying Dynamic Boolean Data

Demo example: whmi_02 and whmi_05 (section 3.1)

Let's look back to the refresh_data function. If user want to display the dynamic boolean value, the below code can be used.

```
...
function refresh_data()
{
  B1.src = "./img/circle_blue" + boolean_val[1] + ".jpg" ;
}
...
<body onLoad="init()">
...
<div style="position: absolute; width: 214px; height: 53px; z-index: 2; left: 102px; top: 79px">
</div>
...
</body>
```

The action of the image object "B1" is defined here.

if boolean_val[1]=1, it display image "B1" as "img/circle_blue1.jpg"
if boolean_val[1]=0, it display image "B1" as "img/circle_blue0.jpg"

The layout (or location) of the image object "B1" is defined here by the "<div" and "</div>" tags.

The declaration of image "B1" is defined here by the "img" tag & name="B1" src= ... ← "src=" defines the initial value of B1

5.4.2.2: Displaying Dynamic Float & Word & Timer Data

Demo example: wphmi_01 , wphmi_03 and wphmi_05 (section 3.1)

If user want to display the dynamic float value, the below code can be used.

```
...
function refresh_data()
{
  F21.innerText = float_val[21] ;
}
...
<body onLoad="init()">
...
<div style="position: absolute; width: 214px; height: 53px; z-index: 2; left: 102px; top: 79px">
<b id="F21"> xxxx </b> </div>
...
</body>
```

The action of the Text object "F21" is defined here.

If want to display Word data, please use "word_val[]"
If want to display Timer data, please use "timer_val[]".
For ex, F21.innerText = timer_val[21] + " ms";

The layout (or location) of the Text object "F21" is defined here by the "<div" "</div>" tags.

The declaration of Text object "F21" is defined here by the "<b" tag & id="F21" & "" tag initial value of this F21 is "xxxx"

5.4.2.3: Displaying Dynamic Long Integer Data

Demo example: wphmi_03 and wphmi_05 (section 3.1)

If user want to display the dynamic long integer value (32-bit format), the below code can be used.

```
function refresh_data()
```

```
{  
  L11.innerText = get_long_val(11) ;  
}
```

The action of the Text object "L11" is defined here.

...

```
<body onLoad="init()">
```

...

The layout (or location) of the Text object "L11" is defined here by the "<div" and "</div>" tags.

```
<div style="position: absolute; width: 214px; height: 53px; z-index: 2; left: 102px; top: 79px">
```

```
<b id="L11"> xxx </b> </div>
```

...

```
</body>
```

The declaration of Text object "L11" is defined here by the "<b" tag and id="L21" and "" tag , the initial value of this L11 is "xxx"

5.4.2.4: Displaying Dynamic String Data

If user want to display the dynamic string value (max length is 255), the below code can be used.

```
...  
function refresh_data()
```

```
{  
  S31.innerText = string_val[31] ;  
}
```

The action of the Text object "S31" is defined here.

...

```
<body onLoad="init()">
```

...

The layout (or location) of the Text object "S31" is defined here by the "<div" and "</div>" tags.

```
<div style="position: absolute; width: 214px; height: 53px; z-index: 2; left: 102px; top: 79px">
```

```
<b id="S31"> empty </b> </div>
```

...

```
</body>
```

The declaration of Text object "S31" is defined here by the "<b" tag and id="S31" and "" tag , the initial value of this S31 is "empty"

5.4.2.5: Trigger A Boolean Object To Blink

Demo example: wphmi_02 and wphmi_05 (section 3.1)

Some application may need a message to blink when the boolean value changes. For example, If `boolean_val[12]` is False, it means "OK". However if `boolean_val[12]` is True, it means "Error !". User may want to make this "Error !" blink to attract viewer's attention.

The below code can do this job.

```
...
var blink_period=500;
setTimeout("blink_obj()", blink_period);
var B12_blink=0; // init as 0:not blink
var blink_step=0;

function blink_obj()
{
  if(blink_step==1)
  {
    blink_step=0;

    if(B12_blink==1)
    {
      B12.innerText="Error !" ;
      font_B12.color="red";
    }

  }
  else
  {
    blink_step=1;

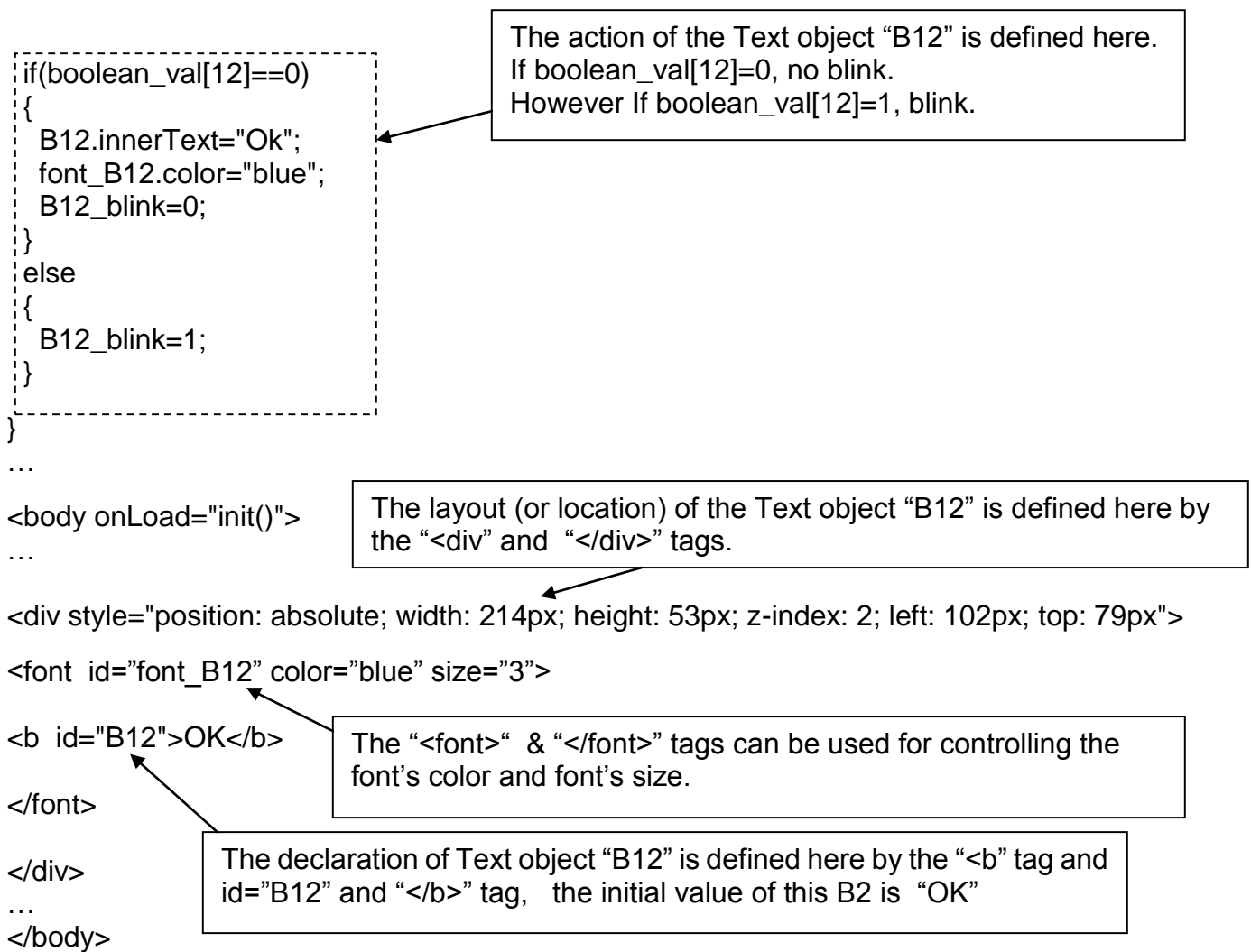
    if(B12_blink==1)
    {
      B12.innerText="" ;
      font_B12.color="red";
    }

  }
  setTimeout("blink_obj()", blink_period);
}

...function refresh_data()
{
```

The diagram illustrates the logic of the `blink_obj()` function with the following annotations:

- The blinking period, unit is ms**: Points to the `blink_period=500` variable.
- Setup a timer to handle the blinking action**: Points to the `setTimeout("blink_obj()", blink_period);` call.
- 1: to blink , 0: no blink**: Points to the `blink_step` variable.
- Blink step 1: To display "Error !" in red color.**: Points to the first `if(B12_blink==1)` block.
- Blink step 2: To display "" (nothing) in red color.**: Points to the second `if(B12_blink==1)` block.



5.4.2.6: Displaying Float Value With Fixed Digit Number Behind The "." Symbol

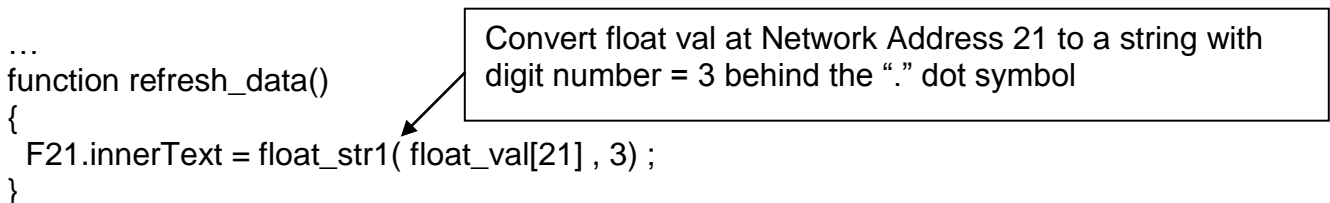
Demo example: wphmi_06 and wphmi_07 (section 3.1)

The `float_str1(para1 , para2)` function can convert float value to a string with fixed digit number behind the dot "." symbol

para1 is the float value to be converted, for ex, 1.234567

para2 is the digit number behind the "." dot symbol, 0 to 6

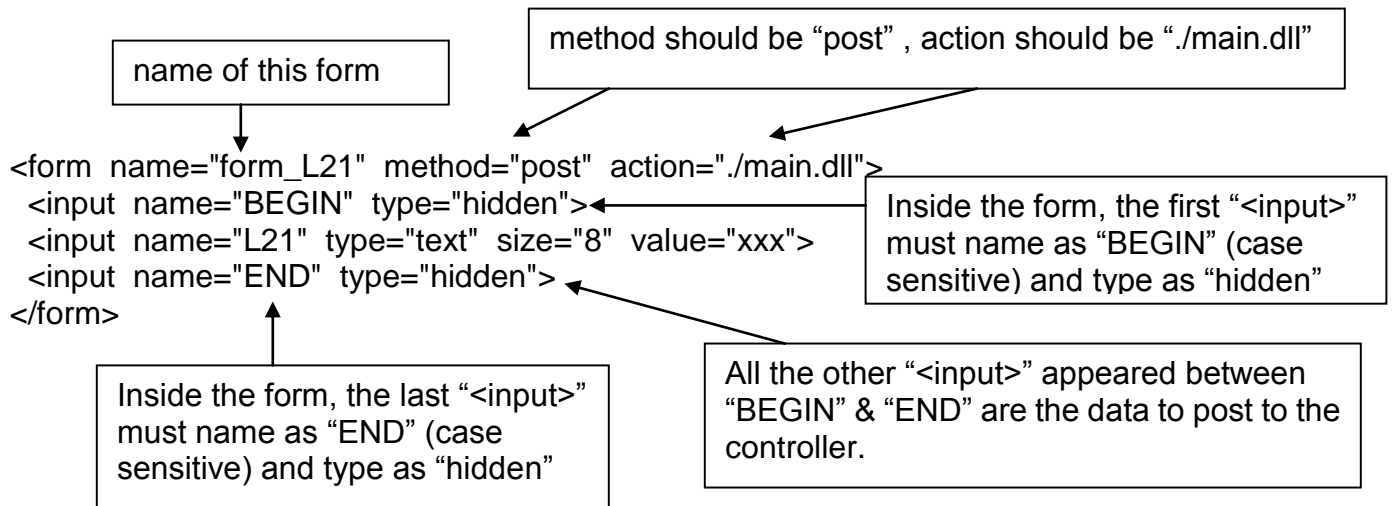
for ex, `float_str1(1.234567, 3)` return "1.234", while `float_str1(1.234567, 2)` return "1.23"



5.4.3: Post Data To The Controller

The former section 5.4.2 listing how to get and display data from the controller. This section focuses on posting data to the controller, in other word to control the WinPAC via the Internet Explorer.

To set a new value to the boolean, word, long integer, float , timer and string variables in the ISaGRAF project, we need “form” object appeared in the main.htm or other multi-pages. A “form” object looks like as below.



The “<input>” name to control the WinPAC’s data must follow below format. The number followed behind the first letter should be in the range from 1 to 1024. This number is point to the variable name in the ISaGRAF project with the same Modbus Network Address No.

B	point to the ISaGRAF boolean data , for ex, B5 , B109
W	point to the ISaGRAF word data (-32768 to +32767), for ex, W9 , W1001
L	point to the ISaGRAF long integer data (-2147483648 to +2147483647), for ex, L21 This “L” Also point to the ISaGRAF timer data
F	point to the ISaGRAF real data, for ex, F13 , F235
S	point to the ISaGRAF message data , for ex, S18

Note:

The long integer, timer and float variable’s Network Address No. must occupy 2 No. in the ISaGRAF project (refer to section 4.2 of “User’s Manual of ISaGRAF Embedded Controllers” or in the CD-ROM:\napdos\isagraf\wp-8xx7\english_manu\ “User_Manual_I_8xx7.pdf”)

That means if you assign a Network Address No.= 11 to a Real type variable(or Timer or integer will have 32-bit value – larger than 32767 or smaller than -32768), the next No. 12 should not assigned to any other variable in the ISaGRAF project. However you may assign No.=13 to one another variable.

5.4.3.1: Post Boolean Value to The Controller

A. To post by the image

...

```
function ON_OFF(form_obj, obj, current_boo_value)
```

```
{
  if(current_boo_value==0)
  {
    flag = confirm("turn ON ?");
    if(flag) obj.value=1;
  }
  else
  {
    flag = confirm("turn OFF ?");
    if(flag) obj.value=0;
  }
  if(flag)
  {
    if(GetUserID(form_obj)==true) form_obj.submit();
  }
}
```

ON_OFF function is used for posting boolean value to the controller by refer to the current boolean value.

The first parameter is the name of the "form".
The second parameter is the "<input>" name inside the form.
The last parameter is the current boolean value.

Demo example: wphmi_02 and wphmi_05

```
}
else
{
  flag = confirm("turn OFF ?");
  if(flag) obj.value=0;
}
if(flag)
{
  if(GetUserID(form_obj)==true) form_obj.submit();
}
```

```
function refresh_data()
{
  B2.src = "img/cmd" + boolean_val[2] + ".jpg" ;
}
```

Display the current boolean image. In this example,
0: display "img/cmd0.jpg" , 1: "img/cmd1.jpg"

```
...
<body onLoad="init()">
...
<div style="position: absolute; width:100px;height:100px; z-index: 5; left: 242px; top: 164px" >
```

The layout (or location) of the image object "B2" is defined here by the "<div>" and "</div>" tags.

"cursor:hand" will display the mouse arrow as a hand when entering the image area

```

```

Name of the form

Name of the image object

```
<form name="form_B2" method="post" action="./main.dll">
  <input name="BEGIN" type="hidden">
  <input name="B2" type="hidden" value="0">
  <input name="END" type="hidden">
</form>
```

Name of "<input>" inside the form. Here is "B2". Because it is inside "form_B2", then we must use the name of "form_B2.B2" to identify it.

The onclick will call ON_OFF() when the mouse click on it.

The first parameter is the name of the "form". Here is "form_B2"

The second parameter is the "<input>" name inside the form. Here is "form_B2.B2"

The last is the current boolean value. Here is boolean_val[2]

```
</div>
```

...

```
</body>
```

B. To post by buttons

```
function ON_(form_obj, obj)
```

```
{
  flag = confirm("turn ON ?");
  if(flag)
  {
    obj.value=1;
    if(GetUserID(form_obj)==true) form_obj.submit();
  }
}
```

Demo example: wphmi_02 and wphmi_05

ON_ function is used for posting boolean value as "True" to the controller .

```
function OFF_(form_obj, obj)
```

```
{
  flag = confirm("turn OFF ?");
  if(flag)
  {
    obj.value=0;
    if(GetUserID(form_obj)==true) form_obj.submit();
  }
}
```

OFF_ function is used for posting boolean value as "False" to the controller .

```
function refresh_data()
```

```
{
  B2.src = "img/big_Tcircle_red" + boolean_val[2] + ".jpg" ;
}
```

Display the current boolean image. In this example, 0: "img/big_Tcircle_red0.jpg" , 1: "img/ big_Tcircle_red1.jpg"

```
...
```

```
<body onLoad="init()">
```

```
...
```

```
<div style="position: absolute; width: 56px; height:40px; z-index: 5; left: 82px; top: 69px" >
```

```

```

```
</div>
```

```
<div style="position:absolute; left:85px; top:124px; width:42px; height:27px;">
```

```
<input type="button" value="ON" style="cursor:hand" onClick="ON_(form_B2, form_B2.B2)">
```

```
<form name="form_B2" method="post" action="/main.dll">
```

```
  <input name="BEGIN" type="hidden" value="">
```

```
  <input name="B2" type="hidden" value="1">
```

```
  <input name="END" type="hidden" value="">
```

```
</form>
```

```
</div>
```

The layout (or location) of the image object "B2" is defined here by the "<div>" and "</div>" tags.

Name of "<input>" inside the form. Here is "B2". Because it is inside "form_B2", then must use the name of "form_B2.B2" to identify it.

A button to call ON_()
First parameter is the name of the form. Here is "form_B2"
The second is the name of the "<input>" inside the form. Here is "form_B2.B2"

```
<div style="position:absolute; left:85px; top:166px; width:47px; height:31px">
```

```
<input type="button" value="OFF" style="cursor:hand" onClick="OFF_(form_B2, form_B2.B2)">
```

```
</div>
```

```
...
```

```
</body>
```

A button to call OFF_()
First parameter is the name of the form. Here is "form_B2"
The second is the name of the "<input>" inside the form. Here is "form_B2.B2"

5.4.3.2: Post Word & Long & Float & Timer & String Value to The Controller

```
...  
function Check(form_obj)  
{  
  flag = confirm("Are you sure?");  
  if(flag)  
  {  
    if(GetUserID(form_obj)==false) { return false; }  
    form_obj.submit();  
    return true;  
  }  
  else  
  {  
    return false;  
  }  
}
```

Check() is used for posting any "form".

Demo example:
wphmi_03, wphmi_04, wphmi_05,
wphmi_06 and wphmi_07

```
function refresh_data()  
{  
  L15.innerText=get_long_val(15);  
  F17.innerText=float_val[17];  
}
```

Display dynamic value here.
If data is word , please use word_val[]
If data is timer, please use timer_val[]
If data is string, please use string_val[]

```
<body onLoad="init()">  
...  

```

The layout (or location) of the text object "L15" & "F17" are defined here by the "<div" "</div>" tags.

```
<div style="position: absolute; width: 195px; height: 25px; z-index: 2; left: 45px; top: 52px" >  
L15 = <b id="L15">xxxx</b></div>  
<div style="position: absolute; width: 196px; height: 29px; z-index: 3; left: 45px; top: 82px" >  
F17 = <b id="F17">xxxx</b></div>
```

```
<div style="position:absolute; left:47px; top:131px; width:204px; height:60px">  
  <form name="form1" method="post" action="./main.dll">  
    <input name="BEGIN" type="hidden" value="">  
    <input name="L15" type="text" value="Enter long val (L15)">  
    <input name="F17" type="text" value="Enter float val (F17)">  
    <input name="END" type="hidden" value="">  
  </form>  
</div>
```

text input L15 & F17 inside the "form1" if data is timer, please use "L" . And "W" for word. "S" for string

```
<div style="position:absolute; width:74px; height:31px; left: 234px; top: 150px;">  
  <input type="button" style="cursor:hand" onClick="return Check(form1)" value="Enter">  
</div>
```

"cursor:hand" will display the mouse arrow as a hand when entering the button area

When mouse click on this button, it calls Check() to post to the controller

```
...  
</body>
```

5.5: Multi-Pages

The Web HMI in the WinPAC-8xx7 supports multi-pages application. You may refer to Chapter 3 to setup the multi-page demo – “wphmi_05” to see how it work.

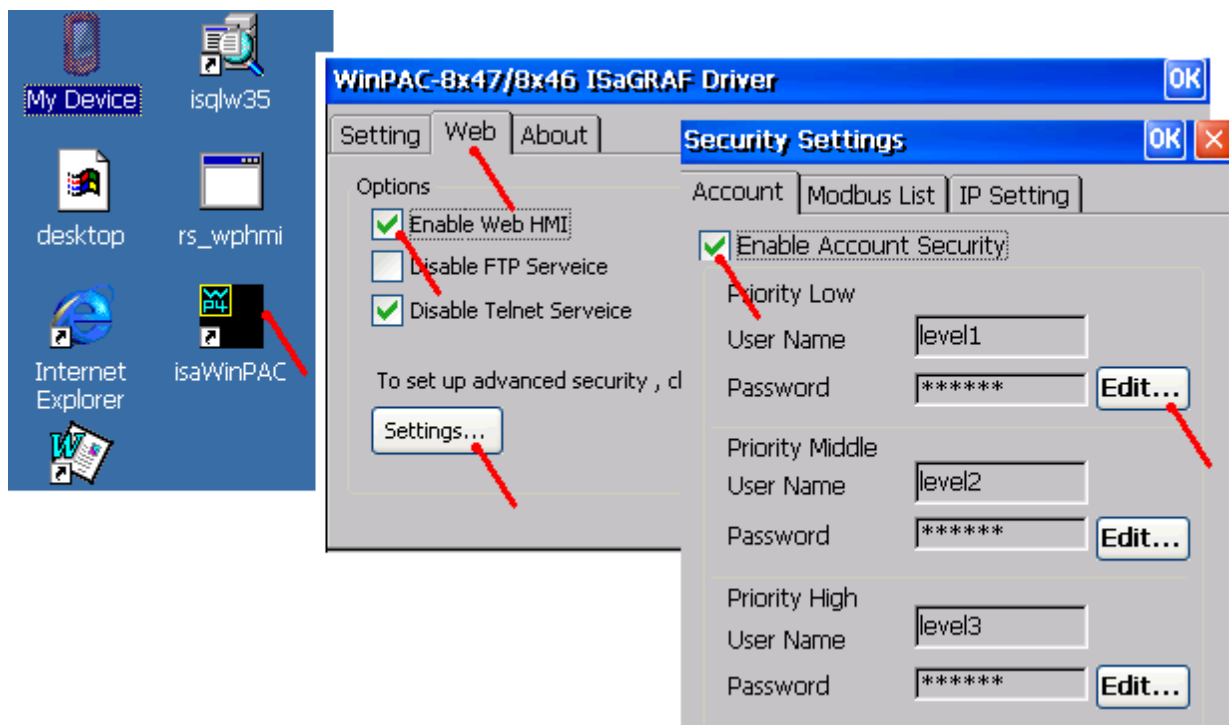
5.5.1: Level 2 And Level 3 Page

The multi-page name can be any valid html file name. For example, “page2.htm”, “kitchen.htm”, “u2-page4.htm” .

If “u2-“ appear in front of the page name, the page will become a Level 2 page. For example, the “u2-Page4.htm” in the “wphmi_05” demo. If “u3-“ appear in front of the page name, the page will become a Level 3 page. For example, the “u3-time.htm” in the “wphmi_05” demo.

What is a Level2 page ? Only users login with the Middle or High priority can get access to it. To access to the Level3 page, users have to login as a High priority user. The page name without “u2-“ and “u3-“ is identified as Level 1 page. That means any user successfully login can access to it. For example, the “main.htm”

The other rules for multi-pages are almost the same as “main.htm” (section 5.4)



Note: If “Enable Account Security” is not checked, any user can easily get access to your WinPAC through the Internet Explorer.

5.5.2: Switch One Page To One Another Page

Please take a look at the “menu.htm” of the “wphmi_05” demo as below. The “goto_R_page()” function can be used for switching to other page.

```
<!-- top_or_left=0 , scrolling=0 , width=110 , resize=1 -->

<html>
<head>
<title>Title1</title>
<meta http-equiv="Content-Type" content="text/html; charset=big5" >
<SCRIPT LANGUAGE="JavaScript" src="./msg/wincon.js"></SCRIPT>

<SCRIPT LANGUAGE="JavaScript">
function start1()
{
  A_11();
}
function refresh_data()
{
  if(run_at_pc==1) return; // if simulate at the PC, just return

  ...
}
</SCRIPT>
</head>
<body onload="start1()">

<!-- Logout button -->
<form name="form_logout" method="post" action="./login.dll">
  <input style="cursor:hand" name="CMD" type="submit" value="Logout" onClick="return
logout(this.form)">
</form>
<br/>
<br/>

<!-- Goto main.htm -->
<A style="cursor:hand" onClick="goto_R_page('main.htm')">第1頁</A>
<br/>
<br/>

<!-- Goto kitchen.htm -->
<A style="cursor:hand" onClick="goto_R_page('kitchen.htm')">Kitchen</A><br/>
<br/>
<br/>
```

“cursor:hand” will display the mouse arrow as a hand when entering the button area

Switch page to “main.htm”

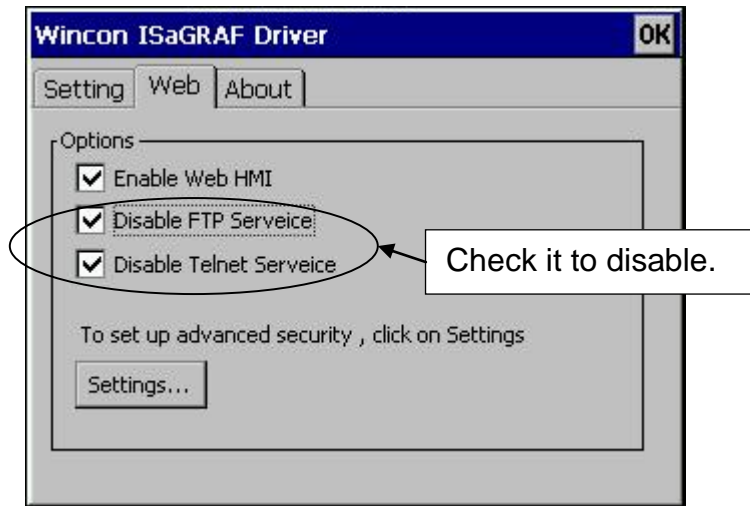
Switch page to “kitchen.htm”

5.6: Web Security

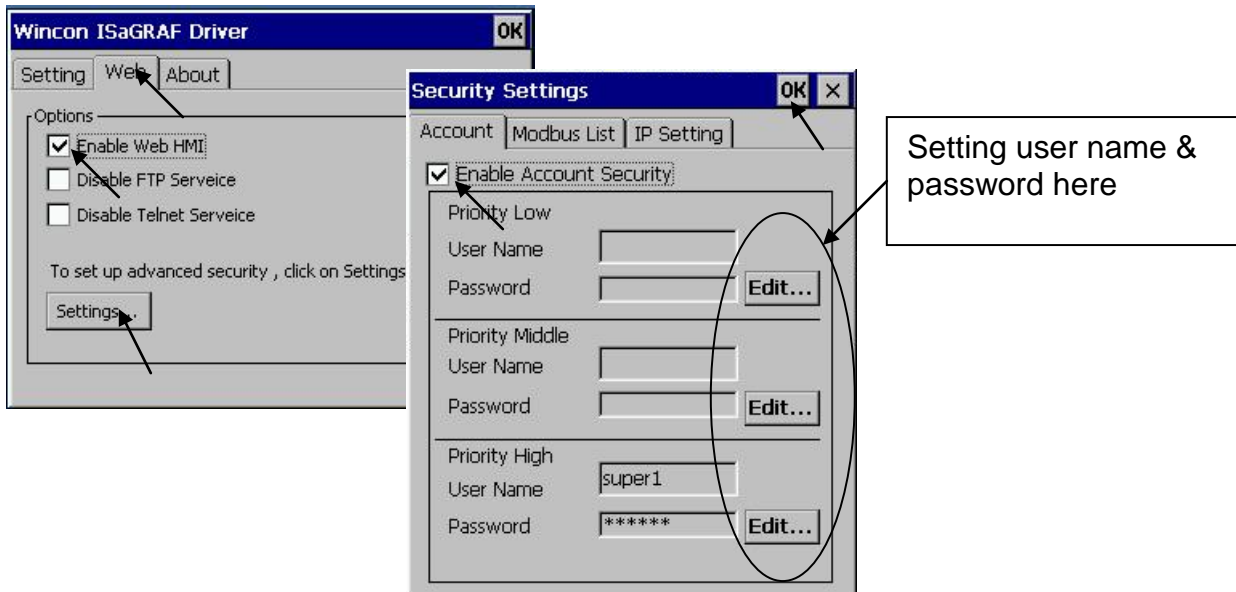
There are some ways user can get access to the WinPAC-8xx7 via its ethernet port.

1. Using Modbus TCP protocol at port No.= 502. (ISaGRAF and other HMI can do this)
2. Using ftp (for example, keyin “ftp://10.0.0.103” on the Internet Explorer)
3. Using telnet (for example, keyin “telnet 10.0.0.103 in the “command” window)
4. Using the Web server (The Web HMI does)

For safety, recommend to disable item 2 and 3 at run time.

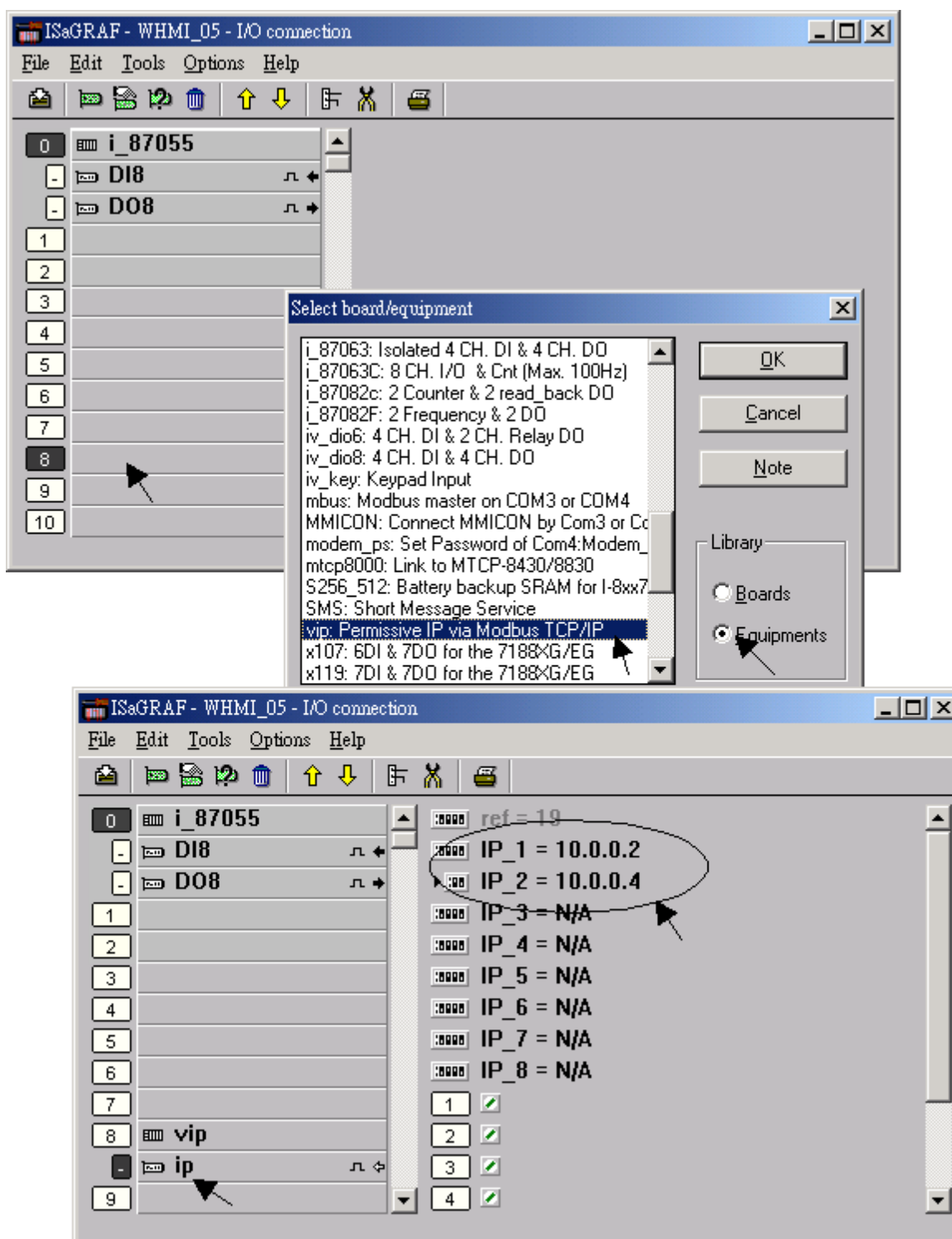


And about item 4, please set proper username & password for the Web HMI.



About item 1, user may set up to 8 IP address for ISaGRAF or other HMI to get access to the WP-8xx7 via the Modbus TCP/IP protocol as below.

On the IO connection window of ISaGRAF, please connect “vip” and entering the IP which can get access to the WP-8xx7 via Modbus TCP/IP protocol. If “vip” is not connected, any remote IP can get access to your WP-8xx7 via Modbus TCP/IP protocol. If “vip” is connected and No IP is entered (all assigned as “N/A”), No HMI and ISaGRAF can get access to it anymore.



Please re-compile your ISaGRAF project and download it to the controller if you have modified the IO connection.

Chapter 6 : VB.net 2008 Program Running In WinPAC-8xx7 Access To ISaGRAF Variables

This chapter lists the procedure for creating the first demo program by Visual Studio .NET 2008 development tool. There is some sample programs in the WinPAC-8xx7 CD-ROM.

WinPAC-8xx7 CD-ROM : \napdos\isagraf\wp-8xx7\vb.net_2008_demo\

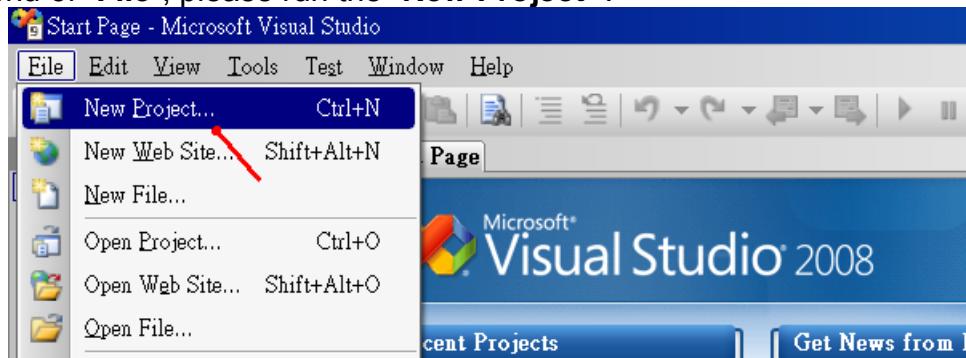
wp_vb01 : Digital I/O demo with one I-87055W in slot 0 of the WinPAC-8xx7.

wp_vb02 : Analog I/O demo with one I-87024W in slot 1 and one I-8017HW in slot 2.

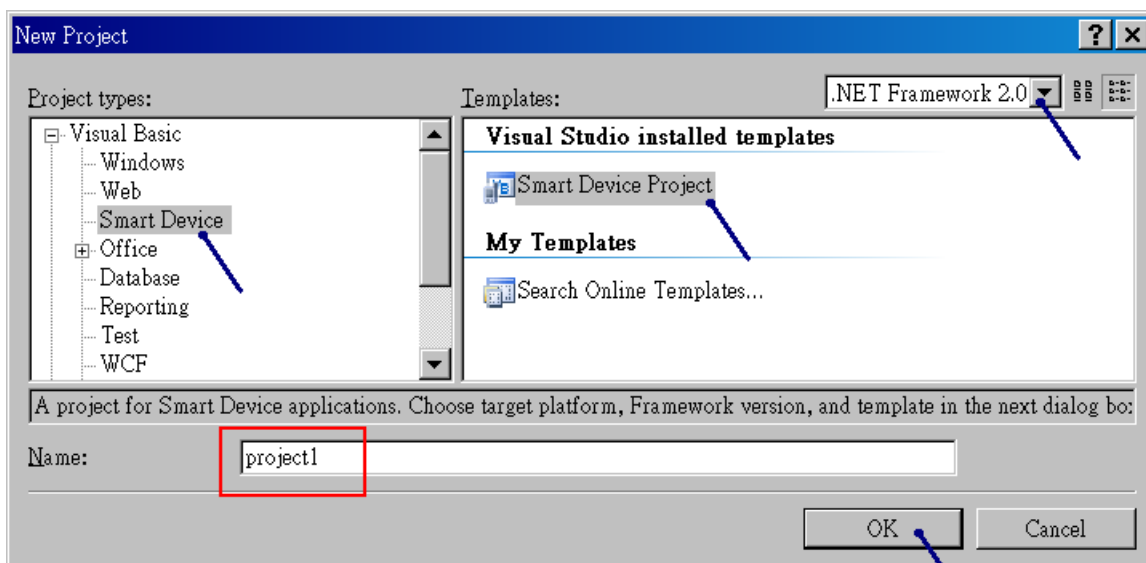
wp_vb03 : Read / Write ISaGRAF internal integers, timers and real variables. (No I/O)

6.1 Create a New Project

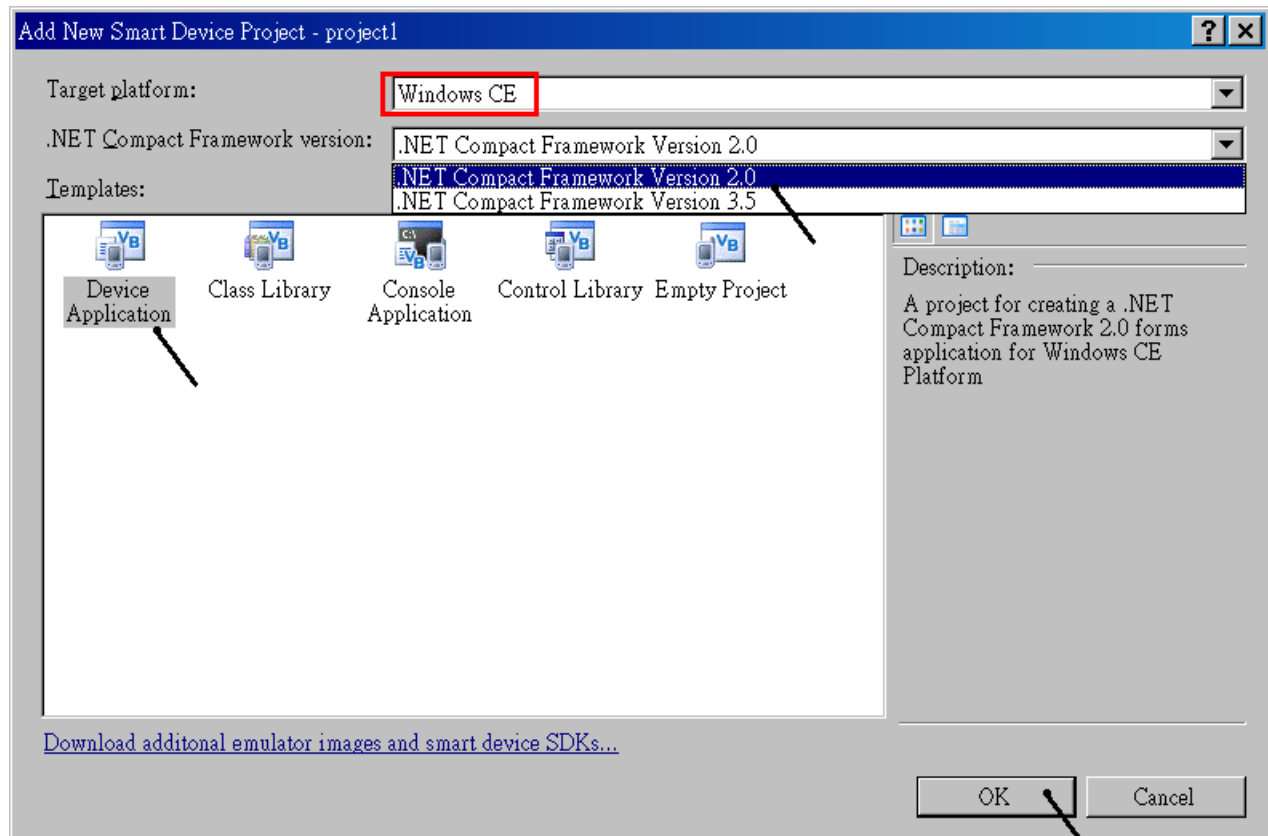
1. In the first, users need to open Microsoft Visual Studio .NET 2008 software. And then in the menu of “File”, please run the “New Project” .



2. Check the “Smart Device” on the left, then selecting the “.NET framework 2.0” and “Smart Device Project”. Then entering a proper project name and the last click on “OK” .



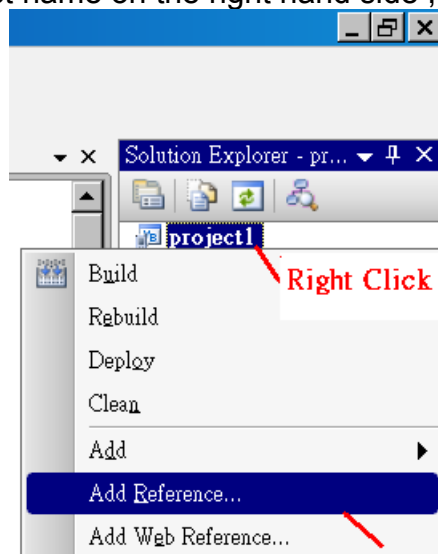
3. Select the "**Device Application**" and "**Windows CE**" and **".NET Compact Framework Version 2.0"** , then click on "OK" .



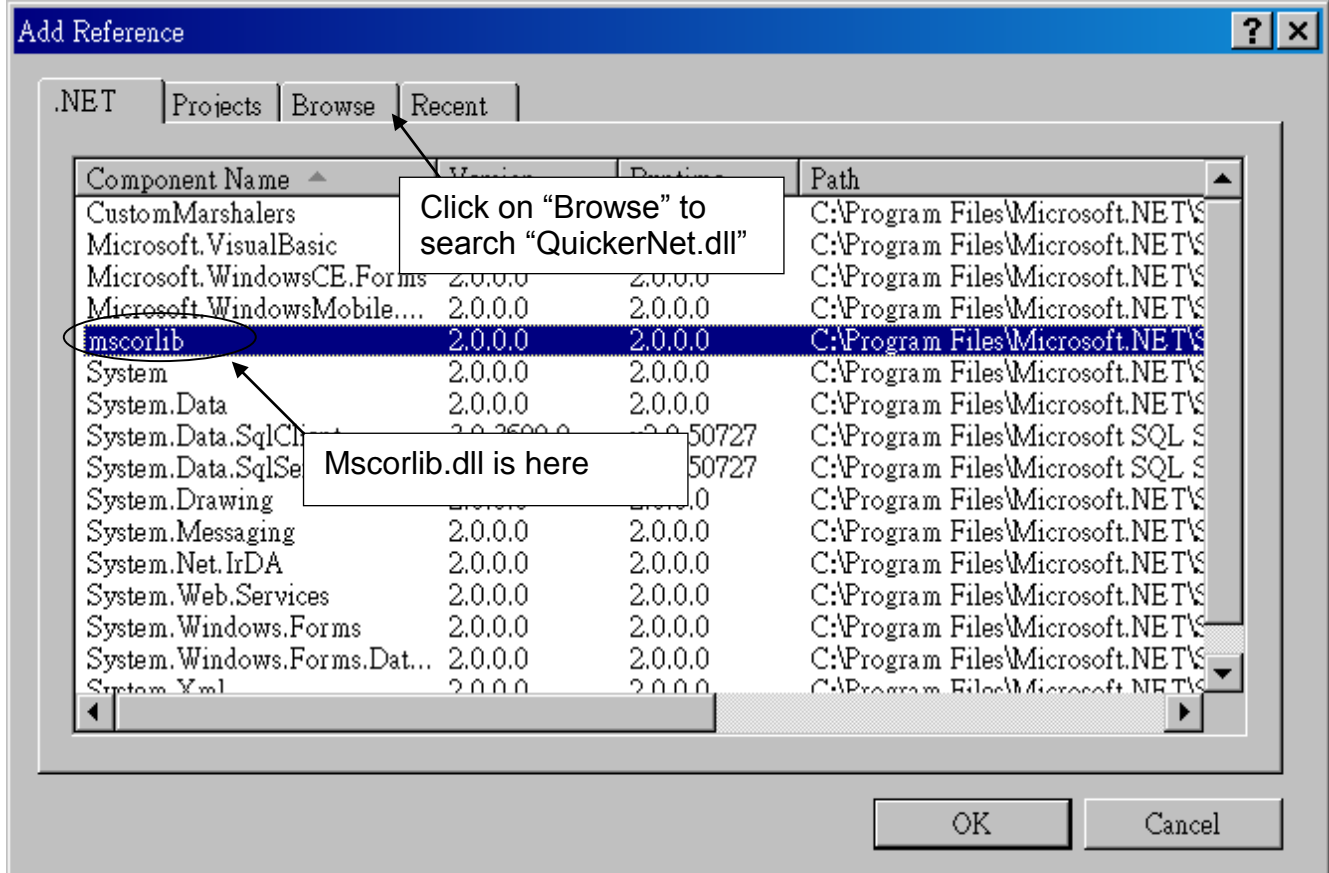
6.2 Add Project Reference for an Application

The "QuickerNet" library contains all modules' functions. Before you use the "Quicker" keyword in the program, you must add the "QuickerNet.dll" into the reference list of your application.

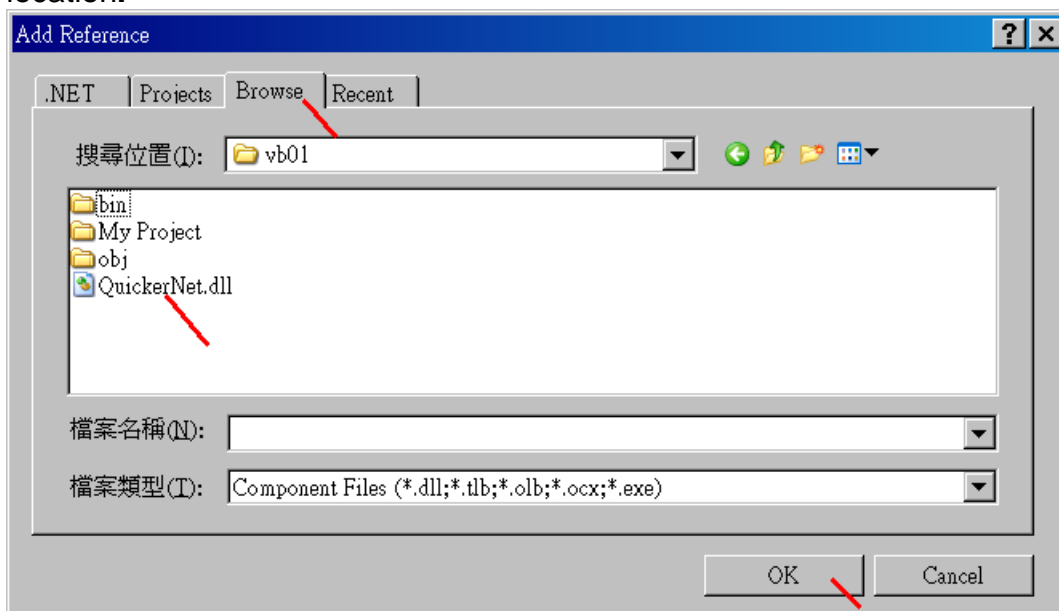
1. Right click on the Project name on the right hand side , then select "Add Reference ..."



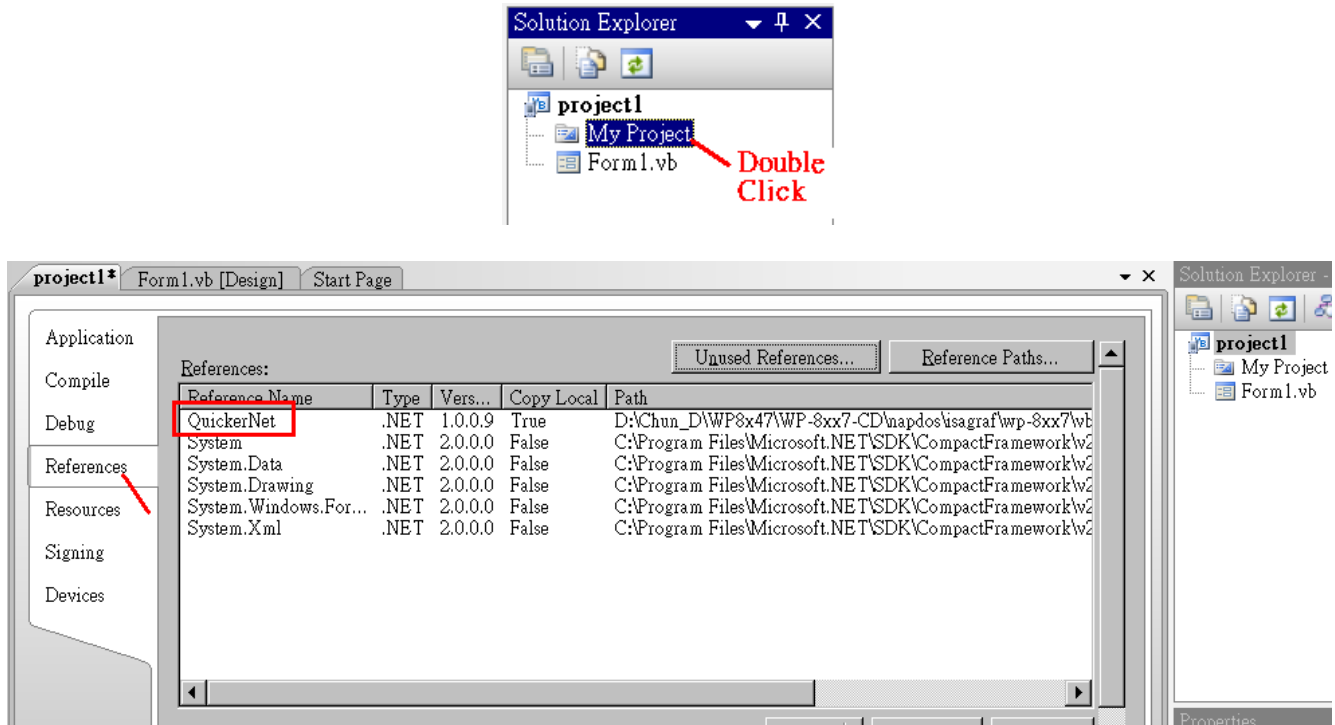
2. Select the “**mscorlib**” in the list box and click the button “**OK**” (the component “**mscorlib**” must appear in the Selected Components area)



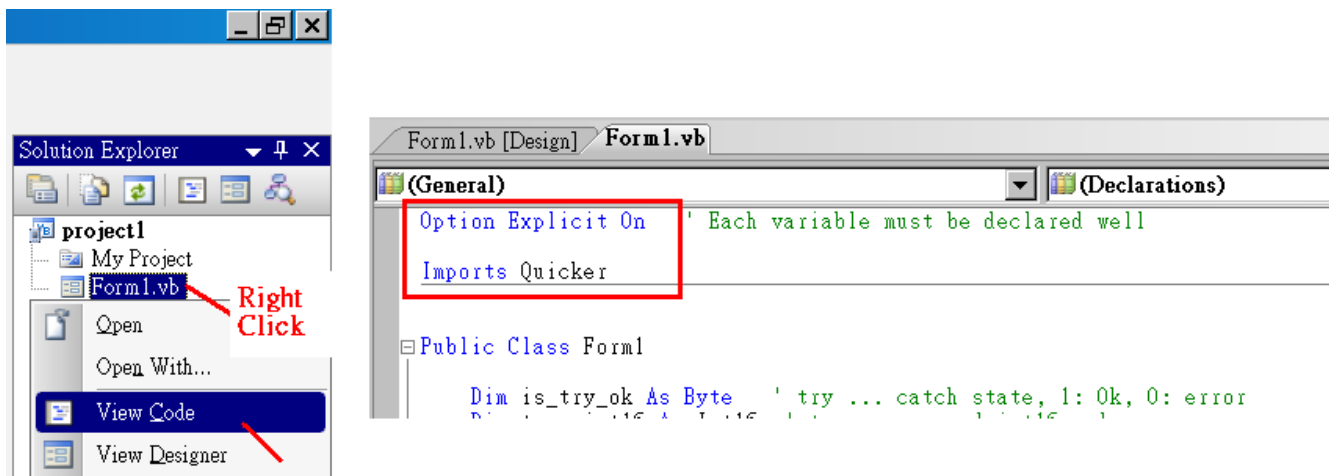
3. Click the “**Browse**” button. Select the “**QuickerNet.dll**” from **WinPAC-8xx7 CD-ROM : \napdos\isagraf\wp-8xx7\vb.net_2008_demo\wp_vb01\vb01** subfolder or from your own location.



4. When both “mscorlib” and “QuickerNet.dll” are added, please double click on “My Project” to check if the “QuickerNet.dll” is well added.



5. Right-click on the “Form1.vb” and select “View Code” from the pop-up. Move cursor to top and insert the “Option Explicit On” and “Imports Quicker” in the first two statements.

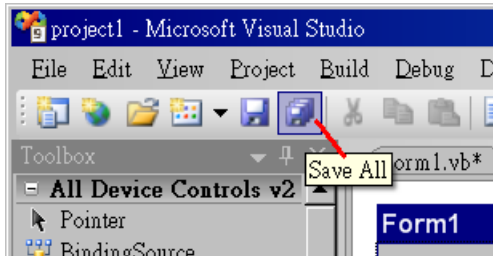


Then you can design all required objects and actions inside your VB Forms.

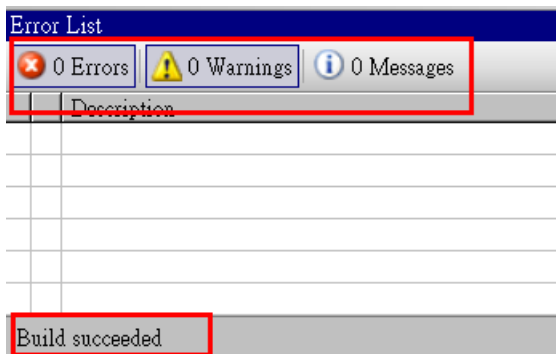
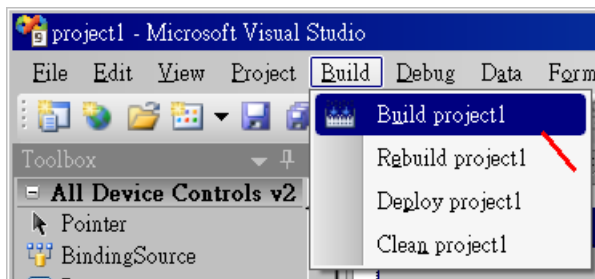
6.3 Compiling an Application Program

When you have finished writing a program, you can build an application by the following steps.

1. Remember to save at any time for safety.



2. Then compile (Build) the project. The result is listed in the “Error List” windows at the bottom .



3. You can find the execution file in

<Your VB.net Project folder> \bin\Release\ <project_name>.exe

Please copy this execution file to the WinPAC-8xx7's \System_Disk\ISaGRAF\ path to run it.

Note: User may copy the VB.net execution file to other path to run it but there should contain at least three DLL files with it or it can not run correctly.

For ex, the project1.exe can run in the \Micro_SD\ path if there is three plus one file in it. The “project1.exe”, “QuickerNet.dll”, “Quicker.dll” and “Mscorlib.dll” . (The “QuickerNet.dll”, “Quicker.dll” and “Mscorlib.dll” can be copied from the WinPAC-8xx7's “\System_disk\ISaGRAF” path)

6.4 QuickerNET.DLL

This section we will focus on the description of the application example of QuickerNET.DLL functions. There are some functions that can be used to R/W data from/to the ISaGRAF softlogic. The functions of QuickerNET.DLL can be clarified as two groups as depicted as below:

1. Digital R/W Functions
2. Analog R/W Functions

6.4.1 Digital R/W Functions

■ UserSetCoil

Description:

This function is to set the value to a Boolean variable by Modbus network address.

Syntax:

ScanKernel.UserShare.UserSetCoil (iUserAddress As System.UInt16, iStatus As byte)

Parameter:

iUserAddress : Specify the Modbus Network Address of Variable (1 to 8191)

iStatus : Set the status. For instance, iStatus = 1 for True, iStatus = 0 for False

Return Value:

None

Example:

‘ Set the output variable of Modbus Network Address “1” to True.

ScanKernel.UserShare.UserSetCoil(Convert.ToUInt16(1), 1)

Demo program :

WinPAC-8xx7 CD-ROM: \napdos\isagraf\wp-8xx7\vb.net_2008_demo\wp_vb01

■ UserGetCoil

Description:

This function is to get the value from a boolean variable by Modbus network address.

Syntax:

ScanKernel.UserShare.UserGetCoil (iUserAddress As System.UInt16, ByRef iStatus As byte)

Parameter:

iUserAddress : Specify the Modbus Network Address of Variable (1 to 8191)
iStatus : Get the variable status , iStatus = 1 for True, iStatus = 0 for False

Return Value:

None

Example:

' Get the variable status of Network Address "1".
Dim iStatus As Byte
ScanKernel.UserShare.UserGetCoil(Convert.ToUInt16(1), iStatus)

Demo program :

WinPAC-8xx7 CD-ROM: \napdos\isagraf\wp-8xx7\vb.net_2008_demo\wp_vb01

6.4.2 Analog R/W Functions

■ **UserSetReg_short** ■ **UserSetReg_long** ■ **UserSetReg_float**

Description:

These functions are to set 16-bit short integer , 32-bit long integer & 32-bit float value to the specified Modbus network address.

Syntax:

ScanKernel.UserShare.UserSetReg_Short (ByVal iUserAddress As System.UInt16, ByRef iStatus As Integer) As Byte

ScanKernel.UserShare.UserSetReg_Long (ByVal iUserAddress As System.UInt16, ByRef iStatus As Integer) As Byte

ScanKernel.UserShare.UserSetReg_Float (ByVal iUserAddress As System.UInt16, ByRef iStatus As Single) As Byte

Parameter:

iUserAddress : Specify the Network Address of Variable (1 to 8191)
iStatus : Set the short or long integer or float value.

Example:

' Set a long value "1234567" to the variable of Modbus Network Address "1".

ScanKernel.UserShare.UserSetReg_long(Convert.ToUInt16(1), Convert.ToInt32(1234567))

' Set a short value "-1234" to the variable of Modbus Network Address "3".

ScanKernel.UserShare.UserSetReg_short(Convert.ToUInt16(3), Convert.ToInt16(-1234))

' Set a float value "2.174" to the variable of Modbus Network Address "4".

ScanKernel.UserShare.UserSetReg_float(Convert.ToUInt16(4), Convert.ToSingle(2.174))

Demo program :

WinPAC-8xx7 CD-ROM:

1. \napdos\isagraf\wp-8xx7\vb.net_2008_demo\wp_vb02 for R/W analog I/O
2. \napdos\isagraf\wp-8xx7\vb.net_2008_demo\wp_vb03 for R/W internal long integer, Timer and Real (floating-point) values.

Note:

The long integer & timer & real variable's Network Address No. must occupy 2 No. in the ISaGRAF project (refer to section 4.2 of "User's Manual of ISaGRAF Embedded Controllers" or in the CD-ROM:\napdos\isagraf\wp-8xx7\english_manul\User_Manual_I_8xx7.pdf")

■UserGetReg_short ■UserGetReg_long ■UserGetReg_float

Description:

These functions are to get 16-bit short integer , 32-bit long integer & 32-bit float value from the specified Modbus network address.

Syntax:

ScanKernel.UserShare. UserGetReg_Short (ByVal iUserAddress As System.UInt16, ByRef iStatus As Integer) As Byte

ScanKernel.UserShare. UserGetReg_Long (ByVal iUserAddress As System.UInt16, ByRef iStatus As Integer) As Byte

ScanKernel.UserShare. UserGetReg_Float (ByVal iUserAddress As System.UInt16, ByRef iStatus As Single) As Byte

Parameter:

iUserAddress : Specify the Network Address of Variable (1 to 8191)

iStatus : Get the short or long integer or float value.

Example:

Dim float_val As Single

Dim short_val As Int16

Dim long_val As Int32

' Get float value of the variable of Modbus Network Address "7".

ScanKernel.UserShare.GetUserReg_float(Convert.ToUInt16(7), float_val)

' Get long value of the variable of Modbus Network Address "9".

ScanKernel.UserShare.GetUserReg_long(Convert.ToUInt16(9), long_val)

' Get short value of the variable of Modbus Network Address "11".

ScanKernel.UserShare.GetUserReg_short(Convert.ToUInt16(11), short_val)

Demo program :

WinPAC-8xx7 CD-ROM:

3. \napdos\isagraf\wp-8xx7\vb.net_2008_demo\wp_vb02 for R/W analog I/O

4. \napdos\isagraf\wp-8xx7\vb.net_2008_demo\wp_vb03 for R/W internal long integer, Timer and Real (floating-point) values.

Note:

The long integer & timer & float variable's Network Address No. must occupy 2 No. in the ISaGRAF project (refer to section 4.2 of "User's Manual of ISaGRAF Embedded Controllers" or in the CD-ROM:\napdos\isagraf\wincon\english_manu\User_Manual_I_8xx7.pdf")

Chapter 7 : EVC++ Program Running In WinPAC Access To ISaGRAF Variables

User can write his EVC++ 4.0 application to access to the ISaGRAF variables running at the same WinPAC-8xx7 by using the below functions for Read/Write boolean, word, long and float value.

The include file and library at design time are “WinConAgent.h” and “Quicker.lib”. (WinPAC-8xx7 CD-ROM: \napdos\isagraf\wp-8xx7\evc++_lib\). The DLL at run time is the “Quicker.dll” which is in WinPAC-8xx7 's \System_Disk\isagraf\ (Please copy the execution file after successfully compilation to the WinPAC 's \System_Disk\isagraf\ and then run it.)

Set boolean value:

```
unsigned char UserSetCoil(unsigned short iUserAddress, unsigned char iStatus);
```

iUserAddress: 1 to 8191 (Variable's network address in ISaGRAF project)

iStatus: 0: set boolean to False, 1: set boolean to True

for ex. UserSetCoil(100 , 1) // set boolean at network addr 100 as True

Set word or float or long value:

```
unsigned char UserSetReg(unsigned short iUserAddress, long *iStatus,  
                        unsigned char iDType);
```

iUserAddress: 1 to 8191 (Variable's network address in ISaGRAF project)

iStatus: A pointer to a long type, which stores the data to set

iDType 0: type is word

1: data type is float

2: data type is long (use long for Timer value in ISaGRAF, unit is ms)

for ex.

```
float float_val;
```

```
long word_val, long_val;
```

```
long *temp_val;
```

```
// set word_val (-32768 to +32767) to ISaGRAF variable with network address 1
```

```
word_val = -20000 ;
```

```
temp_val = (long *)&word_val;
```

```
UserSetReg(1 , temp_val, 0);
```

```
// set float_val to ISaGRAF variable with network address 2
```

```
float_val = 1.2345 ;
```

```
temp_val = (long *)&float_val;
```

```
UserSetReg(2 , temp_val, 1);
```

```
// set long_val to ISaGRAF variable with network address 4
```

```
long_val = 12345678 ;
```

```
temp_val = (long *)&long_val;
```

```
UserSetReg(4 , temp_val, 2);
```

Get boolean value:

```
unsigned char UserGetCoil(unsigned short iUserAddress, unsigned char *iStatus);
```

iUserAddress: 1 to 8191 (Variable's network address in ISaGRAF project)

iStatus: 0: boolean is False, 1: boolean is True

for ex.

```
unsigned char bVal;  
UserGetCoil(5, &bVal) // get boolean value at network addr 5
```

Get word or float or long value:

```
unsigned char UserGetReg(unsigned short iUserAddress, long *iStatus,  
                        unsigned char iDType);
```

iUserAddress: 1 to 8191 (Variable's network address in ISaGRAF project)

iStatus: A pointer to a long type, which stores the data returned

iDType 0: type is word

1: data type is float

2: data type is long (use long for Timer value in ISaGRAF, unit is ms)

for ex.

```
float float_val;  
long word_val, long_val;  
long ret_val;
```

```
// get word_val (-32768 to +32767) of ISaGRAF variable with network address 10
```

```
UserGetReg(10, &ret_val, 0);  
if ( ret_val >= 0 && ret_val <= 32767 ) word_val = ret_val;  
else word_val = ret_val | 0xFFFF0000;
```

```
// get float of ISaGRAF variable with network address 11
```

```
UserGetReg(11, &ret_val, 1);  
float_val = *(float *) (&ret_val);
```

```
// get long of ISaGRAF variable with network address 13
```

```
UserGetReg(13, &ret_val, 2);  
long_val = ret_val;
```

Note:

The long integer, timer and float variable's Network Address No. must occupy 2 No. in the ISaGRAF project (Please refer to section 4.2 of "User's Manual of ISaGRAF Embedded Controllers" or in the WinPAC-8xx7 CD-ROM:

\napdos\isagraf\wp-8xx7\english_manu\ User_Manual_I_8xx7.pdf")

Chapter 8 : InduSoft Project Running In WinPAC Access To ISaGRAF Variables

Note: If the HMI program behavior is not so smooth or slow, please refer to Appendix F. The WinPAC-8xx7 (or WP-8xx7) is the abbreviation of the WinPAC-8147 / 8447 / 8847. The WinPAC-8xx6 (or WP-8xx6) is the abbreviation of the WinPAC-8146 / 8446 / 8846.

Important: Please always set a **fixed IP** address to the WinPAC-8xx7/8xx6. (No DHCP)
Recommend to use NS-205 or NS-208 Industrial Ethernet Switch for WinPAC.

Please refer to WinPAC-8xx7 CD-ROM: \napdos\isagraf\wp-8xx7\english_manu\ "user_manual_i_8xx7.pdf" for detailed ISaGRAF English User's Manual.
WinPAC-8xx6 supports InduSoft and ISaGRAF logic running in the same controller.

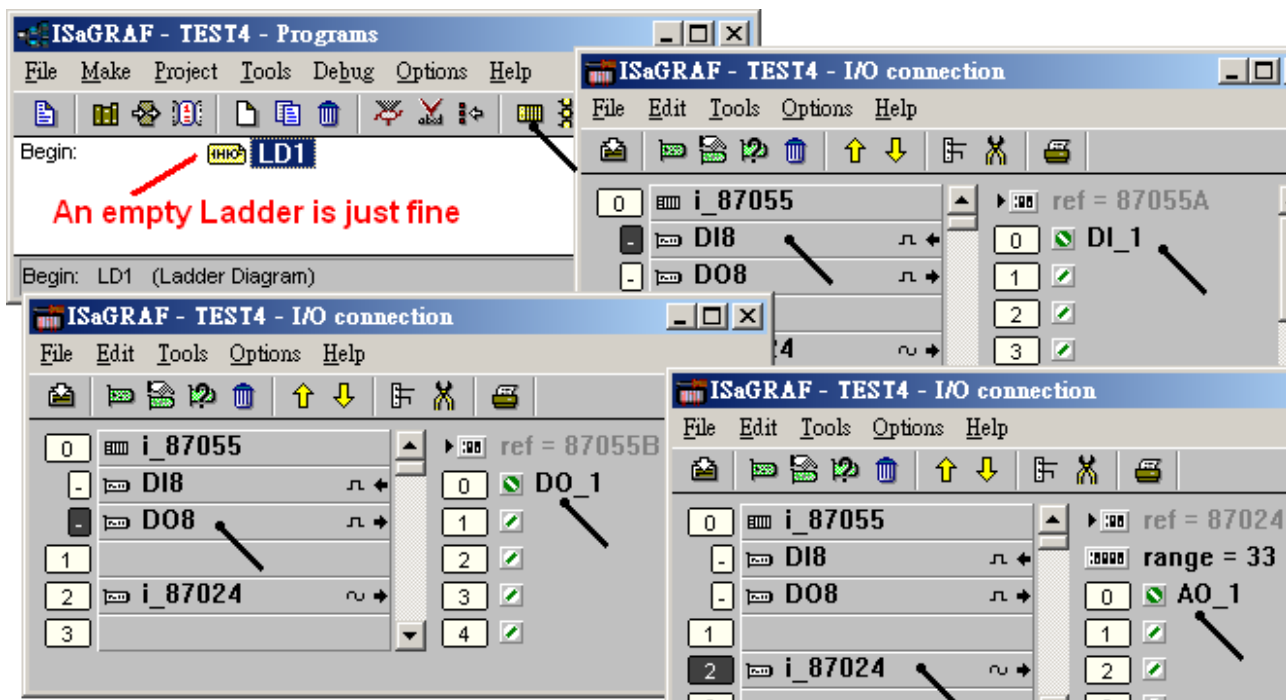
A simple example to run InduSoft & ISaGRAF logic in the same controller:

Step 1: Create a new ISaGRAF project as below.

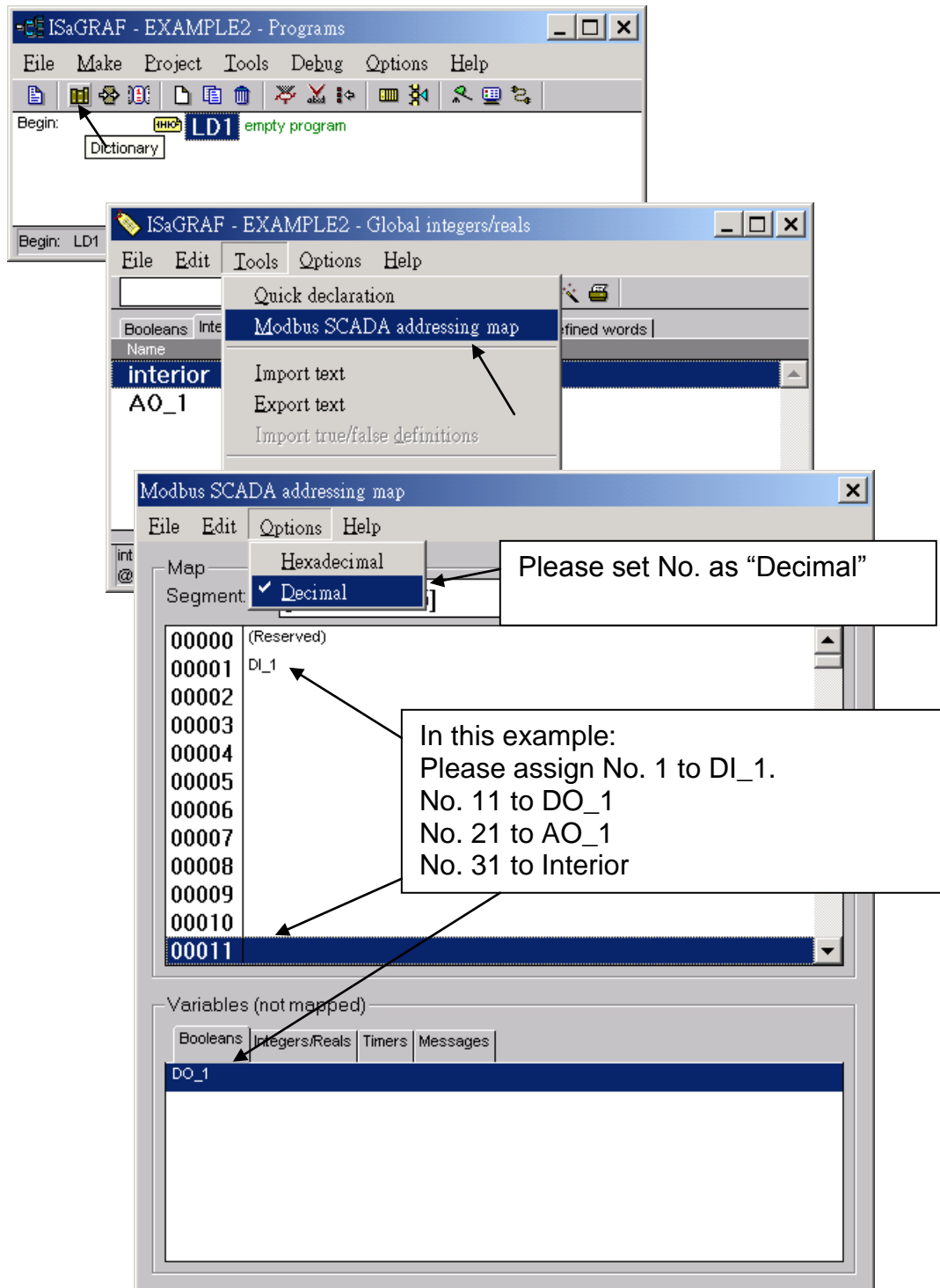
This demo uses a DIO module I-87055W in slot 0 of WinPAC-8xx6, and an AO module I-87024W in slot 2 and one internal variable defined as follow.

Variable Type	Name	Network Address	Comment	Attributes
Boolean	DI_1	1	87055W DI channel 1	Input
Boolean	DO_1	11	87055W DO channel 1	Output
Integers	AO_1	21	87024W AO channel 1	Output
Integers	Interior	31	Internal variable	Internal

If you are not familiar with ISaGRAF, please refer to section 4.1 to 4.3.



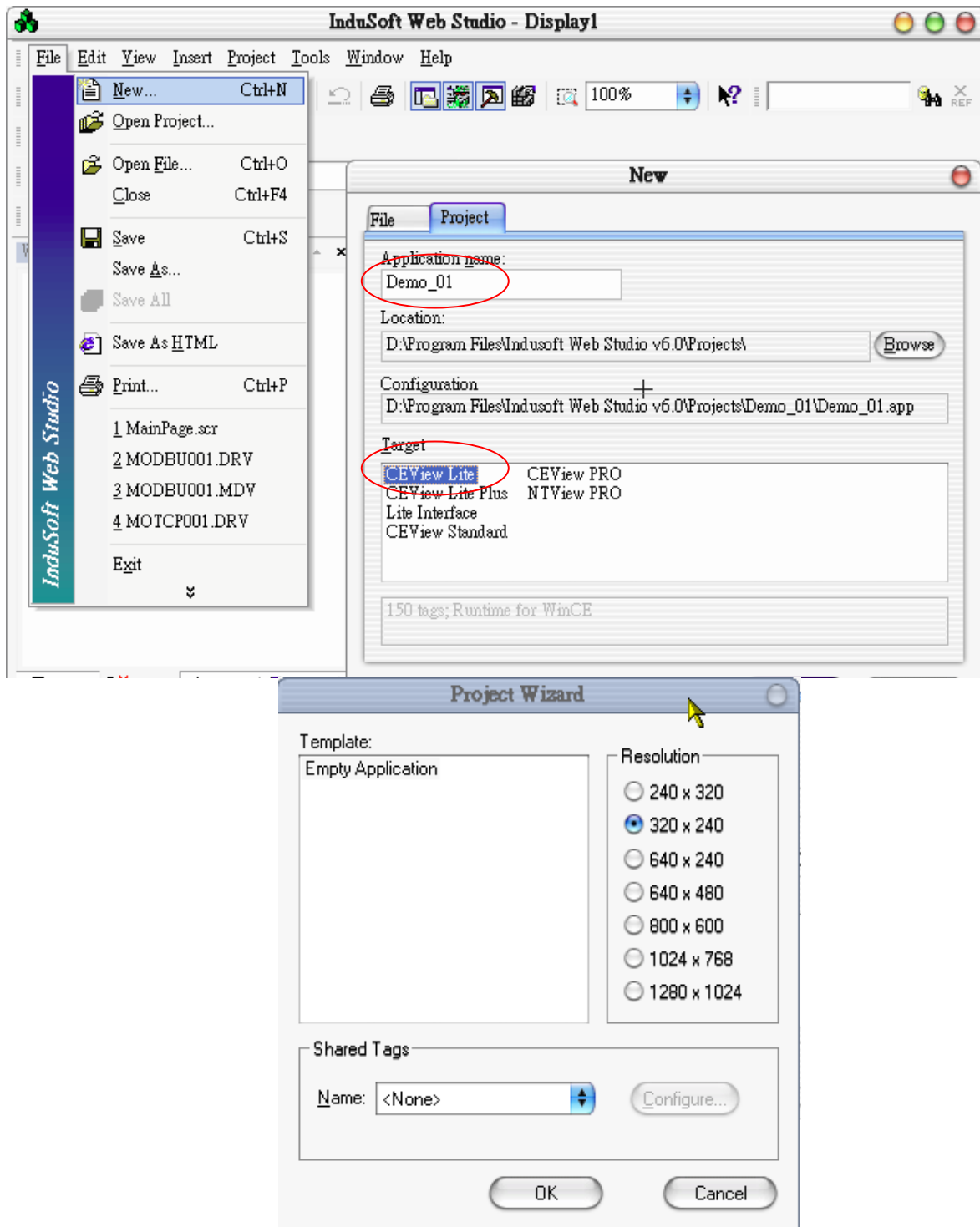
The ISaGRAF variables to be exchanged with InduSoft must be declared with a Modbus “Network Address” as below.



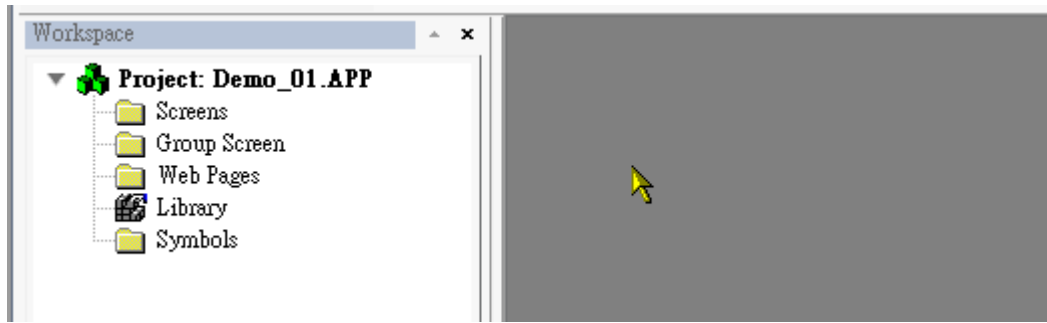
Please save & compile the ISaGRAF example project & then download to the WP-8xx6. If you are not familiar with it, please refer to Section 4.1 to 4.3

Step 2: Create an InduSoft project.

Select “File -> New” from the “InduSoft Web Studio” main menu, the “New” window will appear and click on “Project” tab. Then type in the name for the new user’s project in the “Application name” and select “CEView Lite” in the “Target”. Press “OK” and the “Project Wizard” window will appear. Select “Empty Application” on the “Template”, “320 x 240” on the “Resolution” and “None” on the “Shared Tags”.

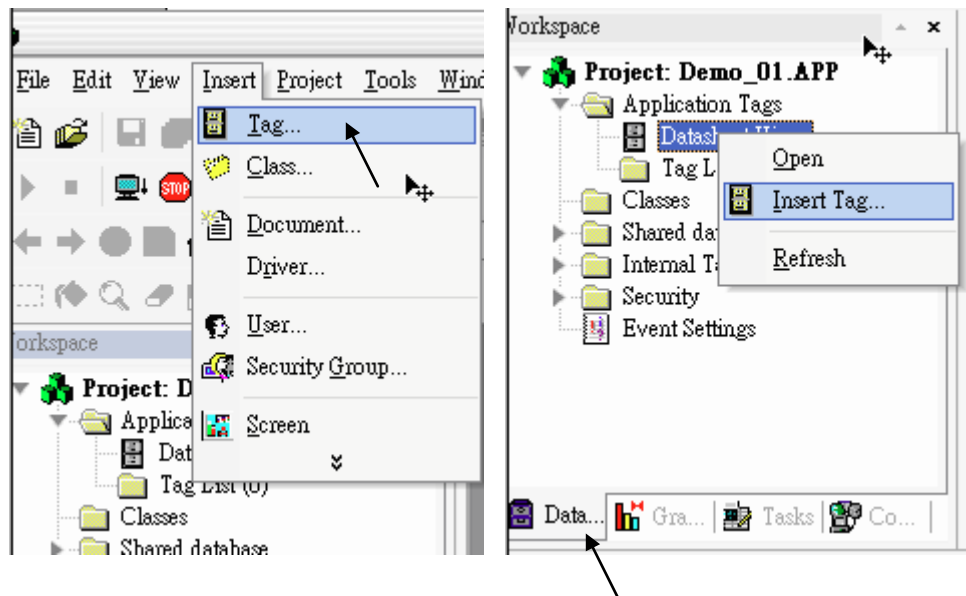


Now you could see the new project shown on the “Workspace” window as below.

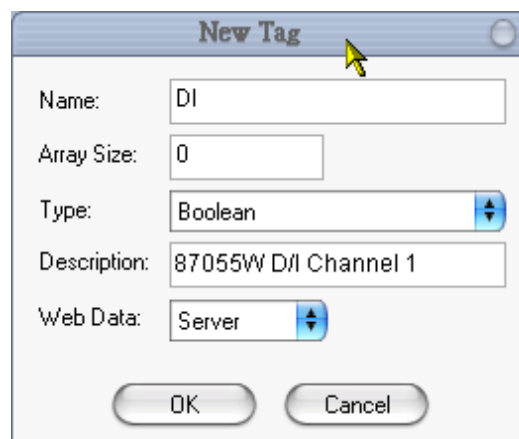


Define application tags

Select “Insert -> Tag” on the main menu bar or click on the right button of the mouse in the “Database” tab of the “Workspace”.



The “New Tag” window will show as below.

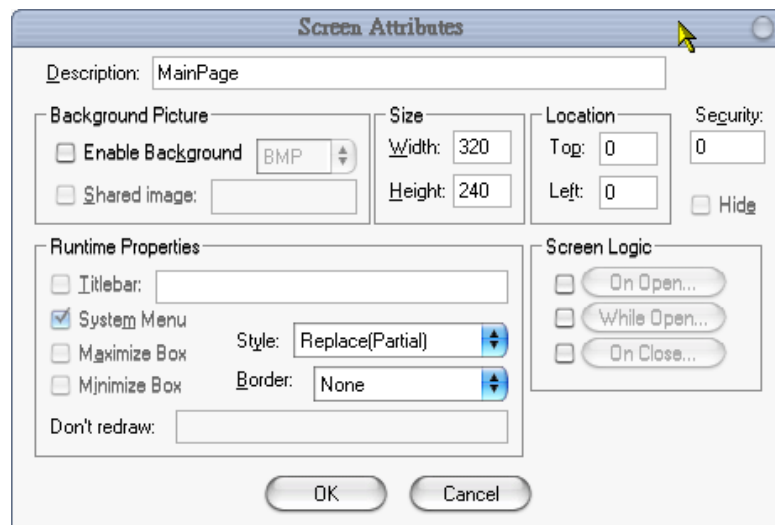
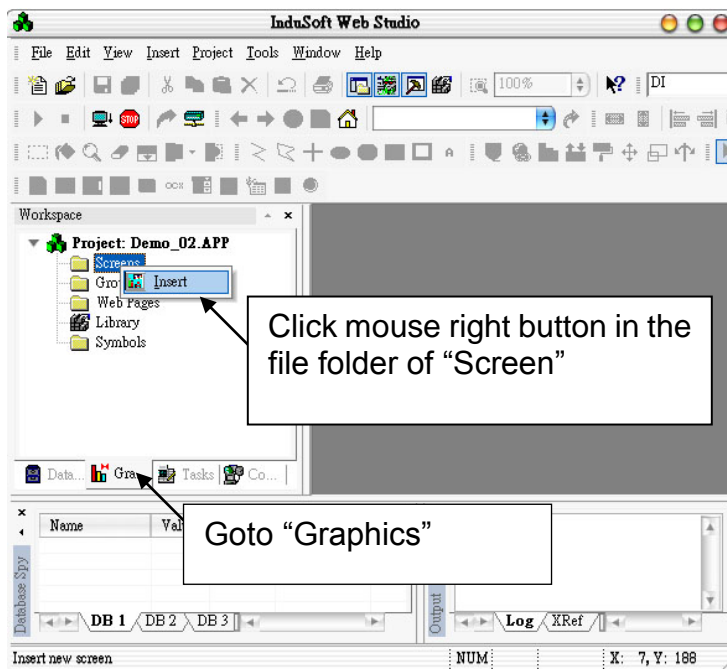


This demo uses a DIO module I-87055W, an AO module I-87024W and one internal variable defined as follow. Please create these tags one by one.

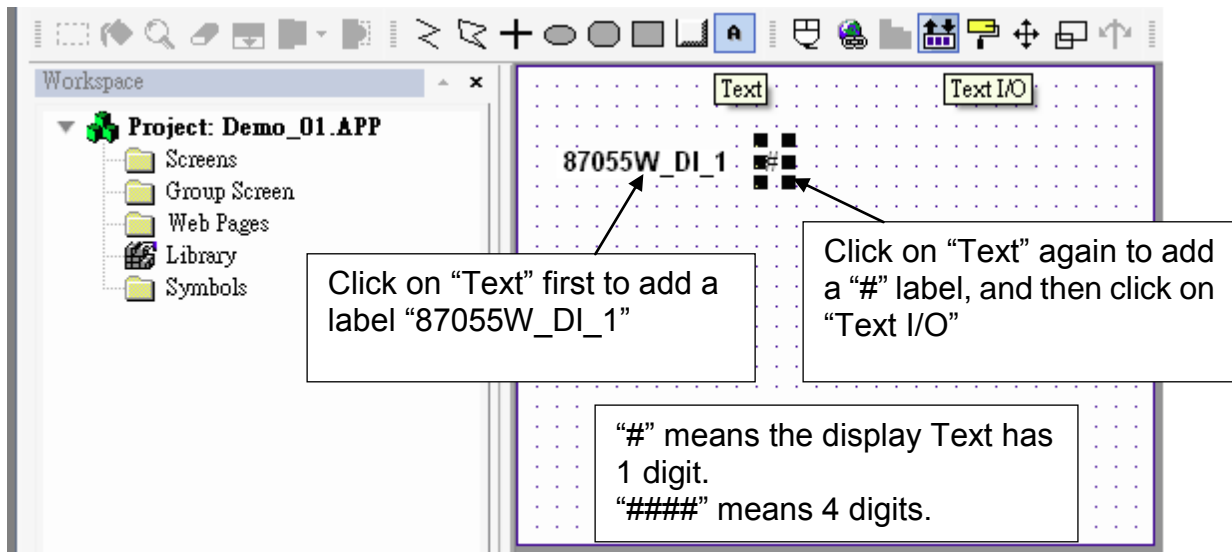
Type	Name	Array Size	Description	Web Data
Boolean	DI	0	87055W DI channel 1	Input
Boolean	DO	0	87055W DO channel 1	Output
Integers	AO	0	87024W AO channel 1	Output
Integers	Interior	0	Internal variable	Internal

Create main screen

Click mouse right button in the file folder of "Screen" of the "Graphics" tab in the "Workspace" then the "Screen Attributes" window appears. Setting screen attributes such as "Size", "Location", "Runtime Properties" and "Background Picture" then press "OK" to edit screen.

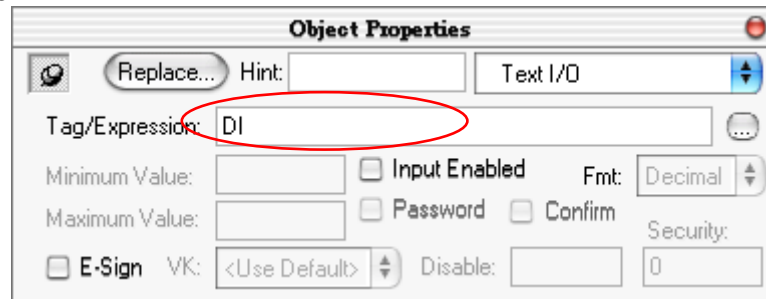


Select “Text” icon, then click on the main screen where want to establish a text and type “87055W_DI_1”. And select “Text” again following the previous text and type “#” then select

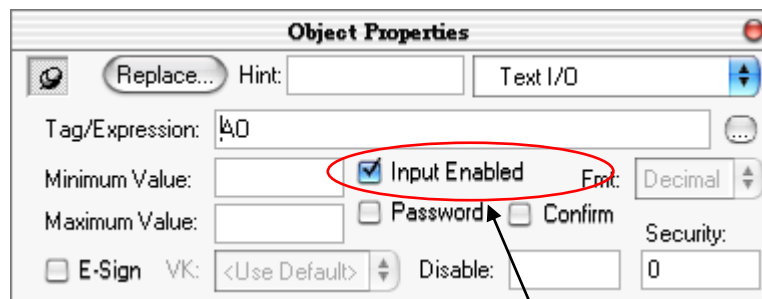


“Text I/O” (# means 1 digit, #### means 4 digits, ##### means 6 digits)

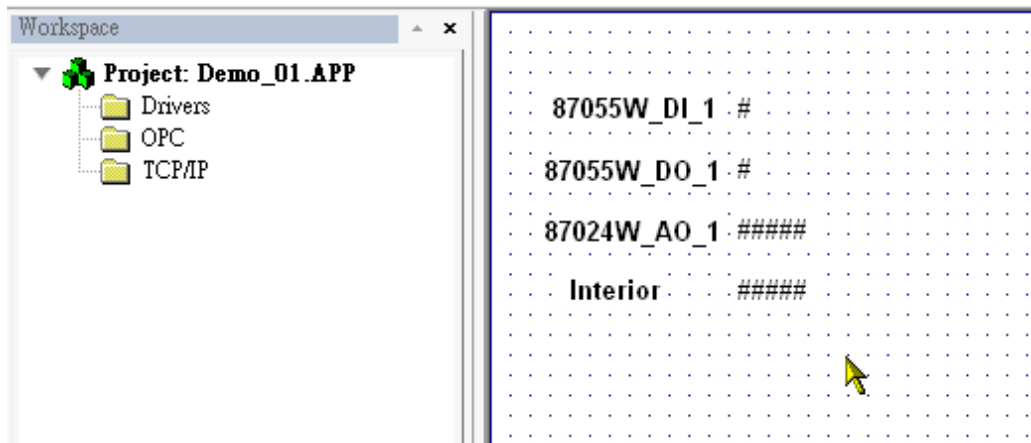
Double click the “#” object and the “Object Properties” window will show as below then type DI in the “Tag/Expression”.



Repeat former method to create other objects and click “Save” icon on the main menu to save this main screen page as “MainPage.scr”.(**Select “File → Save As HTML” to create this screen that can be visualized in a remote station using a regular web browser.**) Note: For the Output object, as 87024W_AO_1 and 87055W_DO_1, the “Input Enabled” of the “Text I/O” should be checked as below.

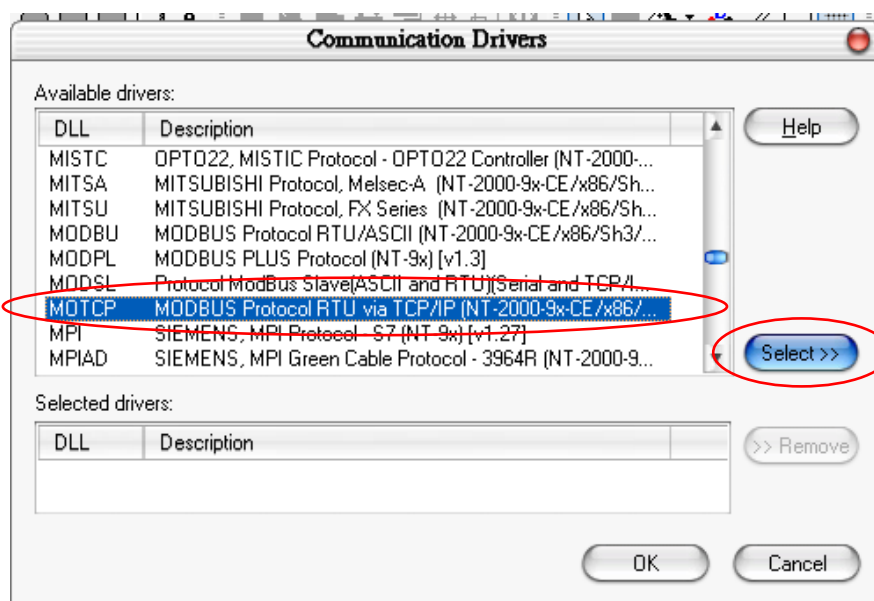
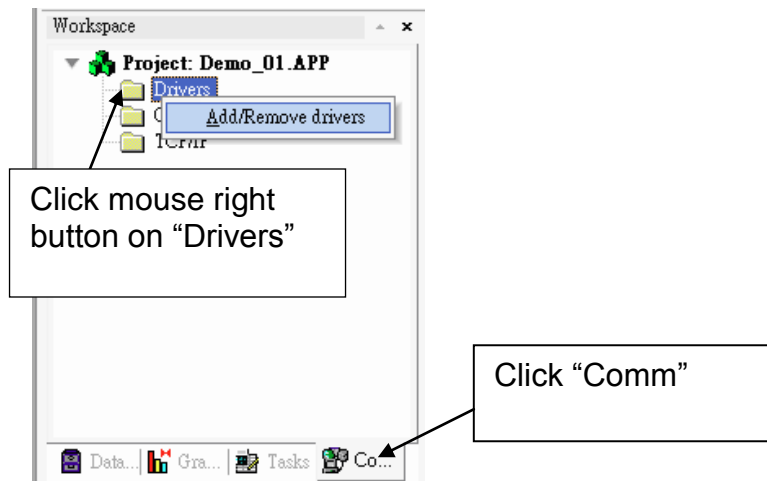


Check on it means user may input its value at run time.

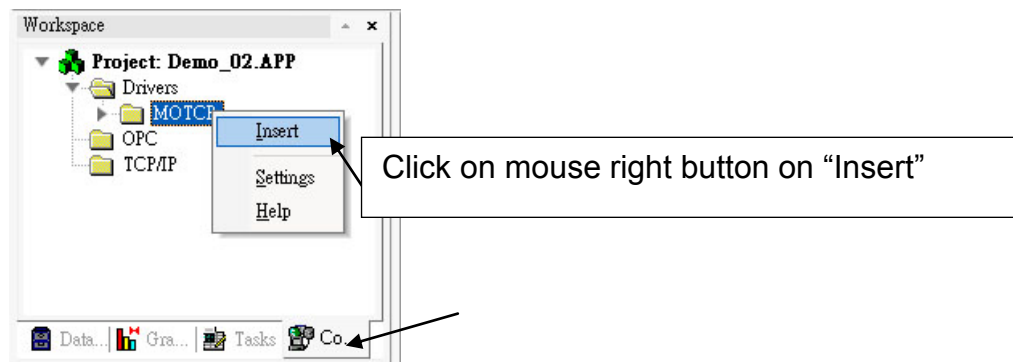


Create Modbus TCP workspace

Click "Comm" in the "Workspace" and in the folder of "Drivers" click right mouse button and select "Add/Remove drivers" to open "Communication Drivers" window as below.



Click “MOTCP” driver then click “Select” and click “OK” to close this window. Expanding file folder of “Drivers” and it will show a file folder named “MOTCP”. Click right mouse button and select “Insert” to add a workspace of Modbus TCP. When a “MOTCP001.DRV” window appears, fill in following data as corresponding field.



What does “127.0.0.1:502:1” mean ?
 “127.0.0.1” is the local host IP address. It means send data to the same controller. “502” is the Modbus TCP/IP port No. , the last “1” is the Net-ID of the WinPAC-8xx7.

Station: 127.0.0.1:502:1 Header: 1X:0

	Tag Name	Address	Div	Add
1	DI	1		

1X:0 is for reading “Boolean” data
 0X:0 is for writing “Boolean” data
 3X:0 is for reading short “integer” data (16-bit integer, Word: -32768 to +32767)
 4X:0 is for writing short “integer” data (16-bit integer , Word: -32768 to +32767)
 DW:0: is for reading & writing long “integer” (32-bit integer, Double Word)
 FP:0: is for reading & writing floating point data (32-bit REAL)
 For more details, please refer form as below

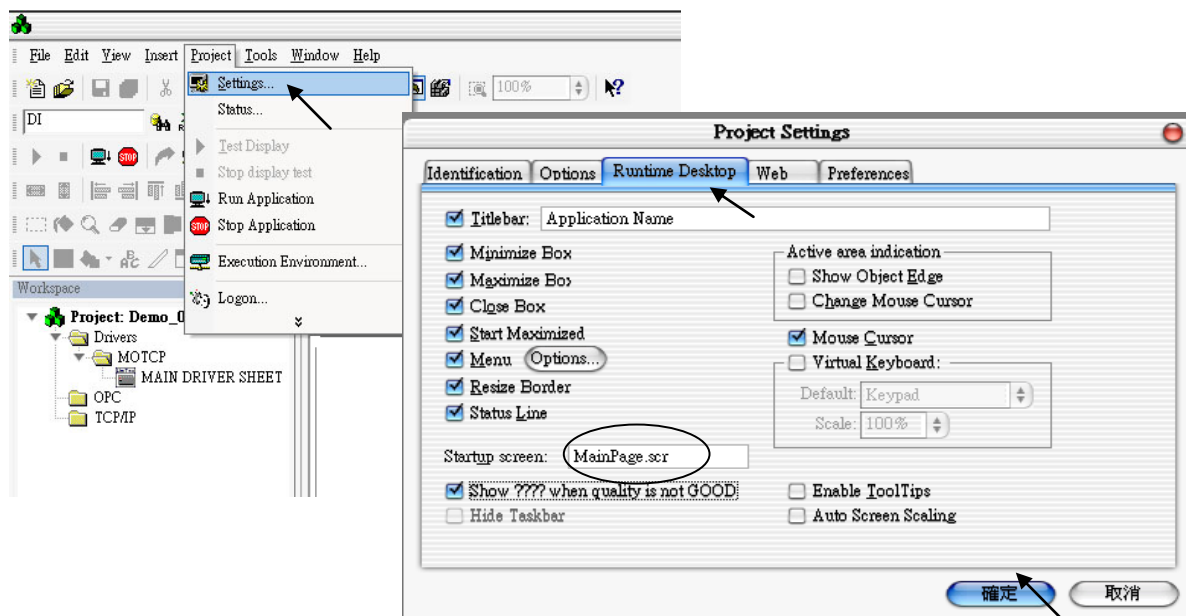
Data Type	Sample Syntax	Valid Range of Initial Addresses per Worksheet	Comments
0X	0X:1	Varies according to the equipment	Coil Status: Read and write events using Modbus instructions 01, 05, and 15
1X	1X:5	Varies according to the equipment	Input Status: Read events using Modbus instructions 02
3X	3X:4	Varies according to the equipment	Input Register: Read events using Modbus instruction 04
4X	4X:5	Varies according to the equipment	Holding Register: Read and write events using Modbus instructions 03, 06, 16
FP	FP:1	Varies according to the equipment	Floating-point value (Holding Register): Read and write float-point values using two consecutive Holding Registers.
DW	DW:2	Varies according to the equipment	32-bit Integer value (Holding Register): Read and write 32-bit integer values using two consecutive Holding Registers.

DRV Name	MOTCP001.DRV	MOTCP002.DRV	MOTCP003.DRV	MOTCP004.DRV
Description	DI	DO	AO	Internal
Station	127.0.0.1:502:1			
Header	1X:0	0X:0	4X:0	3X:0
Tag Name	DI	DO	AO	Interior
Enable Read when Idle	1			1
Wnable Write on Tag Change		1	1	
Address	1	11	21	31

When finished all setting, press “Ctrl + F4” to close all inside windows and save all files.

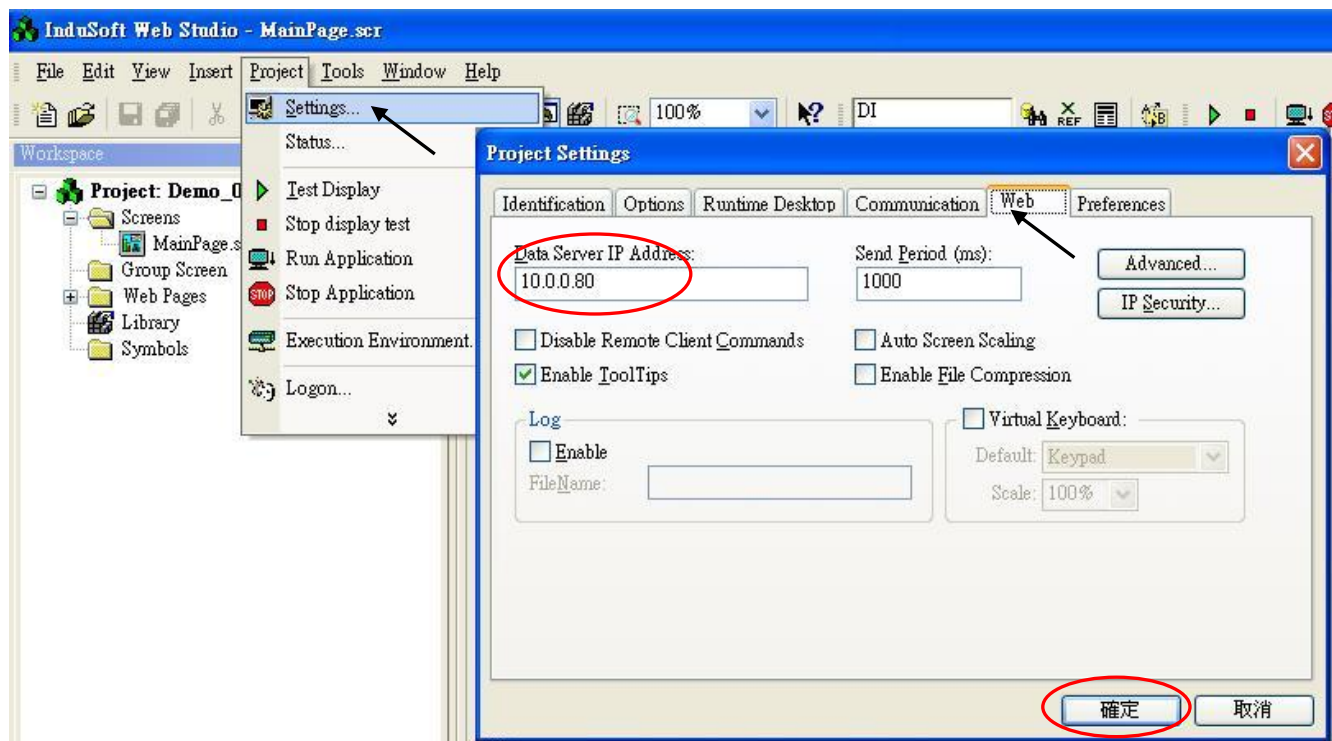
Project Setting

Select “Project -> Settings” to open “Project Settings” window .Click the tab of “Runtime Desktop”. In the “Startup screen” edit box, fill in “MainPage.scr” then click “OK” to close this window.



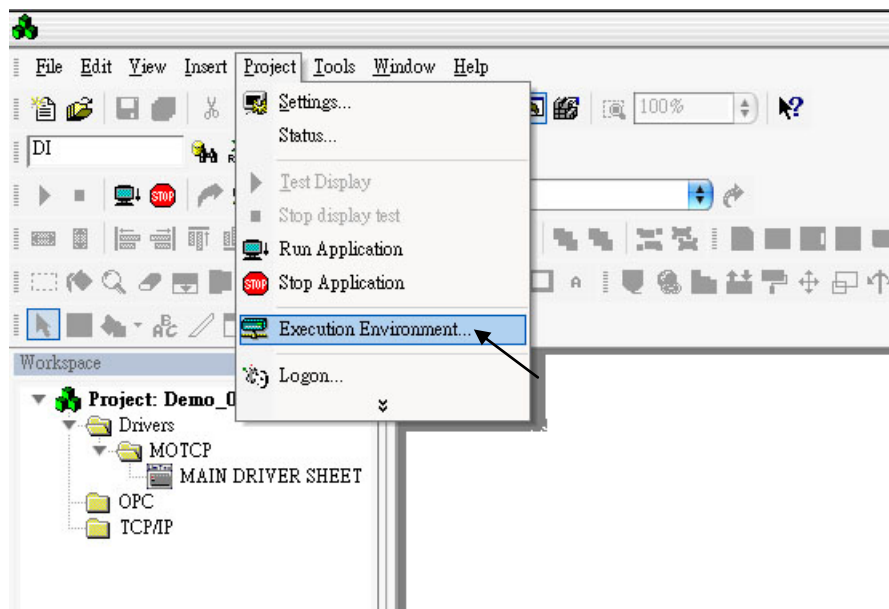
Web Thin Clients

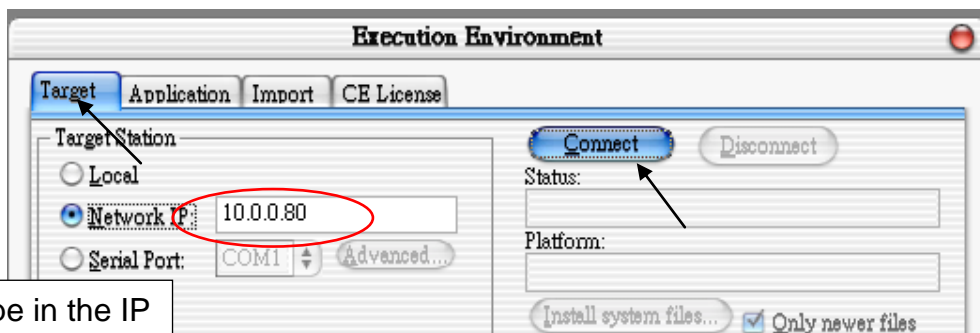
Select “Project -> Settings” to open “Project Settings” window. On the Web tab, select “Data Server IP Address” then type WinPAC-8xx6’s correct IP address and click “OK”.



Download and run the project

Select “Project -> Execution Environment” to open “Execution Environment” window. On the Target tab, select “Network IP” then type WP-8xx6’s correct IP address and click “Connect”.

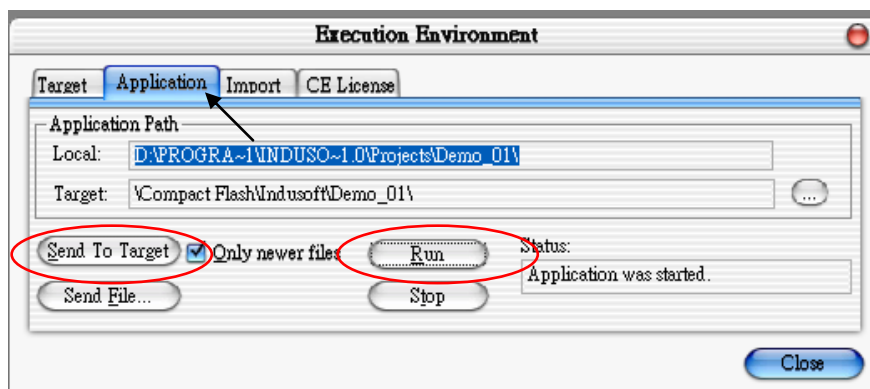
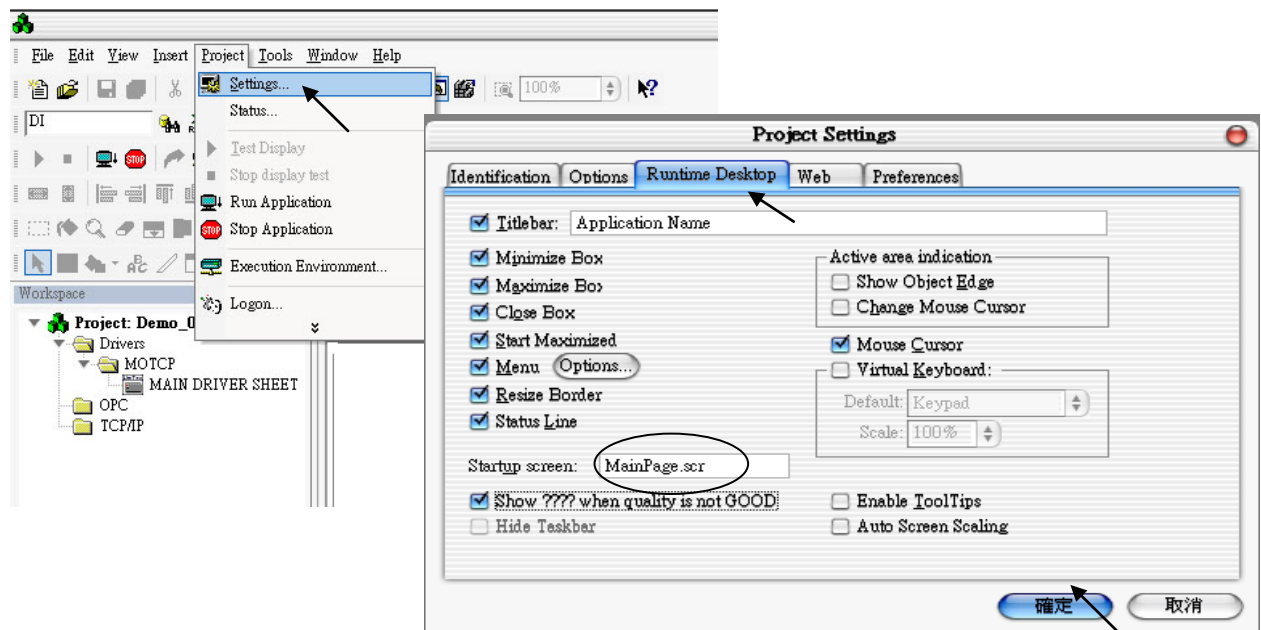




Please type in the IP address of your WP-8xx6.

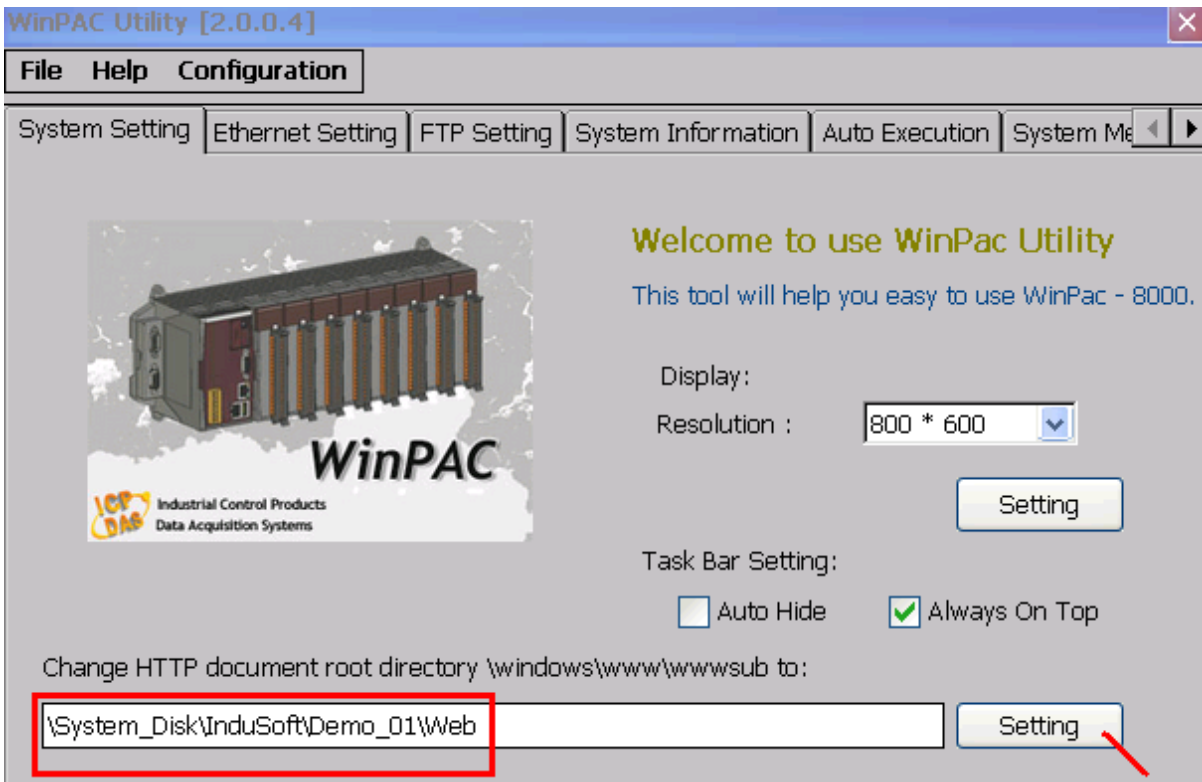
The "Install system files" button can update WinPAC's InduSoft firmware to your PC's current version. Please make sure all file attribution are not "read only" in the WinPAC: 's \System_Disk\InduSoft\ path

If connection is fine, click on the tab of "Application" then click "Send to Target". When download finished, click "RUN" to start the project.



Configuration Web directory of WinCon

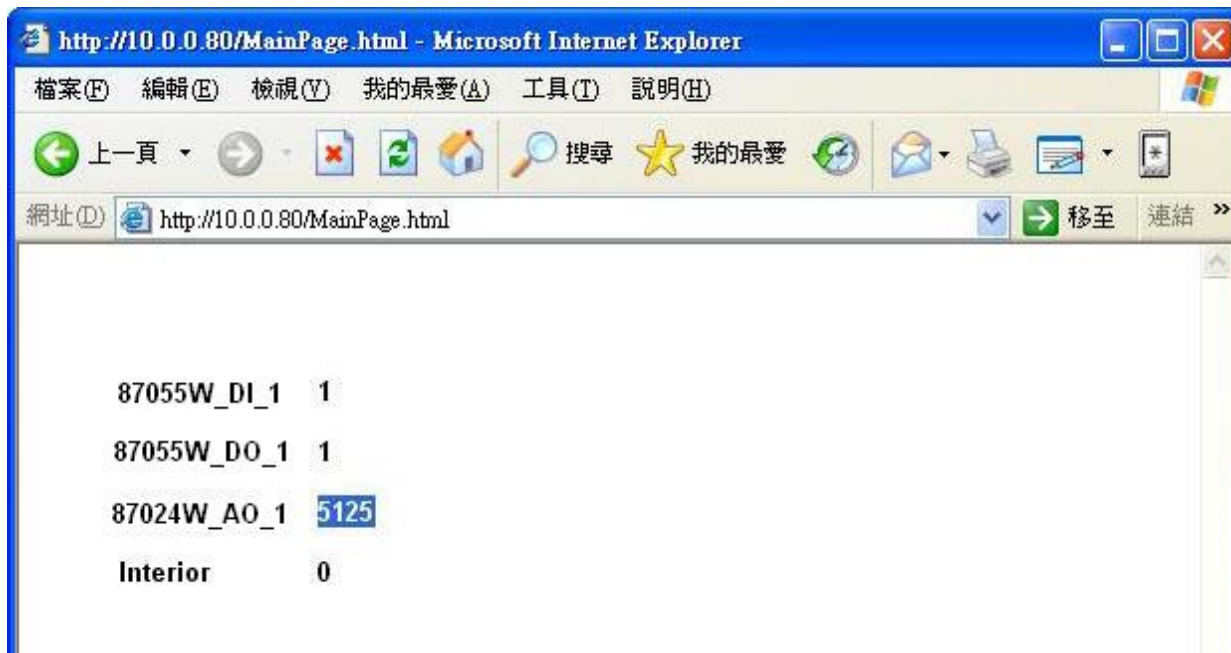
Run WinPAC Utility and change Web directory to “\System_Disk\InduSoft\Demo_01\Web”.



Click “Change” and “Save and Reboot” to finish this configuration.

Visualize your project in a remote station

Run Internet Explorer and type for ex. “<http://10.0.0.80/MainPage.html>”. (use WinPAC's IP)



Note:

Users must install ISSymbol control layer in a remote station at first time. The procedure to install ISSymbol in each operation system is described below:

Windows NT/2K/XP:

Copy the files ISSymbolReg.exe and **ISSymbol.cab** from the \BIN sub-folder of Indusoft Web Studio v6.0 and paste them in any directory of the Web Thin Client station. Make sure that both files are stored in the same directory.

Run ISSymbolReg.exe to register ISSymbol control in the Web Thin Client station.

Windows 9x/ME:

Copy the files ISSymbolReg.exe and **ISSymbolA.cab** from the \BIN sub-folder of Indusoft Web Studio v6.0 and paste them in any directory of the Web Thin Client station. Make sure that both files are stored in the same directory.

Run ISSymbolReg.exe to register ISSymbol control in the Web Thin Client station.

Chapter 9 : ISaGRAF Example Program

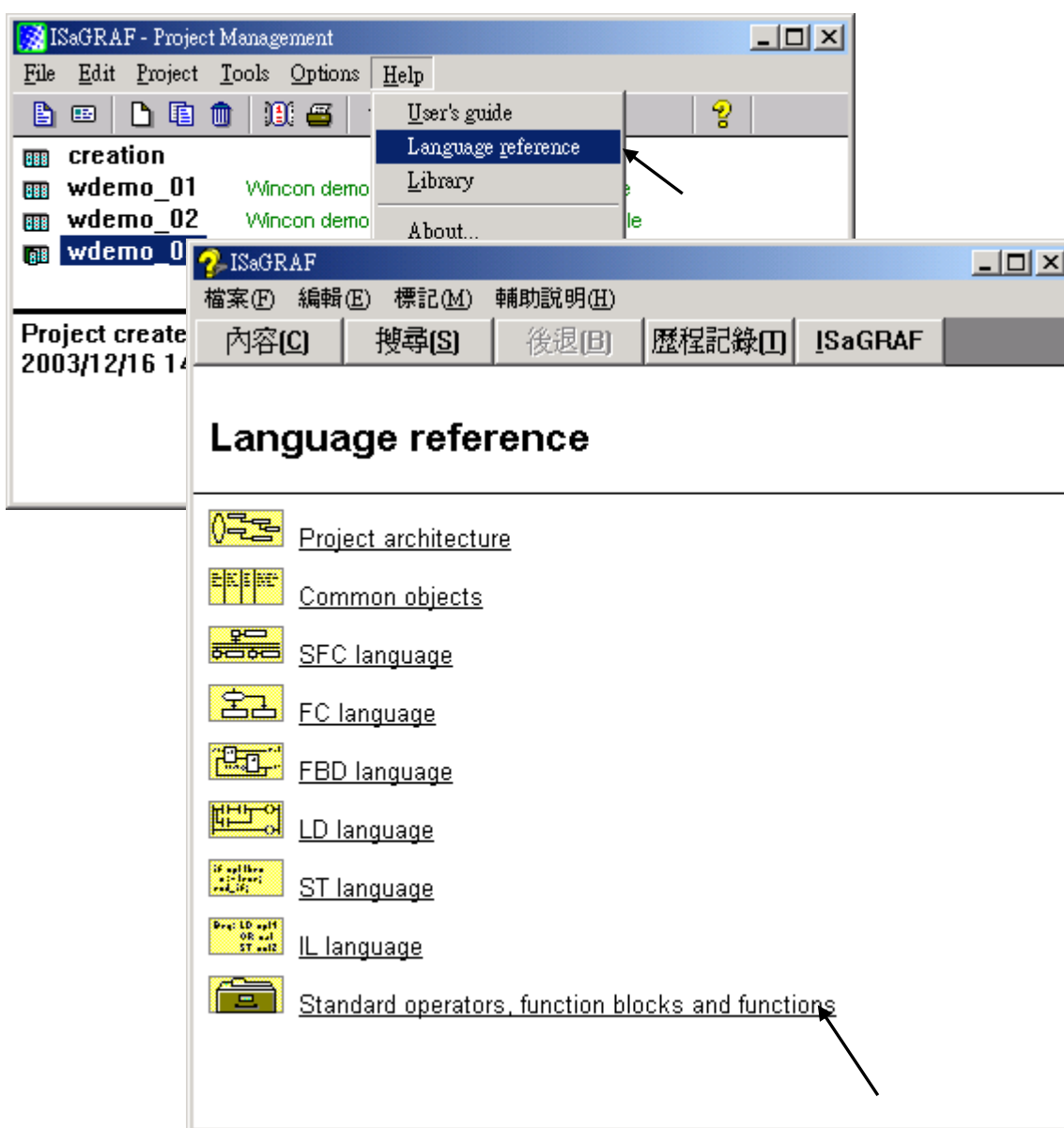
The WinPAC-8xx7 (or WP-8xx7) is the abbreviation of the WinPAC-8147 / 8447 / 8847.

Please refer to WinPAC-8xx7 CD-ROM: \napdos\isagraf\wp-8xx7\english_manu\ "user_manual_i_8xx7.pdf" for detailed ISaGRAF User's Manual.

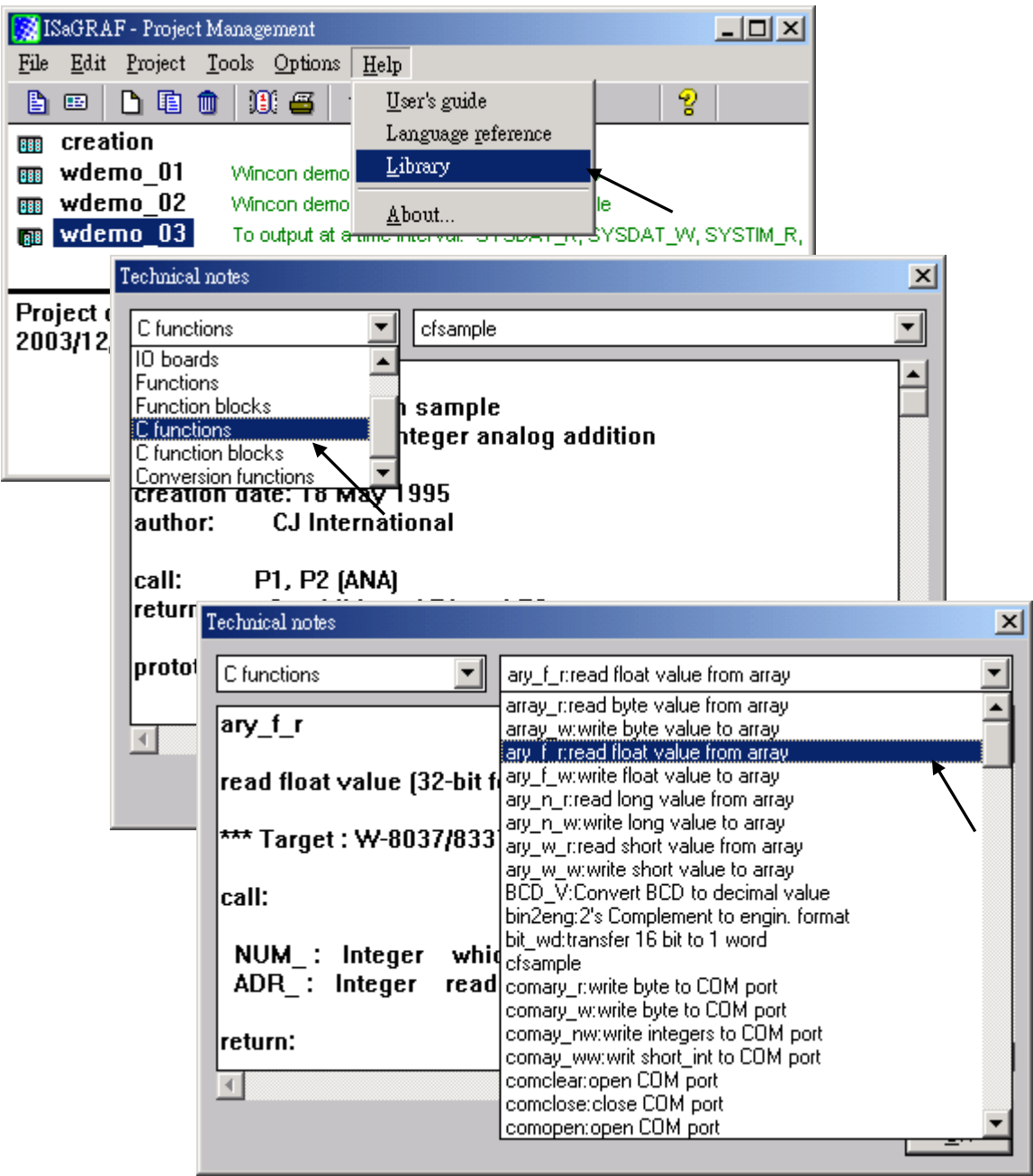
9.1: Get On-Line Help

If you have question, you may email to service@icpdas.com.

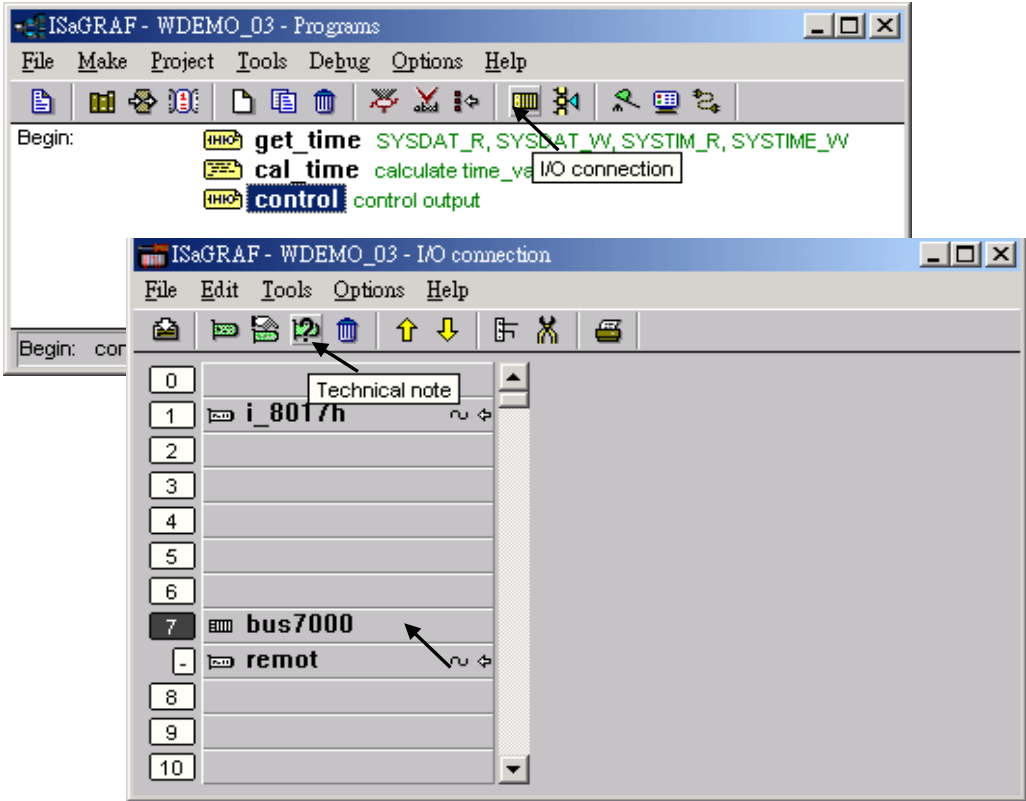
On-line help of ISaGRAF standard functions & function blocks:



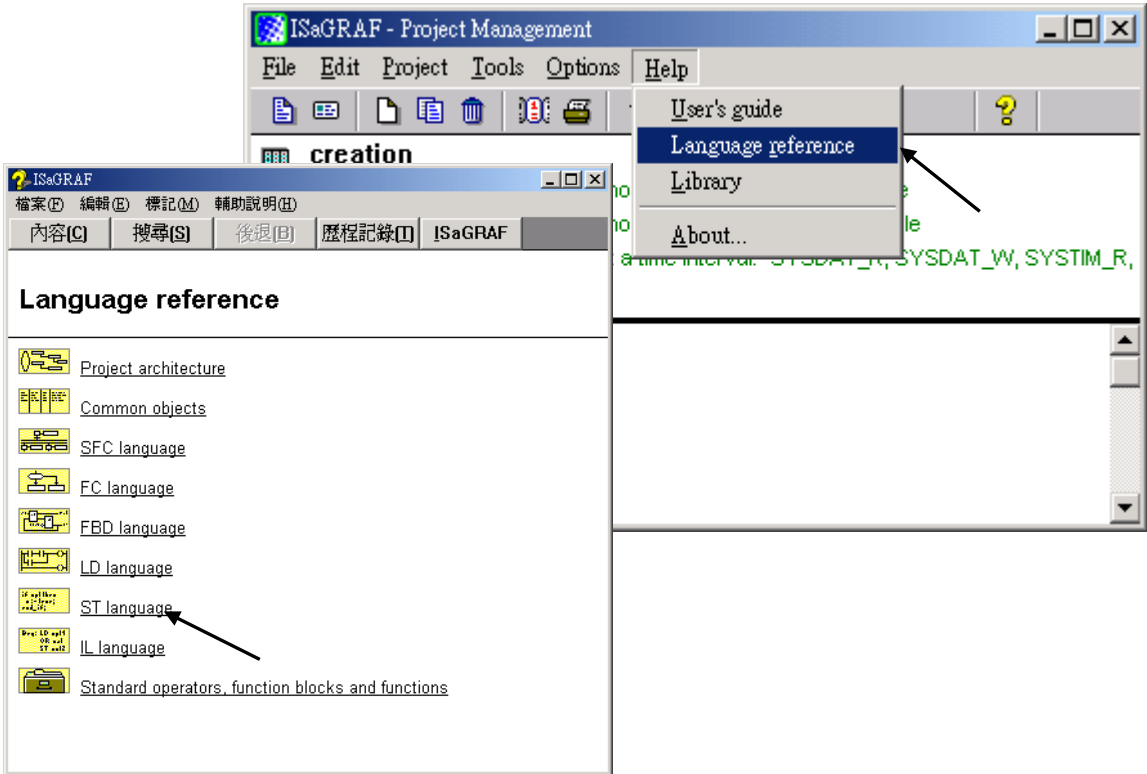
On-line help of ICP DAS add-on functions & function blocks:



On-line help of ICP DAS add-on I/O boards & I/O complex equipments:



On-line help of ISaGRAF languages:



9.2: Installing The ISaGRAF Programming Examples

The ISaGRAF programming examples are installed on the same CD-ROM that you receive with the WinPAC-8xx7 controller system. You will find the programming example files in the **WinPAC-8xx7 CD-ROM: \napdos\isagraf\wp-8xx7\demo**

ISaGRAF User's Manual:

English: \napdos\isagraf\wp-8xx7\english_manu\ "User_Manual_I_8xx7.pdf" and
"User_Manual_I_8xx7_Appendix.pdf"

中文: \napdos\isagraf\wp-8xx7\english_manu\ "Chinese_User_Manual_I_8xx7.pdf"
and "Chinese_User_Manual_I_8xx7_Appendix.pdf"

Example lists:

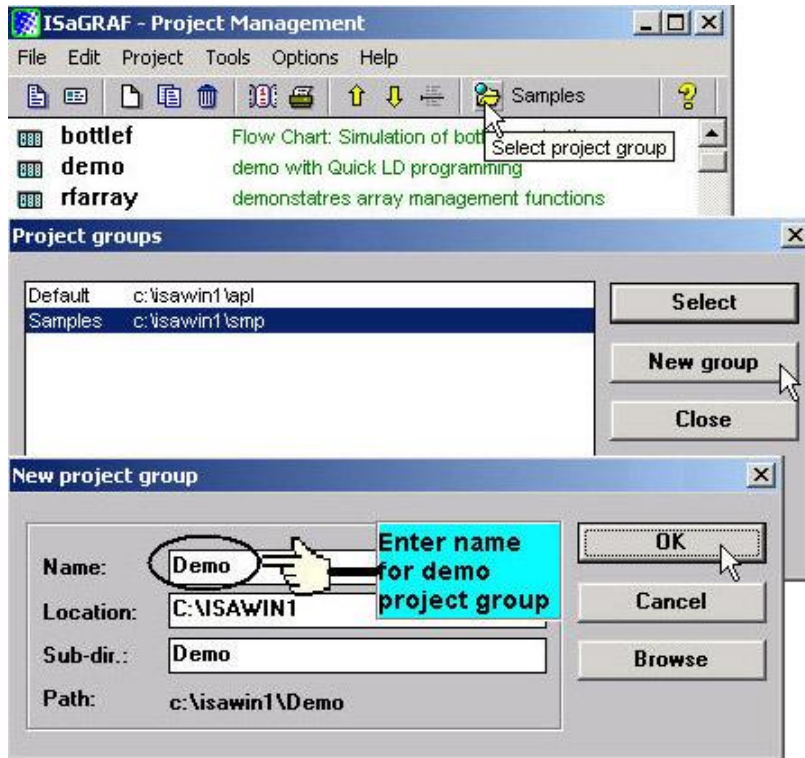
Project Name	Description	I/O Boards Used
wp_vb01	VB.net 2008 demo 01 for WP-8xx7 : DIO demo Please refer to Chapter 6.	slot 0: I-87055W
wp_vb02	VB.net 2008 demo 02 for WP-8xx7. Analog I/O Please refer to Chapter 6.	slot 1: I-87024W slot 2: I-8017HW
wp_vb03	VB.net 2008 demo 03 for WP-8xx7. Read / Write long integer, float & Timer Please refer to Chapter 6.	
wpdmo_01	WinPAC demo_01: R/W float value from file (www.icpdass.com > FAQ > Software > ISaGRAF > 60)	
wpdmo_02	WinPAC demo_02: R/W long integer from file (www.icpdass.com > FAQ > Software > ISaGRAF > 60)	
wpdmo_03	To output at a time interval: SYSDAT_R, SYSDAT_W, SYSTIM_R, SYSTIM_W (ST+QLD)	
wpdmo_04	WinPAC demo_04: User defined Modbus protocol (No using "Mbus")	
	How to use Array Variable ? (www.icpdass.com > FAQ > Software > ISaGRAF > 39)	
wpdmo_05	To do something at some sec later when an event happens (www.icpdass.com > FAQ > Software > ISaGRAF > 17)	slot 0: I-87055W
wpdmo_06	Using Message Array - MsgAry_r , MsgAry_w	
wpdmo_07	Convert float value to string, using real_str & rea_str2	
wpdmo_08	PID control, refer to WinPAC-8xx7 CD: \napdos\isagraf\wp-8xx7\english_manu\ "PID_AL...htm"	
wpdmo_09	Store & backup boolean & long integer value To/From files	
wpdmo_10	Store & backup boolean & long integer value To/From EEPROM	
wpdmo_11	Dir is \Micro_SD ,save 3 values to 3 files per 10 minutes ,change file name per month	

wpdmo_14	Retain variable by Retain_b, Retain_N, Retain_f, Retain_t (www.icpdas.com > FAQ > Software > ISaGRAF > 74)	
wpdmo_16	Dir is \Micro_SD ,save 3 values to 1 file every minute ,change file name every day	
wpdmo19	Send UDP String to PC when alarm happens (using variable array),Time_Gap is 1 sec (Chapter 19.2 of the “ISaGRAF User's Manual”)	slot0: I-87055W
wpdmo19a	Send UDP String to PC 3 sec later, Time_Gap is 250ms (Chapter 19.2 of the “ISaGRAF User's Manual”)	slot0: I-87055W
wpdmo19b	Send UDP Str to PC 3 sec later (wpdmo19a is better), Time_Gap is 250 ms (Chapter 19.2 of the “ISaGRAF User's Manual”)	slot0: I-87055W
wpdmo_20	receive String coming from remote PC or controller via UDP/IP	
wpdmo_21	using "com_MRTU" to disable/enable Modbus RTU slave port,	
wpdmo_22	PWM I/O demo. (Pulse Width Modulation), minimum scale is 2ms for WinPAC	slot 0: I-8055W
wpdmo_23	Send Time String to COM3:RS-232 every second by using COMOPEN, COMSTR_W (www.icpdass.com > FAQ > Software > ISaGRAF > 59)	
wpdmo_24	Send string to COM3 when alarm 1 to 8 happens	slot 0: I-87055W
wpdmo_26	To move some pulse at x-axis of I-8091W of slot 1 in WP-8xx7 (Chapter 18 of the “ISaGRAF User's Manual”)	slot 1: I-8091W
wpdmo_27	Motion x (Chapter 18 of the “ISaGRAF User's Manual”)	slot 1: I-8091W slot 2: I-8090W
wpdmo_28	Motion x-y (Chapter 18 of the “ISaGRAF User's Manual”)	slot 1: I-8091W slot 2: I-8090W
wpdmo_29	Moving to he Abs. position when CMD is given (Chapter 18 of the “ISaGRAF User's Manual”)	slot 1: I-8091W slot 2: I-8090W
wpdmo_30	WP8xx7(10.0.0.102) link two i8KE8 + I/O , one is 10.0.0.108, one is 10.0.0.109 (www.icpdass.com > FAQ > Software > ISaGRAF > 42)	
wpdmo_31	WP8xx7(10.0.0.2) link one i8Ke8 + I/O (10.0.0.109) (www.icpdass.com > FAQ > Software > ISaGRAF > 42)	
wpdmo_32	Set up WP8xx7 as TCP/IP Client & link to other TCP/IP server (1 connection) (Chapter 19.3 of the “ISaGRAF User's Manual”)	slot 0: I-87055W
wpdmo_33	Same as Wpdmo_32 but send message only when event last for larger than 3 seconds	slot 0: I-87055W
wpdmo_36	Read Real Val from Modbus RTU device (www.icpdass.com > FAQ > Software > ISaGRAF > 47 & 75)	
wpdmo_37	Write Real Val to Modbus RTU device (www.icpdass.com > FAQ > Software > ISaGRAF > 47 & 75)	
wpdmo_38	Using Modbus function code 6 to write 16 bits	

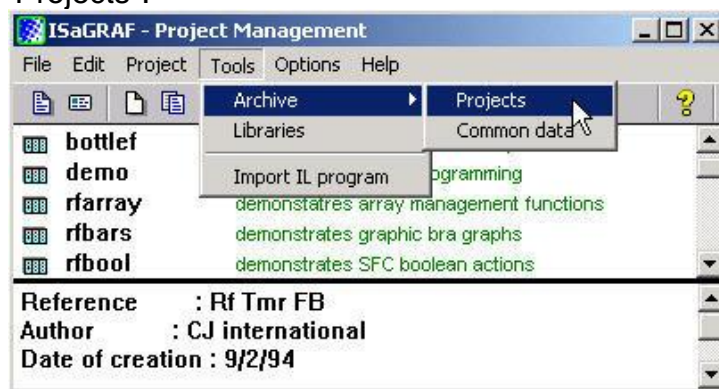
	(www.icpdass.com > FAQ > Software > ISaGRAF > 46 & 75)	
wpdmo_39	WP-8xx7 + I-8172W connecting FRNET I/O modules (www.icpdass.com > FAQ > Software > ISaGRAF > 82)	
	How to read max. 120 Words or max. 60 Long-Integers or max. 60 Real value from Modbus RTU / ASCII devices by using MBUS_XR or MBUS_XR1 function block ? (www.icpdass.com > FAQ > Software > ISaGRAF > 101)	
wpdmo_41	COM3 connecting 1:M7053D + 2:M7045D (MBRTU format, baud=9600) (Chapter 21 of the "ISaGRAF User's Manual")	
wpdmo_42	COM3 connecting 1:M-7053D to get D/I counter value (MBRTU format, baud=9600)	
wpdmo_43	COM3 connecting 1:M7017R + 2:M7024 (MBRTU format, baud=9600)	
wpdmo_44	COM3 connecting 1:M7017RC , Current input, +/- 20mA, 4-20mA (Modbus format)	
wpdmo_45	COM3 connecting 1:M-7019R (set as T/C K-type input) (MBRTU format, baud=9600)	
wpdmo_46	COM3 connecting 1:M7080 (MBRTU format, baud=9600)	
wpdmo_48	VB.net 2005 demo - "MBTCP_demo" (www.icpdass.com > FAQ > Software > ISaGRAF > 51)	
wpdmo_50	Non-linear conversion. like give P to find V (P , V relation listed in a file)	
wpdmo_51	Read 10 REAL value from a file,10 rows,each row has 1 REAL value, use str_real	
wpdmo_52	Msg_F. i8xx7 since 3.19. i7188EG/XG since 2.17/2.15. W8xx7 since 3.36, WP-8xx7	
wpdmo_53	Msg_N. i8xx7 since 3.19. i7188EG/XG since 2.17/2.15. W8xx7 since 3.36, WP-8xx7	
wpdmo_54	Read 20 REAL values from a file,4 rows,each row has 5 REAL values,uses msg_f (www.icpdass.com > FAQ > Software > ISaGRAF > 60)	
wpdmo_55	Read 20 Integers from a file,2 rows, each row has 10 Integers,uses msg_n	
wpdmo56	Retain 17 REAL value in a file, 2 rows, Each row has 10 REAL value	
wpdmo56a	Retain 2 Boo + 17 REAL in a file, 2 rows, Each row has 10 REAL value	
wpdmo56b	Retain 25 Integer in a file, 2 rows, Each row has 10 integer value	
wpdmo56c	Retain 2 Boo + 25 Integer in a file, 2 rows, Each row has 10 integer value (www.icpdass.com > FAQ > Software > ISaGRAF > 60)	
wpdmo56d	Retain 17 Real + 2 Boo + 10 Integer in 2 file, Each row has 10 value	

wpdmo56e	Retain more than 255 Real, 255 Boo,255 Integer in 2 file, up to 1024.	
wpdmo_61	i8xx7, WP8xx7: AutoReport data to PC via UDP.Controller=10.0.0.103,PC=10.0.0.91	
wpdmo_62	Send email via Ethernet port. (To one receiver without attached file) (www.icpdas.com > FAQ > Software > ISaGRAF > 67 , 71, 72, 76 or 77)	
wpdmo_63	For WP-8xx7 & W-8xx7 only. Send email to one receiver with one attached file (www.icpdas.com > FAQ > Software > ISaGRAF > 67 , 71, 72, 76 or 77)	
wpdmo64a	station 1001 , Time synchronization of many controllers via Ethernet.	
wpdmo64b	station 1002 , Time synchronization of many controllers via Ethernet.	
wpdmo65a	WP8xx7: Record temperature per minute to a file. Then send it by email per day (www.icpdas.com > FAQ > Software > ISaGRAF > 67 , 71, 72, 76 or 77)	slot 2: I-87018z
wpdmo65b	WP8xx7: Same as wdm_65a but add time synchronization and state report to PC (www.icpdas.com > FAQ > Software > ISaGRAF > 67 , 71, 72, 76 or 77)	slot 2: I-87018z
wpdmo_66	Record 1 to 4-Ch. i8017HW voltage pe 20ms, then send this record file by Email	slot 2: I-8024W slot 3: I-8017HW
wpdmo71a	WP-8xx7 COM4 connects I-7530 -- "CANopen" ID=1 device (8DI, 8DO, 4AO, 8AI) (www.icpdas.com > FAQ > Software > ISaGRAF > 86)	
wpdmo71b	Similiar as wdm_71A but connecting two I-7530. One is at COM5, one is at COM6	
wpdmo71c	WP8xx7 COM4 – 7530 -- CAN device to get string (with float or integer data inside)	
wpdmo71d	Similiar as wdm_71c but connecting two I-7530. One is at COM5, one is at COM6	
wpdmo71e	WP-8xx7: COM5 --- I-7530 --- CANopen device. COM6 --- I-7530 --- CAN device	
wpdmo72a	New WP-8xx7 redundant system with RU-87P4 + I-87K I/O (Without Touch HMI) (www.icpdas.com > FAQ > Software > ISaGRAF > 93)	
wpdmo72b	Same as wpdmo72a but setup COM1 as Modbus RTU slave port to connect one RS-232 Touch HMI (www.icpdas.com > FAQ > Software > ISaGRAF > 93)	
wpdmo72c	New WP-8xx7 redundant system with I-8KE8-MTCP I/O (Without Touch HMI)	
wpdmo74a	get average value of one REAL value (www.icpdas.com > FAQ > Software > ISaGRAF > 99)	
wpdmo74b	get average value of one Integer value (www.icpdas.com > FAQ > Software > ISaGRAF > 99)	

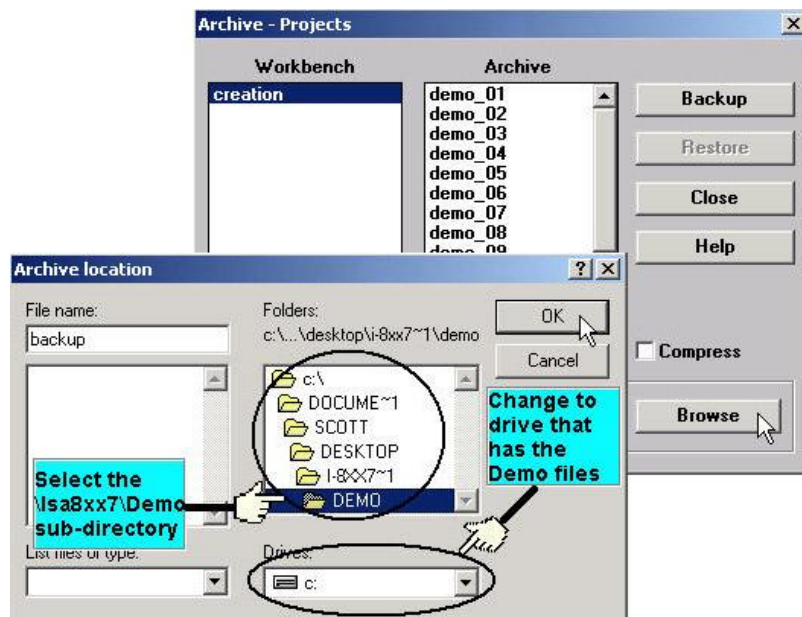
When you install the ISaGRAF programming example for the WinPAC controller it is recommended that you create an "ISaGRAF Project Group" to install the demo program files into.



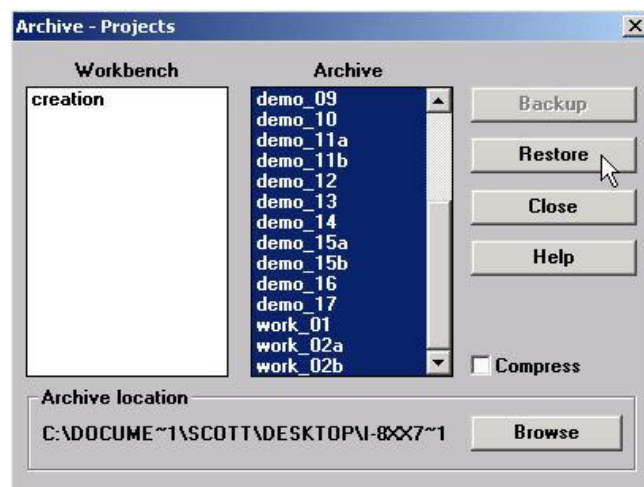
To install the demo programs into the project you have created open the "ISaGRAF Project Management" window to select "Tools" from the menu bar, then select the "Archive" option and then click on "Projects".



When you click on the "Projects" selection the "Archive Projects" window will open. Click on the "Browse" button to select the drive and the sub-directory where the demo files are located (**\\npdos\isagraf\wp-8xx7\demo** in the **WinPAC-8xx7 CD-ROM**) .



To install all of the Demo files, click on the "wdemo_01" file, then press and hold down the "Shift" key, continue to hold down the "Shift" key and use your mouse to scroll down to last file in the "Archive" window. Click on the last file name from the demo file location and that will select the entire group of demo files. Lastly, click on the "Restore" button in the "Archive Projects" window and all of the demo files will be installed into the sub-directory you have created.

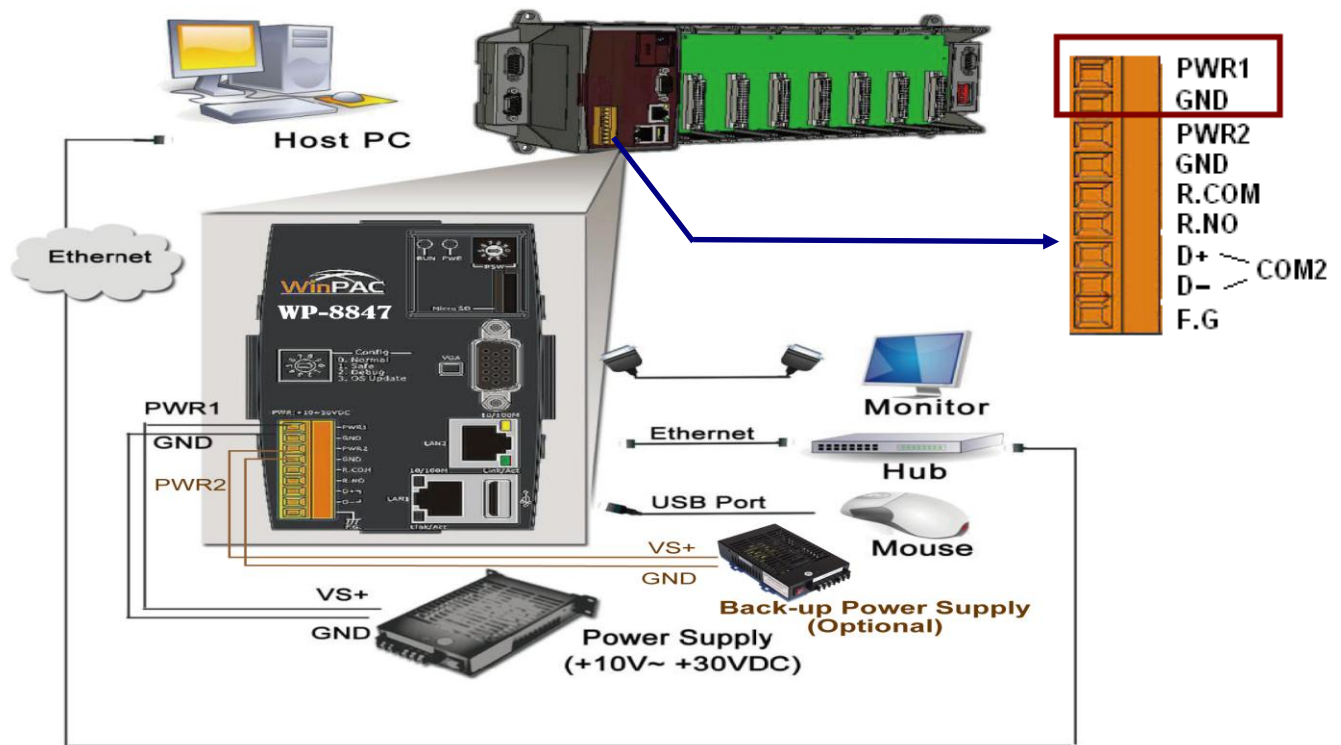


Appendix A: Hardware System & Setting

The WinPAC-8xx7 (or WP-8xx7) is the abbreviation of the WinPAC-8147 / 8447 / 8847.

A.1: Applying Correct Power Supply

Please apply a regular power supply between +10V to +30V (> 25W or higher is better)



Options:

Power supply: http://www.icpdas.com/products/Accessories/power_supply/power_list.htm

DP-660 : 24V/2.5A , 5V/0.5A power supply (DIN-Rail mounting)

DP-665 : 24V/2.5A , 5V/0.5A power supply

DP-1200 : 24V/5A power supply

Industrial Ethernet switch: http://www.icpdas.com/products/Switch/switch_list.htm

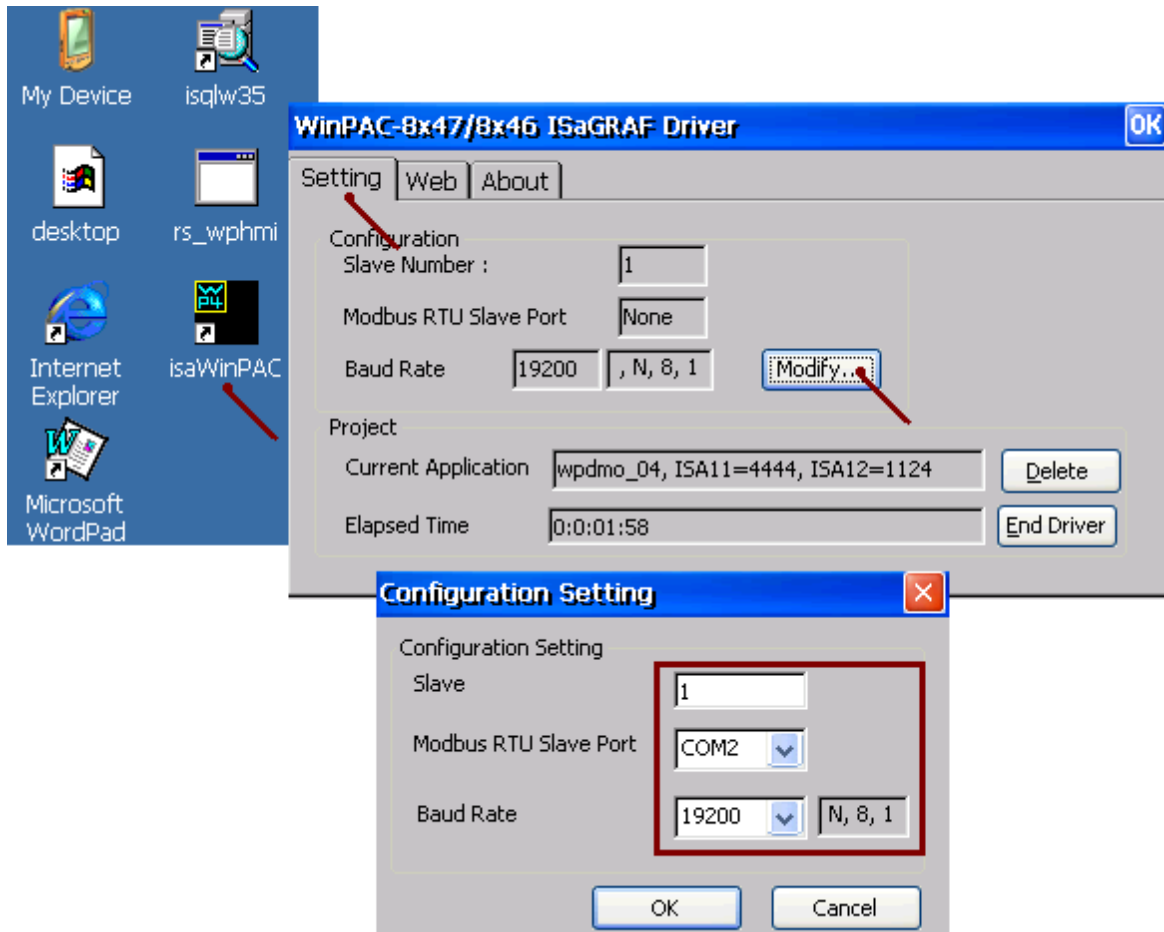
NS-205: 10/100M , 5 ports

NS-208: 10/100M , 8 ports

A.2: Modify The NET-ID & Modbus RTU Port Setting

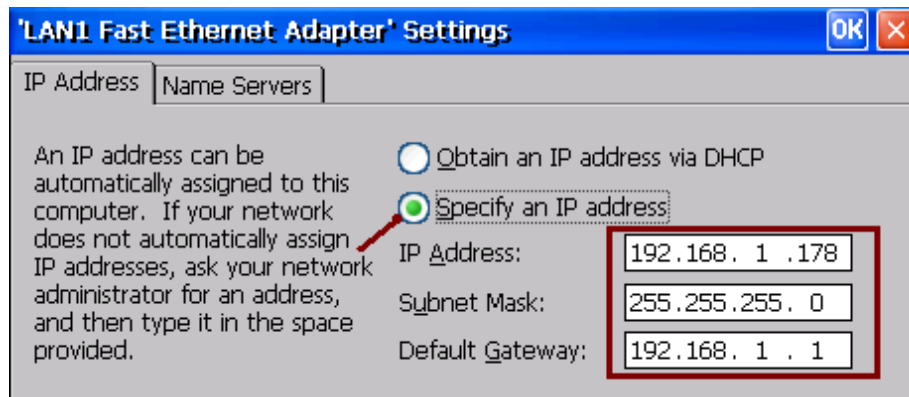
User may set WP-8xx7's Net-ID (Slave Number) to a No. from 1 to 255.

The default Modbus RTU slave port is "None" when shipped out. User may set it to others depends on its application (please also refer to appendix G & E for more Modbus RTU ports). Then please reset the WinPAC-8xx7 once after the modification to make the new setting work.



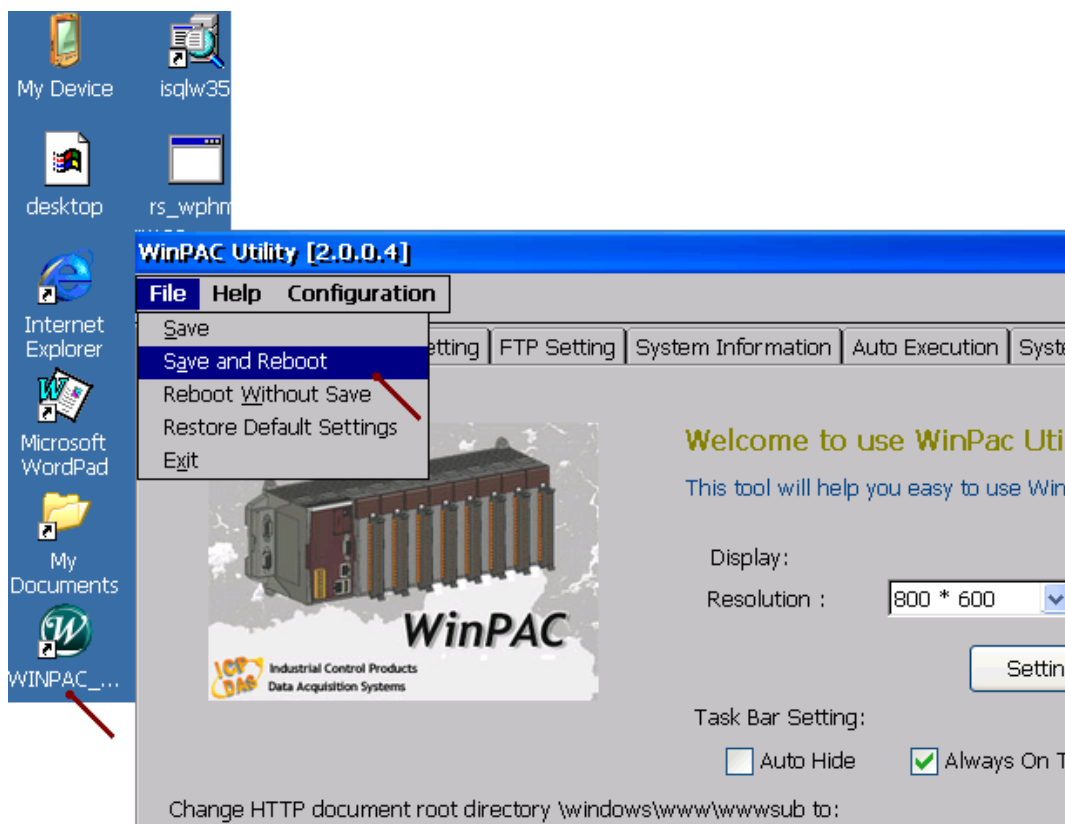
A.3: Setting The IP Address For The WP-8xx7

Please run “Start” – “Setting” – “Control Panel” on the WinPAC, then double click on “Network and Dial-up Connections”. Then click on “LAN1” and “LAN2”. Set your WinPAC’s IP address &



its Subnet Mask. (Please always set as Fixed IP for ISaGRAF application, No DHCP)

Please run “Start” – “Programs” – “WinPAC Utility”, click on “Save and Reboot” to store the setting.



A.4: Connecting Your PC To The WP-8xx7 Ethernet Port

Before you can download an ISaGRAF application to the WP-8xx7 controller using the Ethernet port, you must first setup the Ethernet port to properly communicate with the PC.

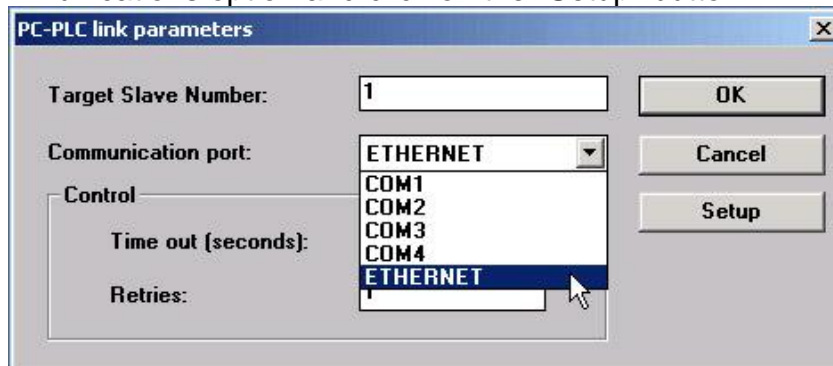
On the WP-8xx7, Set IP, Mask and Gateway address:

Please refer to former section – “A.3: Setting The IP Address For The WP-8xx7”

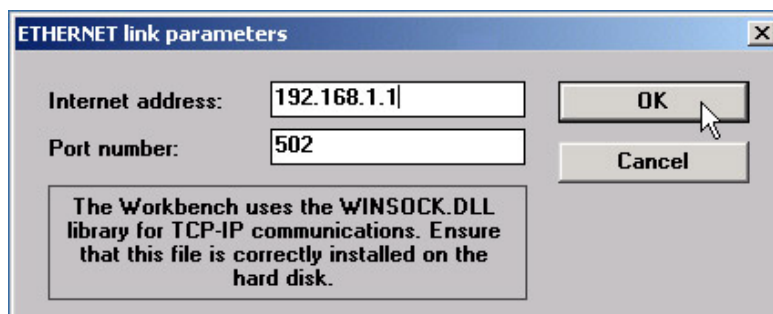
On your PC: First open an ISaGRAF project and select a program you wish to communicate between your PC and the WP-8xx7 controller system. Next, select the "Link Setup" button on the project screen as shown below.



A "PC-PLC Link Parameters" dialog box will appear as shown below. From here select the "Ethernet" communications option and click on the "Setup" button.



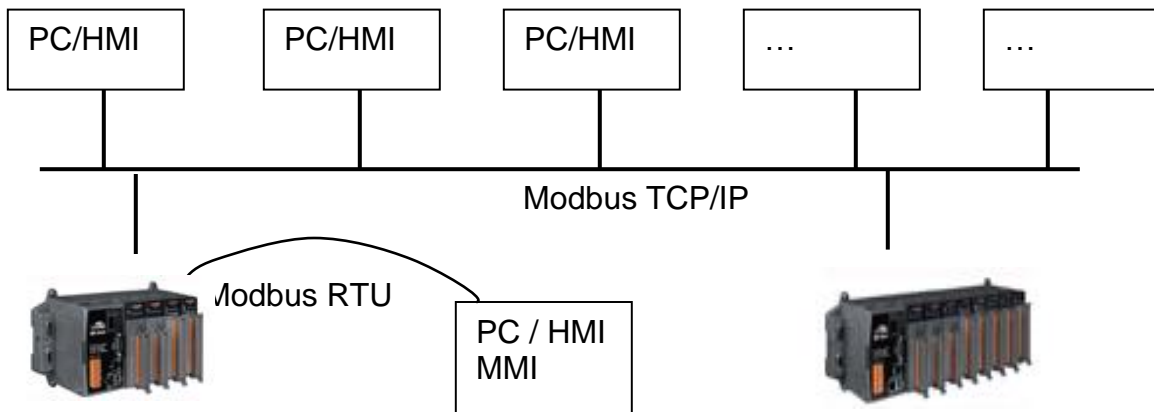
Once you have clicked on the "Setup" button, an "Ethernet Link Parameters" dialog box will appear. Set the "Port Number" to "502" and enter in the **Internet address (IP) of the WP-8xx7 controller**.



Once you have entered the appropriate information, click on the "OK" button, and now you have configured your PC to communicate with the WP-8xx7 through the Ethernet port.

A.5: Pin Assignment of COM1, COM2, COM3 and COM4 and Multi-Clients Connection to The WP-8xx7

Each WP-8xx7 has an IP address and with a fixed Ethernet port No. **502**. Up to 32 PCs can link to one WP-8xx7 throughout Ethernet (Modbus TCP/IP protocol, one TCP/IP connection for each PC). Other PC or HMI can link to COM2: RS-485 port or COM2:RS-485 (or COM1,4, ... appendix G & E) (Modbus RTU slave)

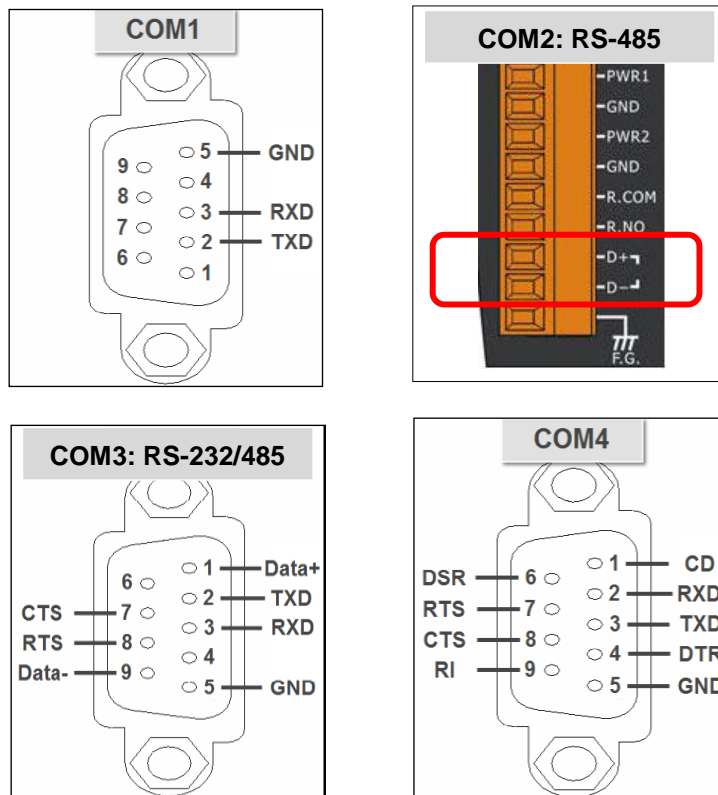


Options: Industrial Ethernet switch: http://www.icpdas.com/products/Switch/switch_list.htm

NS-205: 10/100M , 5 ports

NS-208: 10/100M , 8 ports

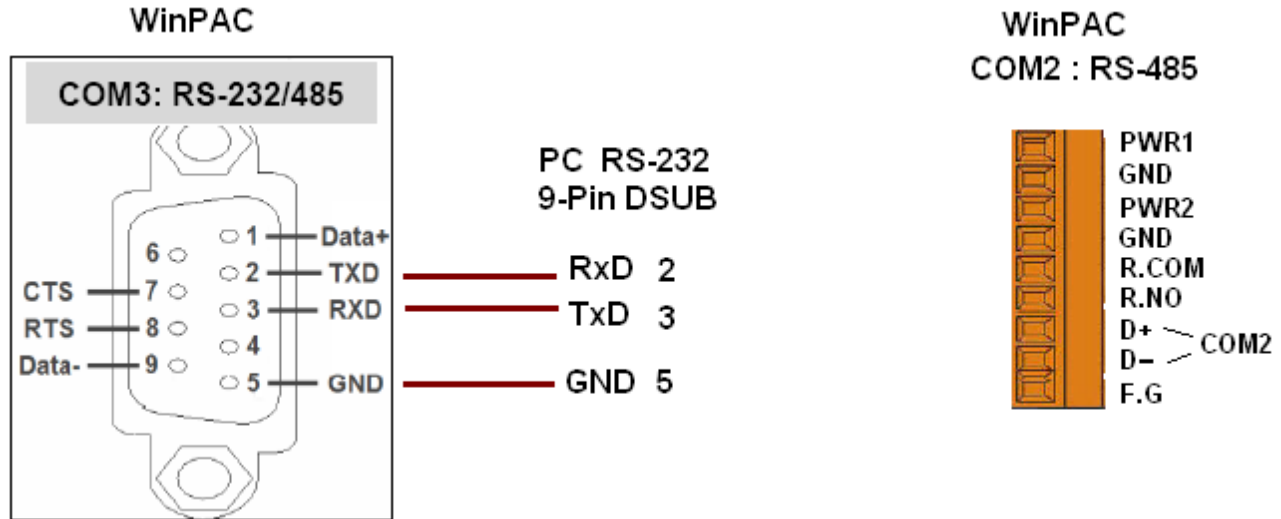
Pin Assignment of COM1 , COM2, COM3 and COM4:



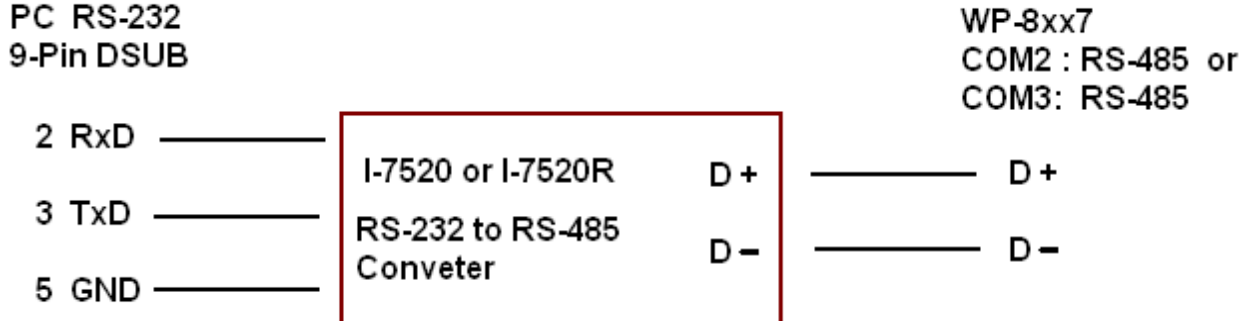
A.6: Connecting PC To WP-8xx7 COM2 or COM3

The default Modbus RTU slave port is "None". User may change it to "COM2:RS-485" or "COM3:RS-232 / 485" or "None". (Please refer to "A.2: Modify The NET-ID & Modbus RTU port setting"). Please refer to appendix G & E for more Modbus RTU ports.

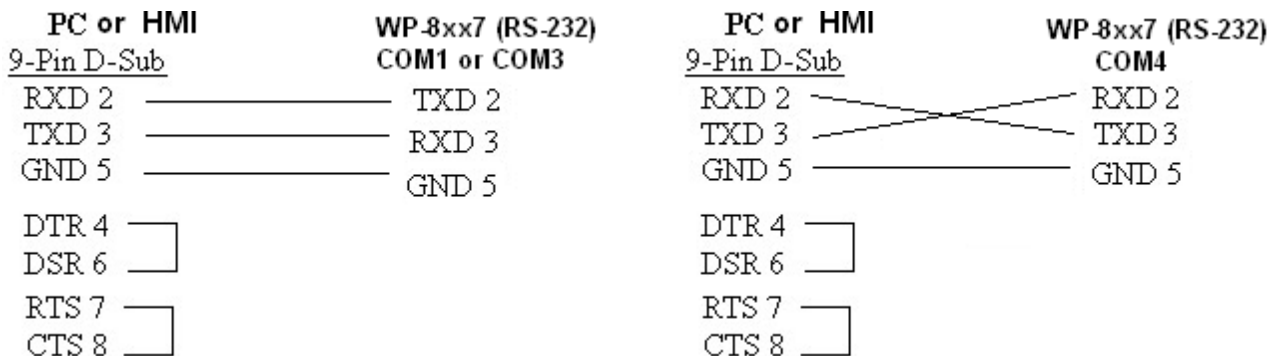
Default communication parameter is "19200,8,N,1"



If connecting PC to WinPAC COM2 or COM3's RS-485, a I-7520 (RS-232/485 converter) is necessary as below.



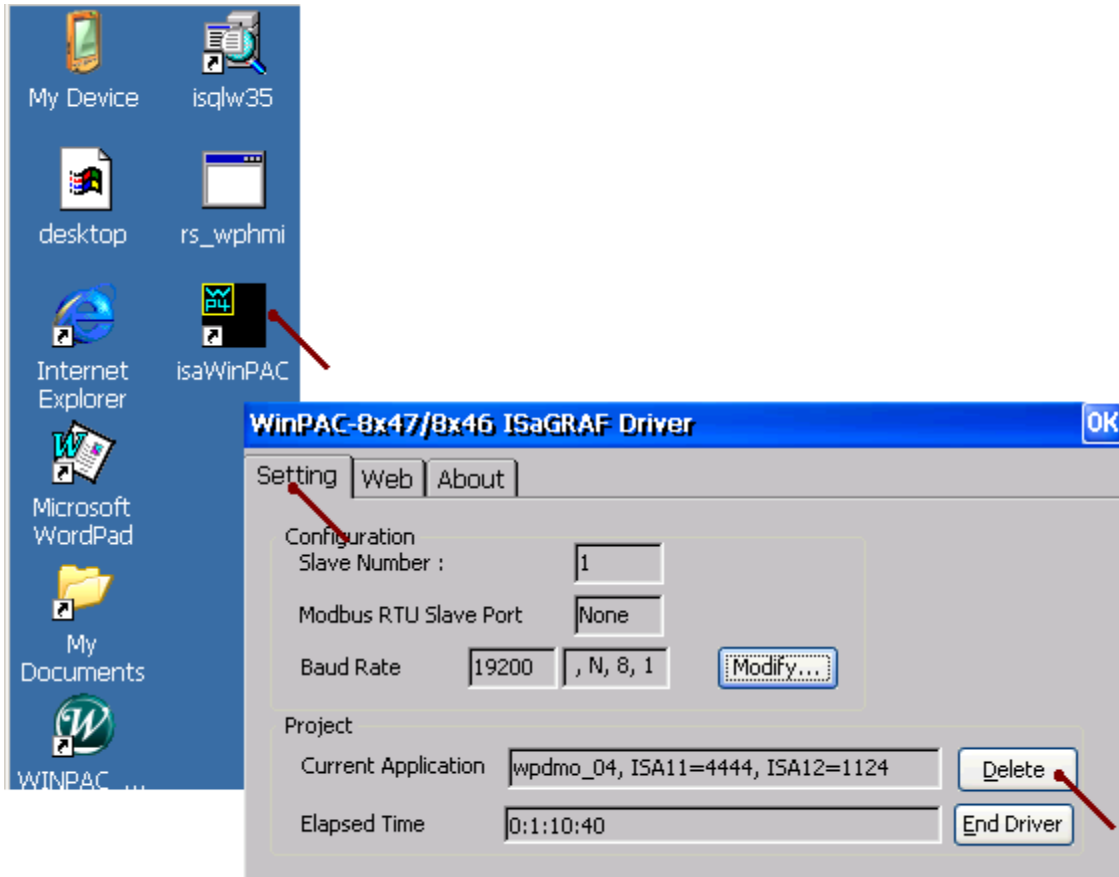
For the ISaGRAF Workbench RS-232 communications to operate properly, only the RXD, TXD, and the GND signals are used. If your PC is running a hardware device or software program that uses the CTS and DSR signals, you will need to wire the RTS-CTS and DTR-DSR signals together as shown below.



A.7: Deleting the ISaGRAF Project From The WP-8xx7

For some reasons, user may delete the ISaGRAF program in the WinPAC-8xx7 controller.

Click on “Setting” & then click on “Delete ISaGRAF Project”.



Delete WinPAC-8xx7 's ISaGRAF program if some software damage happens causing the WinCE software hanging.

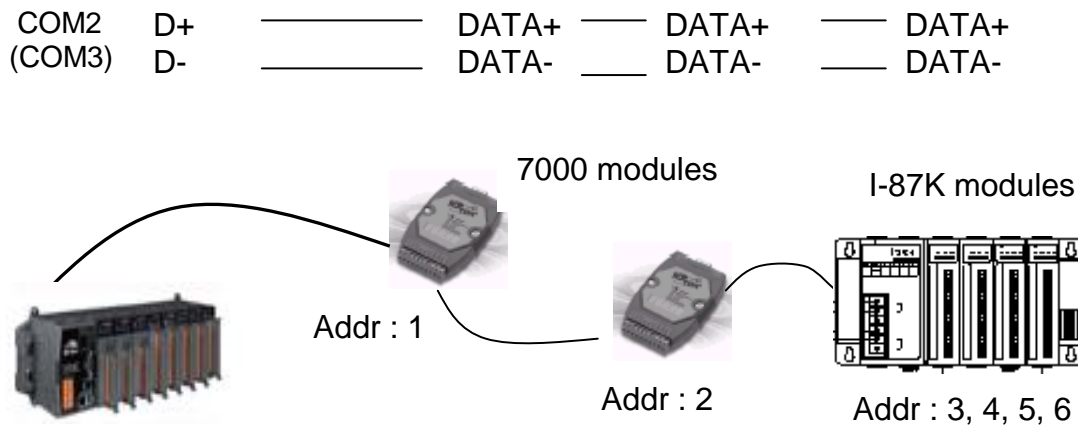
1. Please turn the rotary switch to position 1 (Safe mode) on the front panel of the WinPAC-8xx7 . Then reset the WinPAC-8xx7 once.
2. Then the WinPAC-8xx7 will boot up as safe mode. There will be one pop-up window asking "... reboot right now ...", please answer "No". Then get into the "My Device" on the WinCE desktop. Please goto the "\System_Disk\isagraf" directory . Then delete the "ISA11" . The "ISA11" is the ISaGRAF current running application. (If you find no "ISA11" in the \System_Disk\isagraf\ directory, please goto Explorer > View > Options to modify the setting)
3. Turn the rotary switch to position 0 (normal), then reboot WinPAC-8xx7. Then when ISaGRAF is connected, it will display "No Application" .

A.8: Linking I-7000 and I-87K Modules For Remote I/O

The WP-8xx7 controller system can use one of its COM2 or COM3 's RS-485 signal to link to ICP DAS's "I-7000" and "I-87K" series of remote I/O modules. This configuration can be very useful in applications that require distributed remote I/O throughout the system.

You can link up to **255** I-7000 or I-87K series remote modules to one WP-8xx7 controller system (It is better not to link up to 40 pcs. of I-7000 or I-87K). You must remember to set each I-7000 and I-87K remote module must have a unique address, and be set to the same baud rate as the WP-8xx7 controller system.

For more information regarding setting up and programming an I-7000 / I-87K remote module, please refer to Chapter 6 - "Linking To I-7000 and I-87K Modules" of the "User's Manual Of The ISaGRAF Embedded Controller" .

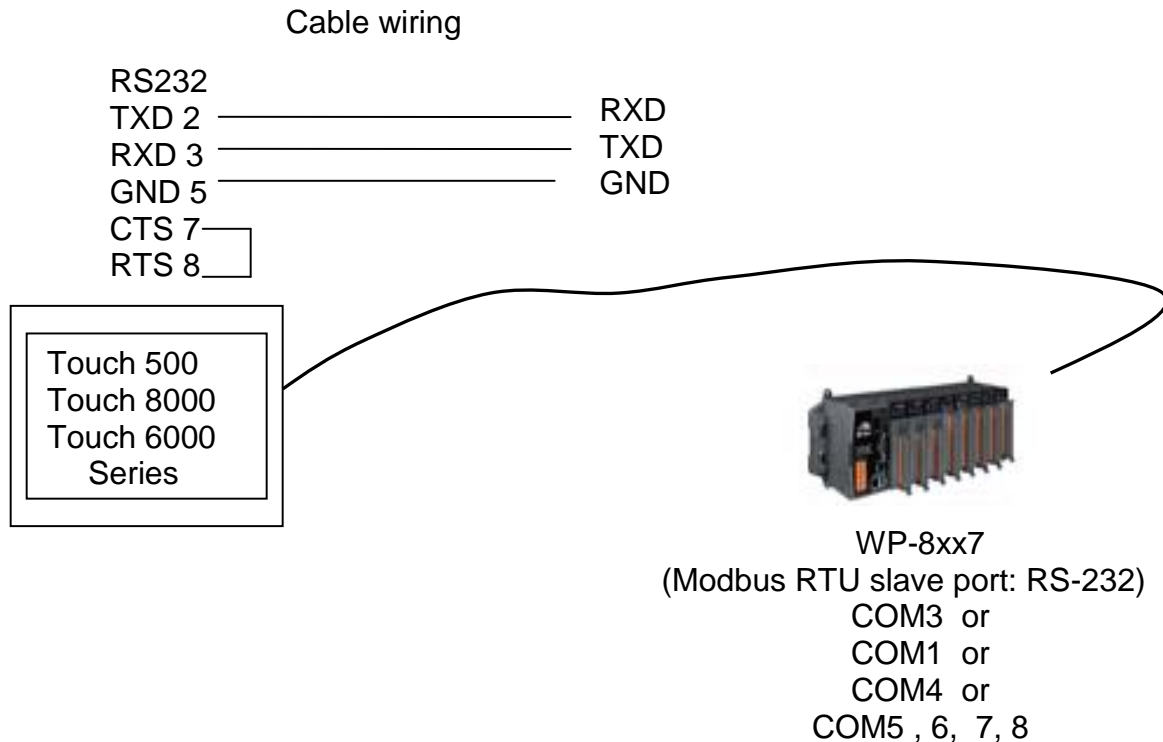


A.9: Linking To An HMI Interface Device

One of the COM2 or COM3 (RS-232 or RS-485) (or max. four of the COM1, 4, 5, 6, 7, 8, please refer to appendix G & E) ports of the WP-8xx7 / 8xx6 controller system can be used to interface with additional Human Machine Interface (HMI) devices such as touch displays.

Please refer to section A.2 first for setting Modbus RTU port at one of COM2 or COM3. ICP DAS provides a full line of touch screen displays, such as the "Touch" series screens. The models in the product line include the Touch 506, and Touch 510 HMI or other Touch 8000 series products.

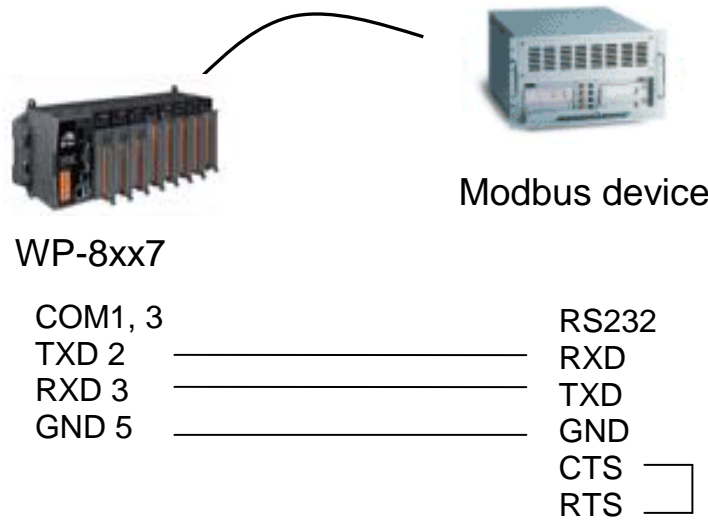
For more information regarding interfacing the Touch series of MMI devices to the WP-8xx7 / 8xx6 controller system, please refer to Chapter 4- "Linking The I-8xx7 To HMI Devices" of the "User's Manual Of The ISaGRAF Embedded Controller" ..



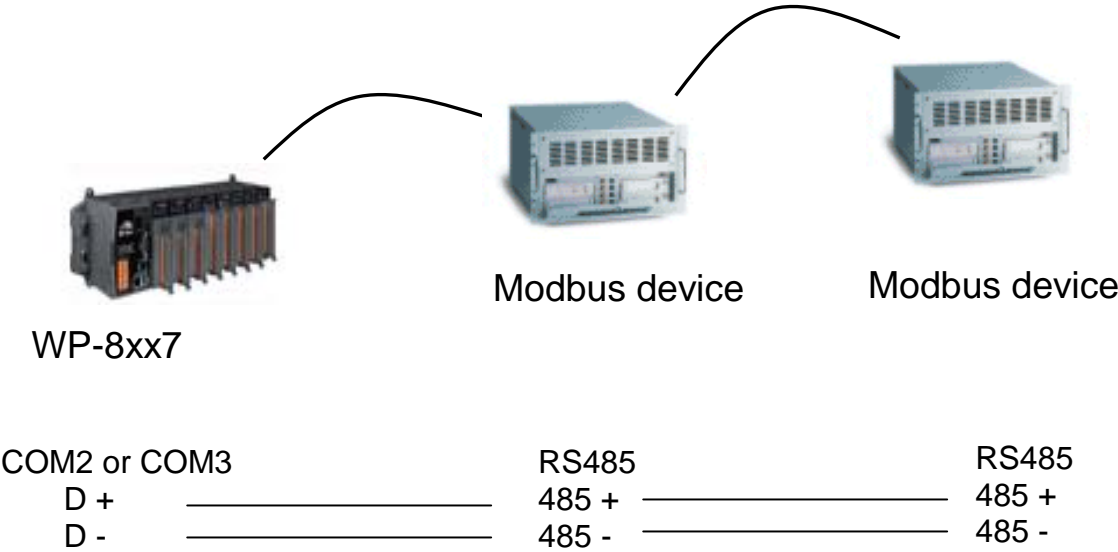
A.10: Linking To Other Modbus Devices

The COM2 (RS-485) or COM3 (RS-232 / 485) (or COM1, 4, COM5 to 14, refer to appendix E) supports Modbus Master protocol. Please refer to Chapter 8 of the “User’s Manual Of The ISaGRAF Embedded Controllers” for more information.

RS-232:



RS-485:



Appendix B: Upgrade WinPAC's ISaGRAF Driver to Newer Version

Note:

If you have purchased WP-8xx7, the ISaGRAF Driver is already installed with license when shipping out. You don't need to install it. However if you want to upgrade to newer version, you may upgrade it by yourself.

The WinPAC ISaGRAF driver can be obtained in the WinPAC-8xx7 CD-ROM:

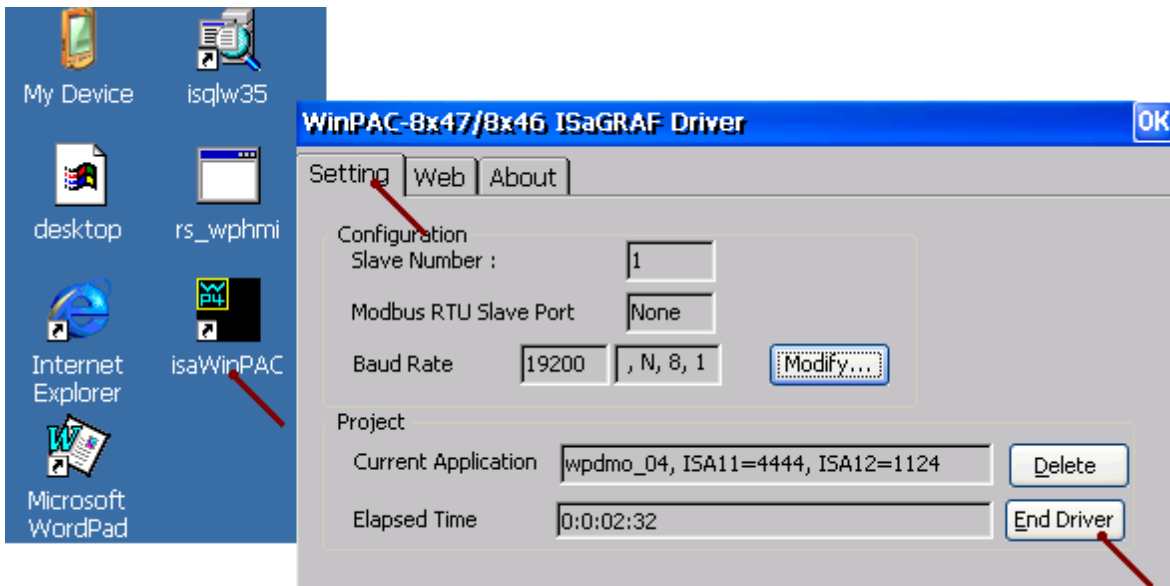
\napdos\isagraf\wp-8xx7\driver\wp-8x47\<version Number>\

For example, version 1.01 is located at

\napdos\isagraf\wp-8xx7\driver\wp-8x47\1.01\

Or download it from <http://www.icpdas.com/products/PAC/i-8000/isagraf.htm> > Driver

1. If your WinPAC is WP-8xx7 / WP-8xx6, please stop "WinPAC ISaGRAF Driver" first. (Click on "End Driver" to stop it.)

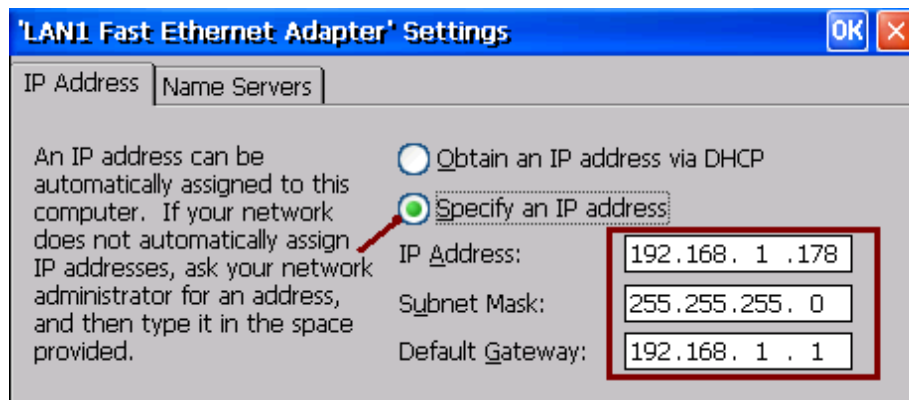


However if it is WP-8xx1/8xx9 (WinPAC without ISaGRAF license), please goto step 2.

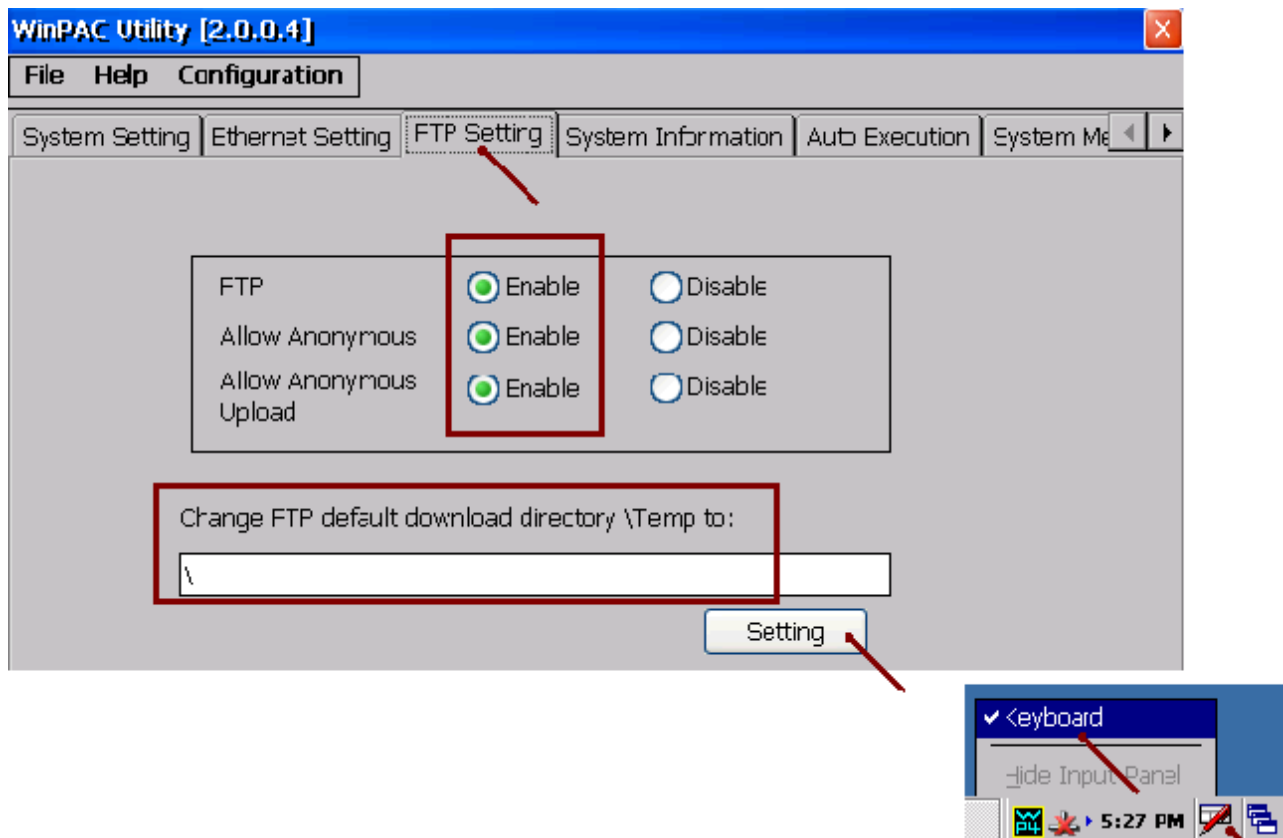
2. Set up WinPAC 's IP, Mask, FTP directory & Auto-execute

- A. Please create a folder "isagraf" inside "\System_Disk" folder in your WinPAC controller. Then it will be \System_Disk\isagraf\
- B. Please run "Start" – "Setting" – "Control Panel" on the WinPAC, then double click on "Network and Dial-up Connections". Then click on "LAN1" and "LAN2". Set your

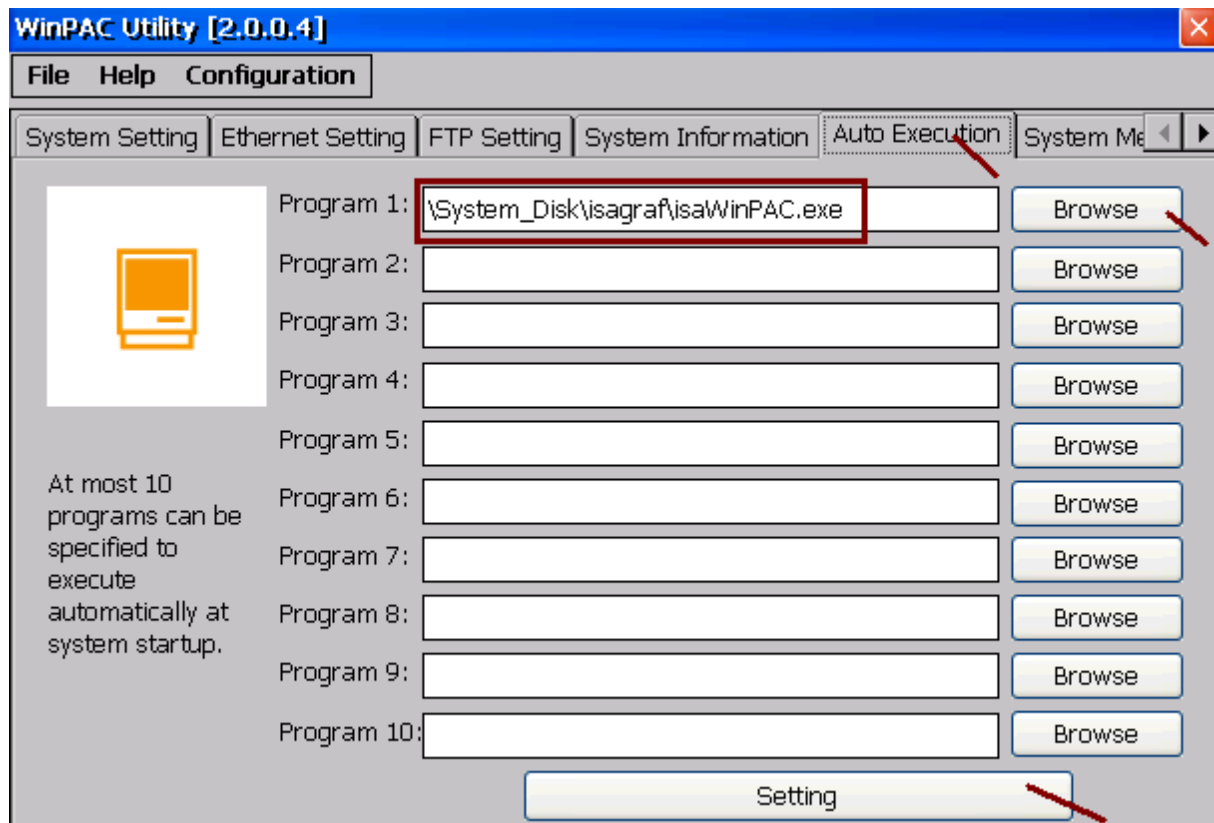
WinPAC's IP address & its Subnet Mask. (Please always set as Fixed IP for ISaGRAF application, No DHCP)



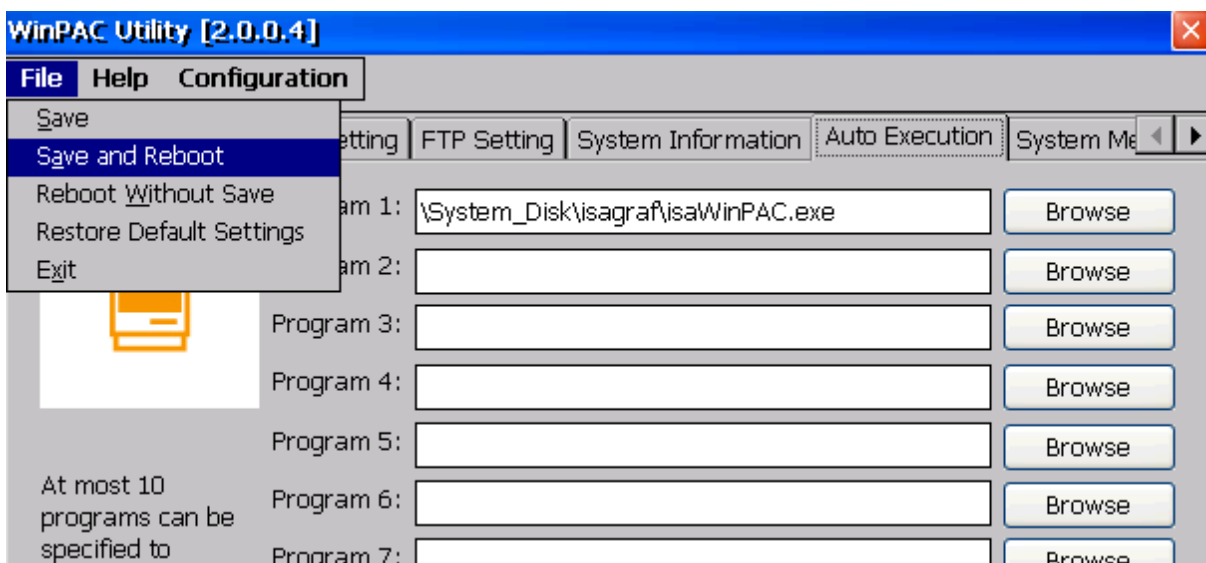
- C. Please run "Start" – "Programs" – "WinPAC Utility". Set FTP directory to the root directory "\". Then check all three ftp options as "Enable". Remember to click on "Setting". Then click on "Auto Execution" to do the next step



- D. Please click on "Browse" to select or type "\System_Disk\isagraf\isaWinPAC.exe" and click on "Setting"

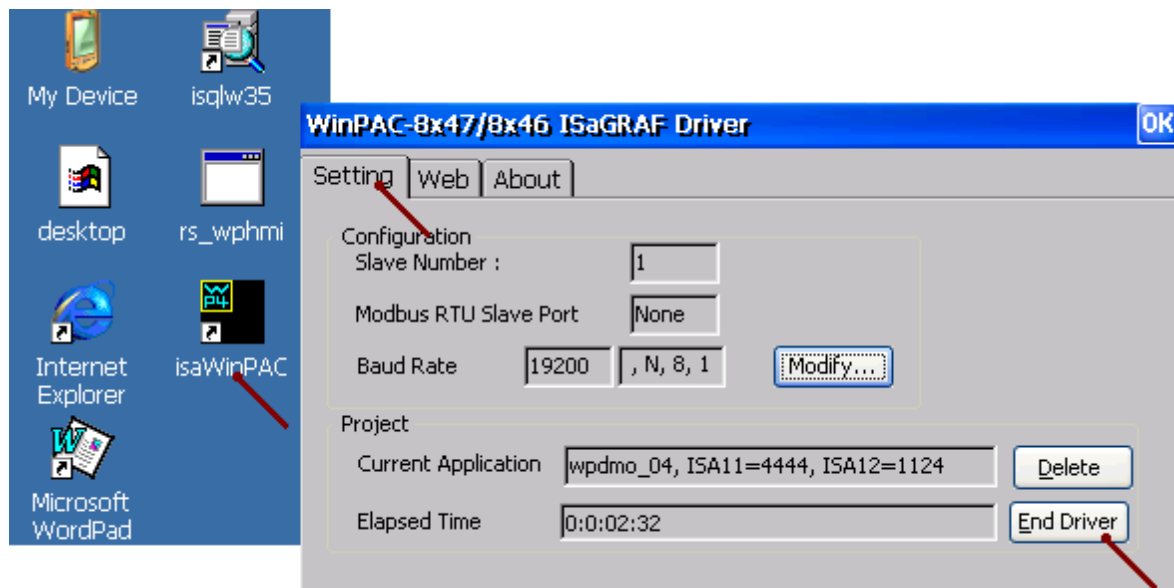


E. Run “Save and Reboot” to store the setting in step A thru. D and then it will auto-reboot the WinPAC once.



3. After the WinPAC reboot successfully, please stop the ISaGRAF driver again . (The original WP-8xx1 / 8xx9 doesn't have the ISaGRAF driver running, only the WP-8xx7 / 8xx6 have it)

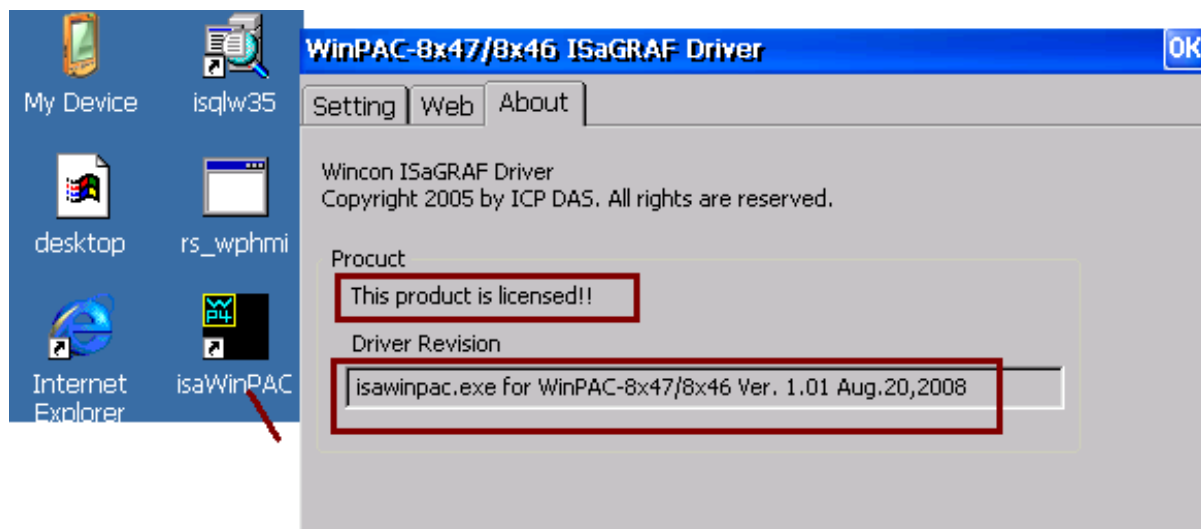
Note: If the ISaGRAF driver is still running, the files copied are failed even your eye tell you it is successful.



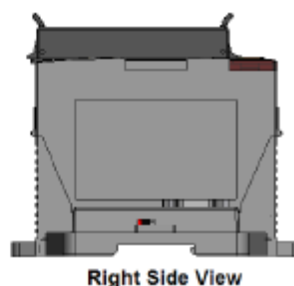
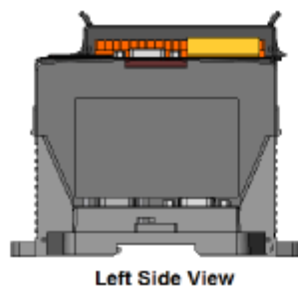
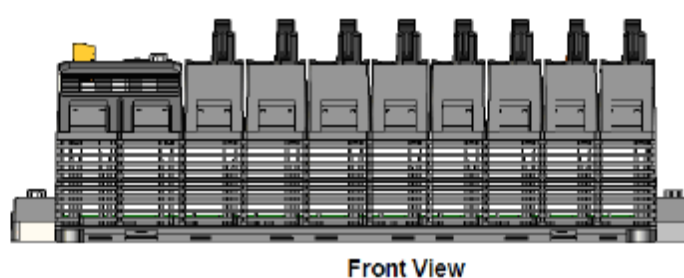
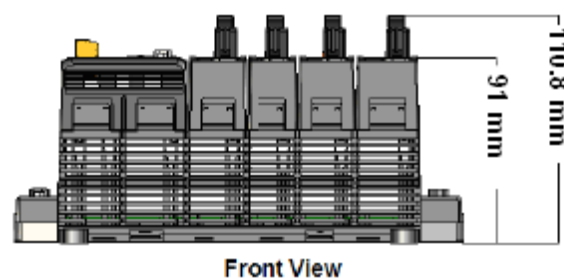
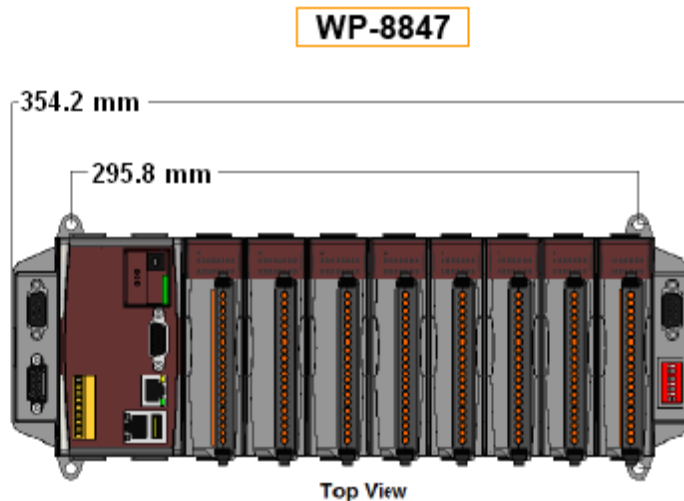
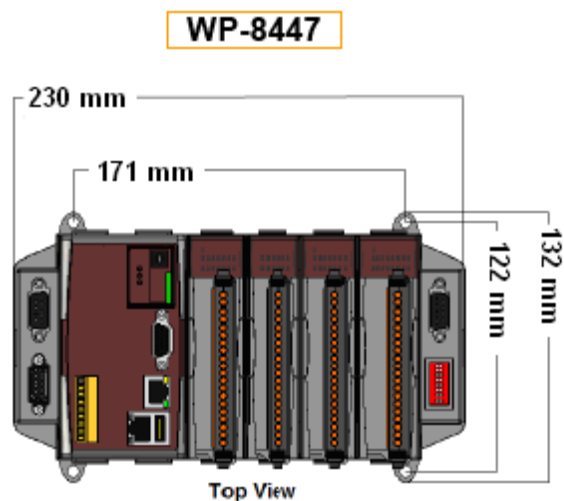
Then download the “isaWinPAC.exe”, “rs_wphmi.exe”, “mscorlib.dll”, “QuickerNet.dll”, “Quicker.dll”, “login.dll”, “main.dll”, “whmi_filter.dll” and “isaWinPAC.lnk” (and “license.bin” if your WinPAC is WP-8xx1 / 8xx9) from the PC to the WinPAC controller. Please copy them to the “\System_Disk\isagraf\”. & then re-cycle your WinPAC’s power. You may use PC’s ftp utility to download these files.

Please open Internet browser and then type in <ftp://<IP address>>, for ex. [Ftp://192.168.1.178](ftp://192.168.1.178), browse it to the \System_Disk\isagraf\ . Then copy all of them & past it.

Then remember to re-cycle your WinPAC's power again. After it re-boot again, it will have the new ISaGRAF driver running. You can check if the version is correct.



Appendix C: Dimension



Appendix D: How to Enable/Disable WP-8xx7's LAN2

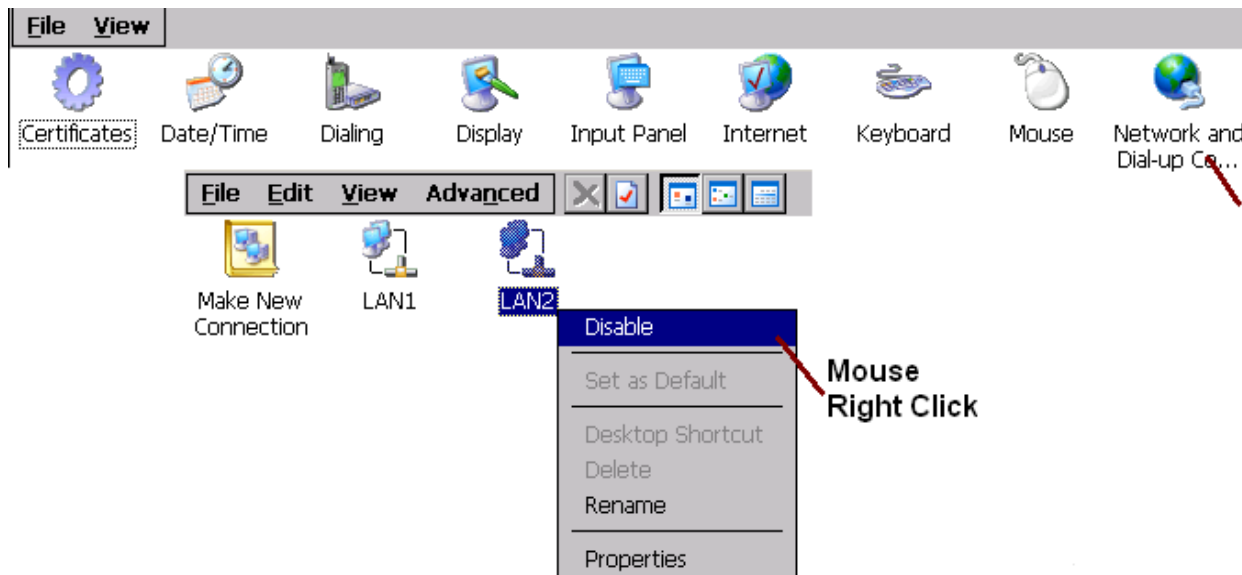
Important:

1. Recommend to use NS-205 or NS-208 Industrial Ethernet Switch for WinPAC-8xx7/8xx6.
2. Please always set a fixed IP to LAN1 (and LAN2 if it is enabled) for ISaGRAF applications.

The default setting of LAN2 of WP-8xx7 is disabled. User must enable it before using LAN2 port.

ISaGRAF **must** use WP-8xx7's LAN2 when using "Ebus" (section 7.5 of the ISaGRAF User's Manual) and "New Redundant system" (please refer to www.icpdas.com > FAQ > Software > ISaGRAF > 093). ISaGRAF **may** use LAN2 when using "Delivering message via UDP or TCP" (section 19.2 and 19.3 of the ISaGRAF User's Manual).

Please open "Start" – "Setting" - "Control Panel" and then click on "Network and Dial-up Connections" to set as LAN2 as Enable (or Disable)



Then run "Start" – "Programs" – "WinPAC Utility", click "Save and Reboot" to save the setting.

Appendix E: Using Expansion RS-232 or 485 or 422

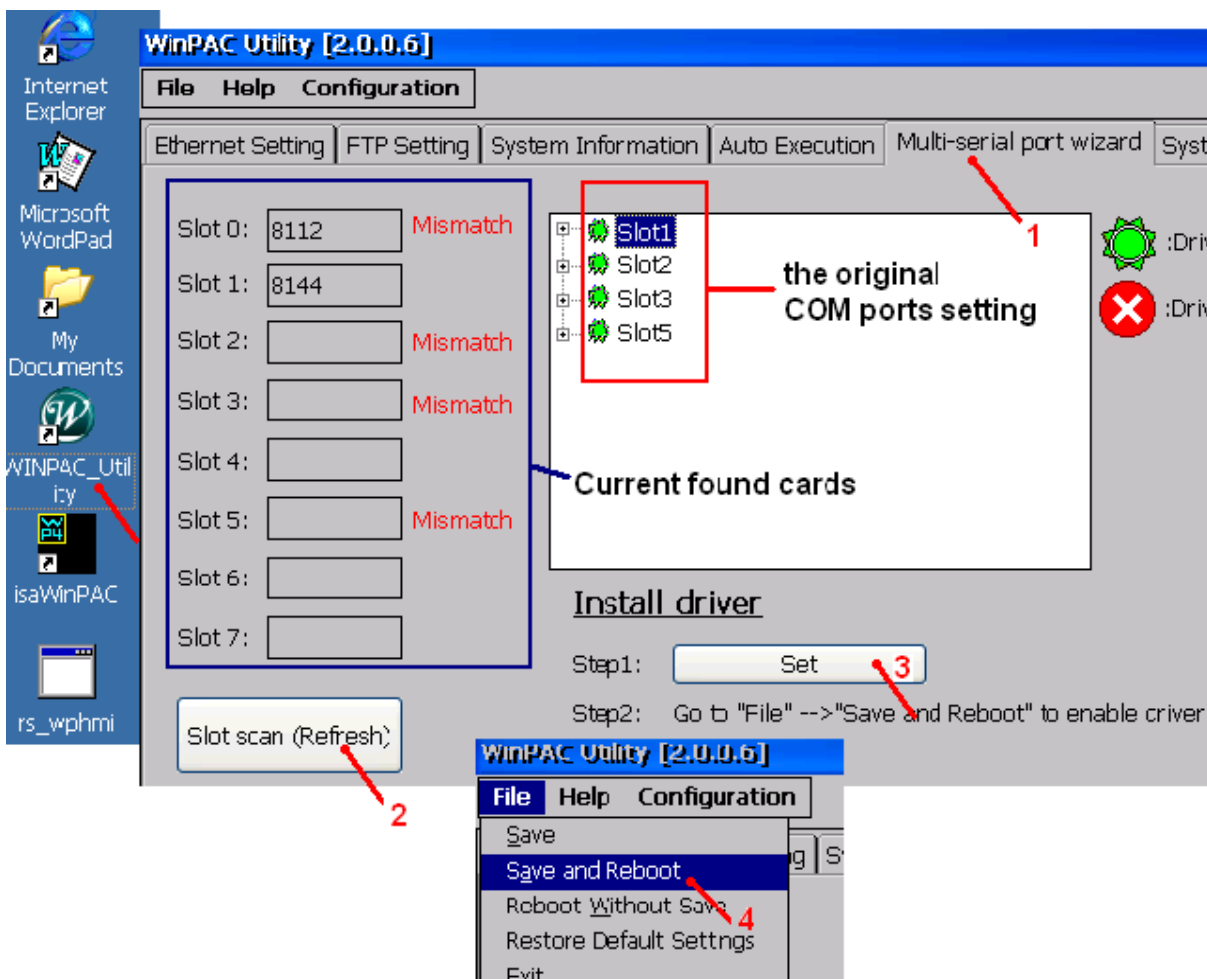
The WinPAC can expand 16 more COM ports in its slot No. 0 to 7 by using following modules.

- I-8112iW : 2-channel isolated RS-232
- I-8114iW : 4-channel isolated RS-232
- I-8114W : 4-channel non-isolated RS-232
- I-8142iW : 2-channel isolated RS-422/RS-485
- I-8144iW : 4-channel isolated RS-422/RS-485

Before user can use them, please configure them by the “WinPAC utility” (since version 2.0.0.6) first.

Please plug them in the WinPAC's slot 0 to 7 (It is better to be in slot 0 to 3) and then run “WinPAC utility” > “Multi-serial port wizard”, then click on “Slot scan”. The current found multi-serial port cards will be listed on the left. The original COM port setting is listed on the right.

Then click “Set” to update the original setting to become the current found cards. Then remember to run “File” > “Save and Reboot” to save the new setting and then WinPAC will re-boot itself once.

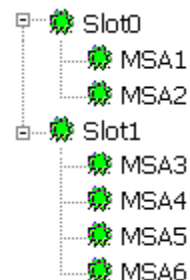


After the configuration succeeds, the COM port No. for the expansion board is COM5 to COM20 in the ISaGRAF definition.

The relation between WinPAC's COM setting and the ISaGRAF definition is as the following.

WinPAC	ISaGRAF	WinPAC	ISaGRAF
MSA1	COM5	MSB1	COM13
MSA2	COM6	MSB2	COM14
MSA3	COM7	MSB3	COM15
MSA4	COM8	MSB4	COM16
MSA5	COM9	MSB5	COM17
MSA6	COM10	MSB6	COM18
MSA7	COM11	MSB7	COM19
MSA8	COM12	MSB8	COM20

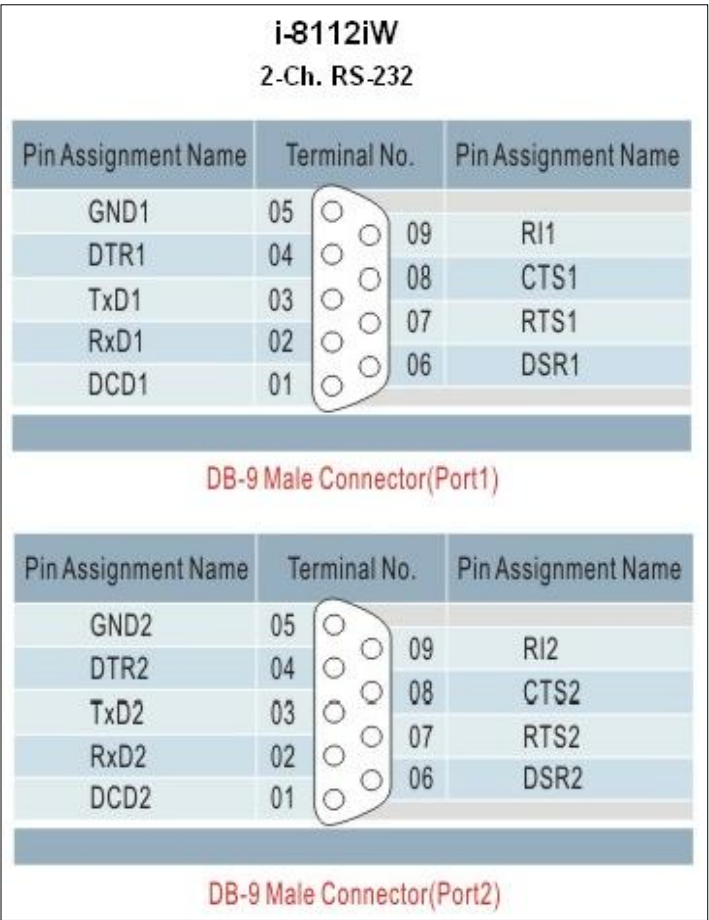
Example of slot 0: I-8112iW
and slot 1: I-8144iW



Note:

1. Please refer to the section 8.4 of the ISaGRAF User's Manual for multi-ports Modbus Master. (WinPAC-8xx7 can setup max. 10 Modbus RTU / ASCII Master ports in COM1 thru. 14)
2. Please refer to the Appendix A.4 of the ISaGRAF User's Manual for COM_OPEN, COM_READ, ... functions to read write COM ports.
3. Please refer to Appendix G of this manual for setting up more Modbus RTU slave ports.

Pin assignment:



i-8114W / i-8114iW

4-Ch. RS-232

Pin Assignment Name	Terminal No.		Pin Assignment Name
N.C.	01	○	
DCD3	02	○	20 RI3
GND	03	○	21 DTR3
CTS3	04	○	22 DSR3
RxD3	05	○	23 RTS3
RI4	06	○	24 TxD3
DTR4	07	○	25 DCD4
DSR4	08	○	26 GND
RTS4	09	○	27 CTS4
TxD4	10	○	28 RxD4
DCD2	11	○	29 RI2
GND	12	○	30 DTR2
CTS2	13	○	31 DSR2
RxD2	14	○	32 RTS2
RI1	15	○	33 TxD2
DTR1	16	○	34 DCD1
DSR1	17	○	35 GND
RTS1	18	○	36 CTS1
TxD1	19	○	37 RxD1

37-Pin Female D-Sub Connector(Port1~Port4)

i-8142iW





















2-Ch. RS-422 / RS-485

RS-485 Ch.1 = (D1+ , D1-)

RS-485 Ch.2 = (D2+ , D2-)

RS-422 Ch.1 = (TxD1+ , TxD1- , RxD1+ , RxD1-)

RS-422 Ch.2 = (TxD2+ , TxD2- , RxD2+ , RxD2-)

Terminal No.	Pin Assignment Name
 01	D1+/TxD1+
 02	D1-/TxD1-
 03	RxD1+
 04	RxD1-
 05	GND1
 06	D2+/TxD2+
 07	D2-/TxD2-
 08	RxD2+
 09	RxD2-
 10	GND2
 11	N.C.
 12	N.C.
 13	N.C.
 14	N.C.
 15	N.C.
 16	N.C.
 17	N.C.
 18	N.C.
 19	N.C.
 20	N.C.

i-8144iW

4-Ch. RS-422 / RS-485

RS-485 Ch.1 = (D1+ , D1-)

RS-485 Ch.2 = (D2+ , D2-)

RS-485 Ch.3 = (D3+ , D3-)





















RS-485 Ch.4 = (D4+ , D4-)

RS-422 Ch.1 = (TxD1+ , TxD1- , RxD1+ , RxD1-)

RS-422 Ch.2 = (TxD2+ , TxD2- , RxD2+ , RxD2-)

RS-422 Ch.3 = (TxD3+ , TxD3- , RxD3+ , RxD3-)

RS-422 Ch.4 = (TxD4+ , TxD4- , RxD4+ , RxD4-)

Terminal No.	Pin Assignment Name
 01	D1+/TxD1+
 02	D1-/TxD1-
 03	RxD1+
 04	RxD1-
 05	GND1
 06	D2+/TxD2+
 07	D2-/TxD2-
 08	RxD2+
 09	RxD2-
 10	GND2
 11	D3+/TxD3+
 12	D3-/TxD3-
 13	RxD3+
 14	RxD3-
 15	GND3
 16	D4+/TxD4+
 17	D4-/TxD4-
 18	RxD4+
 19	RxD4-
 20	GND4

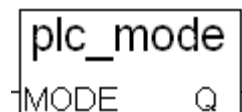
Appendix F: Slow Down ISaGRAF Driver's Speed

You may wonder why? The faster speed is not good?

The reason to slow down the speed of ISaGRAF driver is when you running some other HMI program (For example, InduSoft, or VB.net program) with ISaGRAF at the same time. Because the CPU is the only one CPU, all programs running in WinPAC must share execution time of the same CPU. If you feel the HMI program behavior is not so smooth, or slow, you may use ISaGRAF function – "PLC_Mode()" to slow down the speed of the ISaGRAF driver.

PLC_Mode

Function	Description:
	Change the ISaGRAF driver speed



Argument:

MODE_	integer	Can be 0 , 1, 2, or 3
--------------	---------	-----------------------

- 0: Fast Mode, Default setting, the minimum PLC scan time is about 2 ~ 3 ms
- 1: Slow Mode, the minimum PLC scan time is about 6 to 7 ms
- 2: Slower Mode, the minimum PLC scan time is about 9 to 11 ms
- 3 or other value: Slowest Mode, the min. PLC scan time is about 19 to 21 ms

Return:

Q_	boolean	always return True
-----------	---------	--------------------

Note:

1. The system's default setting is "Fast Mode"
2. User may call "PLC_mode()" in the first PLC scan to change the PLC speed.
3. The reason to slow down the PLC speed is to improve the speed performance of other HMI program running with ISaGRAF driver at the same time, for example, running InduSoft with ISaGRAF in the same WinPAC.

Example:

```
(* TMP is declared as Boolean internal variable *)
(* INIT is declared as Boolean internal variable and init at TRUE *)
if INIT then
  INIT := False ; (* Only do it once in the 1st PLC scan *)
  TMP := PLC_mode(2) ; (* Set PLC speed to 2:slower mode *)
end_if ;
```

Appendix G: Setup More Modbus RTU Slave Ports

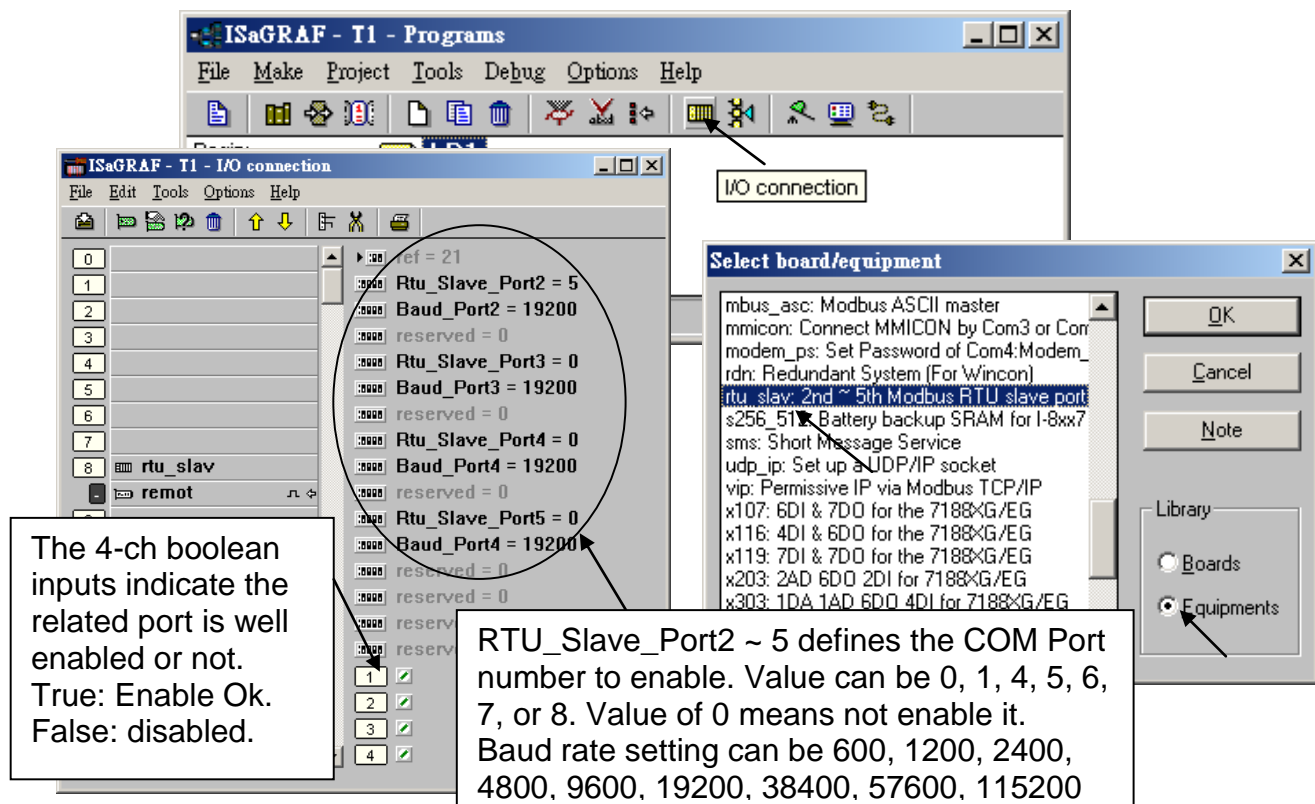
The WinPAC-8xx7/8xx6 can setup up to five Modbus RTU slave ports in one of the COM2 or COM3 and in four of the COM1, COM4, COM5, COM6, COM7, COM8 (expansion multi-serial ports in slot 0 to 3, refer to the appendix E) .

Note: WP-8147 doesn't have COM3 and COM4. (only WP-8447/8847 have them)

1. The first Modbus RTU slave port can be one of the COM2 or COM3 which can be set on the "WinPAC's monitor" by mouse (refer to the appendix A.2).
2. User may enable 2nd, 3rd, 4th or 5th Modbus RTU slave port in COM1, COM4, COM5, COM6, COM7 or COM8 only. (No support other COM port number)
3. Before using this function, please make sure COM4, (or COM5, COM6, COM7, COM8) does exist and well configured. (refer to the appendix E)
4. Via 2nd, 3rd, 4th or 5th Modbus RTU slave port, user may use ISaGRAF to Debug/Set_val to the controller, however user can not Stop/Download/Update the ISaGRAF program.
5. To Debug/Set_val/Stop/Download/Update the ISaGRAF program, please use Ethernet port (or the first Modbus RTU slave port, that is COM2 or COM3 if enabled). COM1, COM4 and COM5 to COM8 are not for ISaGRAF to Stop/Download/Debug.

How to setup ?

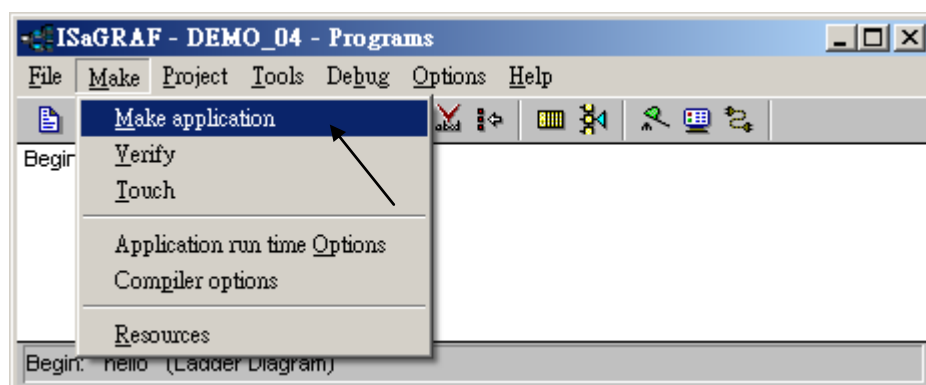
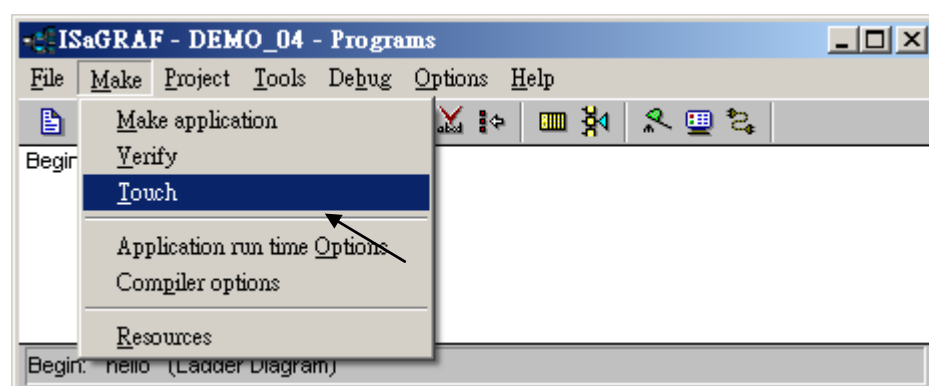
Please connect "Rtu_slav" in the ISaGRAF IO connection window. Re-compile the project and download to the WinPAC via Ethernet (or first Modbus RTU port if it is enabled)



Appendix H: Compiling Error Result In Different ISaGRAF Version

In the recent years since 2003, all the ISaGRAF example programs provided in the ICP DAS CD-ROM & Web site are written in ISaGRAF workbench version of 3.46. If your ISaGRAF workbench is version of 3.51 or newer version, it may generate error when you re-compile these example programs.

To erase this kind of error in different ISaGRAF workbench version, please run “Make” – “Touch” once. And then re-compile this example project.



The “Make” – “Touch” command will reset all files that have been successfully compiled to become “Not compiled yet”. Then the next “Make” – “Make application” command will re-compile all of them.

Appendix I: Using RS-232 Serial/USB Touch Monitor

There are three types of RS-232 Serial or USB Touch monitor supported by the WinPAC.

“penmount_serial_touch” and “penmount_usb_touch” or penmount-compatible Touch monitor .
“elo_serial_touch” and “elo_usb_touch” or elo-compatible Touch monitor.
“egalax_serial_touch” and “egalax_usb_touch” or egalax-compatible Touch monitor.

Note: Recommend to use serial Touch monitor.

The Serial Touch monitor is much better, it is stable.

The USB Touch monitor may not be stable.

To use them with WinPAC, please follow below steps.

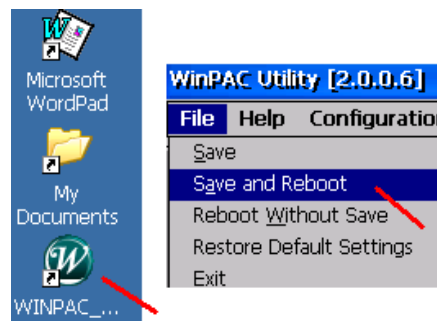
1. Please connect the Touch monitor and its RS-232 signal to the WinPAC's COM4 (Please refer to the appendix A.5 for the COM4 's pin-assignment) and connecting one USB mouse to your WinPAC for configuring the touch driver. Then power on your WinPAC, run the proper Touch monitor driver in the WinPAC's \System_Disk\external_device_driver\ .

There are 6 drivers inside this path. Please run only the correct one for your Touch Monitor. These 6 driver files look like as the following. (If you cannot find them, please visit http://ftp.icpdas.com/pub/cd/winpac/napdos/wp-8x4x_ce50/system_disk/ to download them. Then copy the “external_device_driver” dictory to your WinPAC's \System_Disk\ by ftp)

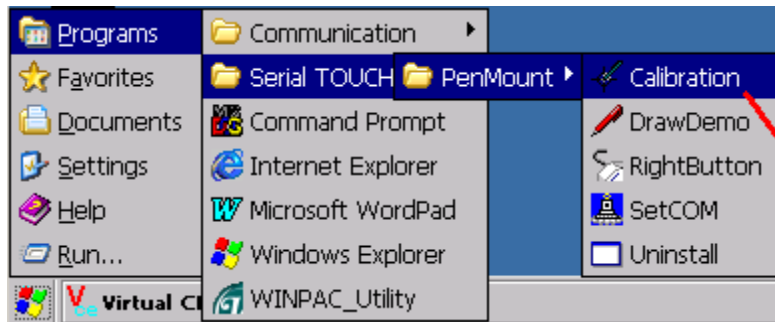
For example, if the Touch monitor is “ADP-1080T-R” (the serial Touch monitor at http://www.icpdas.com/products/HMI/touch_lcd/adp-1080t-r_u.htm : 8" industrial display monitor w/sealed touch screen (plastic bezel), RS-232 Interface), please double-click on “penmount_serial_touch_v20081117.cab” in the WinPAC's \System_Disk\external_device_driver\ to install it. (The last “v20081117” may be a different name depends on its modification date)

penmount_serial_touch_v20081117.cab
penmount_usb_touch_v20081023.cab
elo_serial_touch_v20081027.cab
elo_usb_touch_v20081118.cab
egalax_serial_touch_v20081027.cab
egalax_usb_touch_v20081107.cab

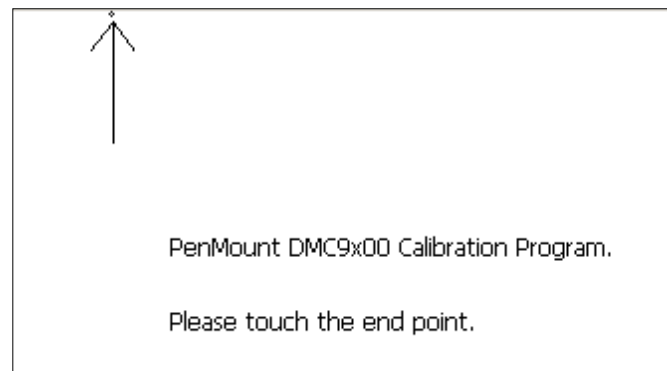
Then run the “WinPAC utility” > “Save and Reboot” to save and reboot it.



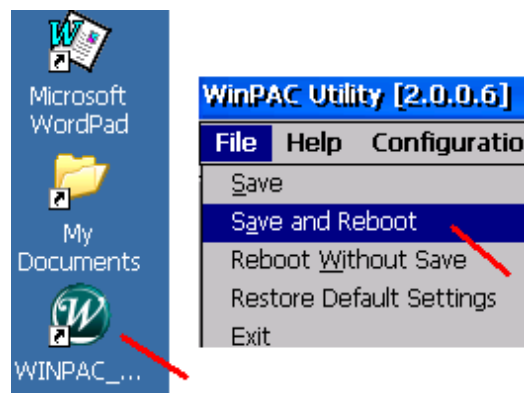
2. Reset your WinPAC. When it boot up, run the “Calibration” to calibrate your Touch monitor.
(If you installed a wrong driver, please run “Uninstall” to remove it first and then go back to step 1 again to install the correct driver.)



Then please follow the calibration command to touch the given point on the monitor by your finger.



When the calibration is done, please run the “WinPAC utility” > “Save and Reboot” to save the calibration setting and reboot it again. Then remove the USB Mouse.



Normally, the default WinPAC's COM4 is Ok for the serial Touch monitor. However for some reasons, user may change it to different COM port as below if he wants.

