# Chapter 4. Linking Controllers To An HMI Program

**Note**: For communicating to W-8347/8747 via Modbus TCP/IP protocol, there are two Ethernet ports built in the W-8x47/8x46 controller, please connect your PC/HMI to W-8347/8747 's "LAN1" port. And please using "NS-205" or "NS-208" Ethernet switch.

This chapter details how to make data from the I-8xx7, I-7188EG/XG, uPAC-7186EG, iPAC-8447/8847 & W-8xx7 controller system available to Human Machine Interface (HMI) programs. This is a powerful feature that allows customers to create their own custom HMI programs and link them to the controller system.

After you realize the material described in section 4.1, if you would like to use the I-8xx7, I-7188EG/XG controller as a **Modbus or Modbus TCP/IP I/O**, you may refer to section 4.3. Additionally there are "touch screen" monitors provided by ICP DAS that support the "Modbus" protocol, and these touch screen monitors can also access data from an controller . Section 4.4 illustrates how to link a "Touch 510" monitor to an ISaGRAF controller system.

#### Note:

1. I-7188EG / 7186EG , uPAC-7186EG, I-8437 / 8837, I-8437-80 / I-8837-80, iPAC-8447/8847 and W-8xx7 controllers all support Modbus TCP/IP Slave protocol at its Ethernet port.

2. I-8417 / 8817 's COM1:RS-232 and COM2:RS-485 default supports Modbus RTU Slave.

3. I-8437 / 8837 's COM1 default supports Modbus RTU Slave protocol. To enable its COM3 to support Modbus RTU Slave , please refer to Chapter 3.5 & 3.10 of I-8xx7 "Getting Started Manual" delivered with the hardware.

4. I-7188EG/XG & uPAC-7186EG 's COM1 default supports Modbus RTU Slave protocol. To enable its COM3 to support Modbus RTU Slave , please refer to Chapter 3.7 & 3.6 of I-7188EG/XG "Getting Started Manual" delivered with the hardware.

5. W-8xx7 defaut no support Modbus RTU Slave port. To enable its COM2 or COM3 or COM5 to COM8 to support Modbus RTU Slave, please refer to the Appendix A.2, Appendix G & F of W-8xx7 "Getting Started Manual" delivered with the hardware.

### 4.1: Declaring Variable Addresses For Network Access

To make data from an I-8xx7, I-7188EG/XG & W-8xx7 controller system available to other software programs or HMI devices, you must first declare the variable with a "Network Address". The variable must be declared with a network address number that is in the "Modbus" format. Other software programs or HMI devices will access the controller information through these network addresses.

There are two methods available to declare a variable for network address access. The first method is described below. Open an "ISaGRAF Programs" windows and click on the "Dictionary" icon, then double click on the variable to assign a network address number.

#### Note:

1. The valid network addresses for an I-8417/8817/8437/8837, I-7188EG/XG, uPAC-7186EG and iPAC-8447 / 8847 controller system is from 1 to FFF in hexadecimal (1 ~ 4095). Network address 5001 to 8072 is for word and integer arrays, please refer to Section 4.5.

2. The valid network addresses for an **W-8037/8337/8737 & W-8347/8747** controller system is from 1 to 1FFF in hexadecimal (1 ~ 8191). Network address 10,001 to 19,216 is for word and integer arrays, please refer to Section 4.5.

There are two ways to assign a Modbus network address No. to a variable. One is as below figure. (To assign many Modbus Network address No. to the "Variable Array", please refer to Chapter 2.6)

ISaGRAF -	ST_INTER - P	rograms			×		
File Make Pi	roject Tools	Debug Options	Help				
8 1 6	00 00	1 × × 10	III 💥	202			
Begin: Diction	nary 🕾 ST	Inter ST Example	Using Inten	nal Variables			
💊 ISaGRAF -	ST_INTER - 0	Global booleans				1	
File Edit Too	ols Options	Help				I г	
		000	* 🗈 🎖	1			Note:
Booleans Inte	gers/Reals   Ti	mers Messages F	B instances	Defined wo	rds		The value displayed here
Name	Attrib	. Addr.	Comment		and the second second		is always in hexadecimal.
D1	[input	0000	Real Input	#1	<b>A</b>		
D2	1 [input	0000	Real Input	#2			
D3	[input	0000	Real Input	#3			/
OUT1	[outpu	.t] [0000	Real Outp	ut #1		/	
Boolean Varia	ble					•	×
Name:	D1			Networl	Address: (	5	
				1.1.1.200000.00000		$\sim$	
Comment:	// Real In	put #1	00000000		3		
Attribute		Set Net	work Ad	dress —		Commence	
Attribute	•	In This	Data Fie	ld	_	Stor	e
CInterr	nal	raise:					12
Input	E0		-		_	Canc	el
COutp	ut	True:					
C Const	tant	E ant to	in at ini			Nex	R
Cons	Conc	1 sec to	uue at inii			Provic	
1		l Retain					
						Extend	ded

When you click on the "Store" button you will see that "ISaGRAF Global Variables" window will now be updated with the new network address for the variable.

SaGRAF-ST_IN File Edit Tools Op	TER - Global bo tions Help	oleans		
	🗃 🔾 💽 (	9 🗏 🕯	× 🗈 🤞 🖄 🛎	
Booleans   Integers/Re	als   Timers   Me	ss <u>ag</u> es   FE	instances   Defined wo	rds
Name	Attrib.	Addr.	Comment	
D1	[input]	0005	// Real Input #1	-
D2	[input]	0000	Real Input #2	
D2	-pput]	0000	Real Input #3	
Network	ut at	0000	Real Output #1	
Address IS No	utput]	0000	Real Output #2	
Set 10 "5"	"Jutput]	0000	Real Output #3	-
) (* // Real Input #1 *	)	1		

The second method for assigning network addresses to variables requires that you declare the variables BEFORE you assign them. This method allows you to assign numerous network address variables before you link them to an ISaGRAF program.

	- ST_INTER - Programs		
File Make	Project Tools Debug Options He	lp	
🖹 🖪 🗧	8 🛈 🗅 💼 🐞 🗶 io 🛛	s 🖻 🕺 🙀 🎟	
Begin: Dict	ionary 🏝 ST Inter ST Example Us	ing Internal Variables	
💊 ISaGRAI	- ST_INTER - Global booleans		×
File Edit	Tools Options Help		
	Quick declaration	: 🗈 🤞 📉 📇	
Booleans	Modbus SCADA addressing map	nstances Defined words	_
Name	Import text	Comment	
D1	Export text	(Real Input #1	-
D2	Import true/false definitions	teal Input #2	
D3 -		Real Input #3	
OUT1	Sort	Real Output #1	
OUT2	Renumber addresses	Real Output #2	
0UT3	I/O connection	Real Output #3	+
D1 (* // Re @0005 [ir	Conversion tables Cross references		

When you click on "Modbus SCADA Addressing Map" (SCADA is an industrial process control acronym that stands for "Supervisory Control And Data Acquisition") the "Modbus SCADA Addressing Map" window will open.

Note that one of the variables (D1) is already assigned from our previous example. You will note that the other variables that are not yet mapped are displayed in the lower portion under the "Variables (Not Mapped)" portion of the "Modbus SCADA Addressing Map" window.

ile Edit Op	tions Help			_
Map Segment:	[00000FFF]		•	
0000 (R 0001 0002 0003 0004 0005 № 0006 0007 0008 0007 0008 0009 000A 000B	eserved) (* // Real Input #1 *)	Y Fi	ou Can See The irst Variable We ssigned In The Fir lethod Example	st
Variables Booleans D2 (* Real D3 (* Real OUT1 (* Re OUT2 (* Re OUT3 (* Re	net mapped) Integers/Reals Timers nput #3 *) al Output #1 *) al Output #2 *) al Output #3 *)	Messages	Select The Va "Type" You W Declare Here	riable lant To

To assign the other variable address click on an unassigned "Map Segment" number, and then double click on the variable you want to assign to the address and the variable will automatically assign itself to the "Map Segment".

Segmer	at: [00000FFF]
0000 0001 0002 0003 0004 0005 0006	(Reserved)
0007 0008 0009 000A 000B	λ <del>α</del>
Variable	es (not mapped)
Boolea	ns Integers/Reals   Timers   Messages
D3 (* Re	eal Input #3 *)
OUT1 (*	Real Output #1 *)
OUT2 (*	Real Output #2 *)

To assign continuous Network address to similar variables, for example, assigning No. 1 to 5 for D001 to D005, please select those variable names and then click on "Edit" – "Map selected variable".

Modl	bus S	CADA ad	ldressing	g map			×
File	Edit	Options	<u>H</u> elp		_		
- Ma	M	lap selected	variable		L		
Se	<u>R</u>	emove var	iable from	ı map	•	-	
Π	Fi	ind					
0	)01						
00	)02						
00	003						
	JU4						_
00	106						_
	007						
00	)08						
00	009						
	JUA						
	100						_
−Va	riable:	s (not mapp	oed)				
E	loolea		rs/Reals	Timers [ N	lessades	1	
DC	001	1					
DC	002						
DC	003						
DO	004						
	105						
DC	07						<b>_</b>

For human's thinking way, network address represented in hexadecimal format is inconvenient and it increases the chance to make mistake. Therefore, it's better to change it to be represented in decimal format. To do that is as following.

Modbus SCA	DA addressing map	×
<u>E</u> ile <u>E</u> dit	Options Help	
-Мар —	<u>H</u> exadecimal	
Segmer	🖌 🗹 Decimal	J5] 🔹 🗸
00000	(Reserved)	
00001		
00002		
00003		
00004		
00005		
00006		
00007		
80000		
00009		
00010		
00011		▼

#### IMPORTANT NOTE REGARDING MODBUS NETWORK ADDRESSING

The Modbus network address definition scheme is sometimes different between HMI devices and other software programs. The difference is typically that the other programs may assign a network address number that is one (1) less than that of the I-8xx7, I-7188EG/X & W-8xx7 controller system.

HMI or devices such as Indusoft, Iconics, Citech, Wizcon, Kepware's OPC server, iFix, Wonderware's "Intouch", National Instruments "Labview", and ICP DAS's Touch 506L, Touch 506T and Touch 510T do have the exact same addressing scheme as the I-8xx7, I-7188EG/X & W-8xx7 controller system.

Known addressing disparities include "LabLink" and "Hitech" HMI software programs and devices. If you are assigning a network address of "B" (hexadecimal) of these products the I-8xx7 network address should be set to "C". A network address of "2" should be associated with a network address of "3" in the ISaGRAF controller system.

Another things mistaked very often is the first digit of the network address of many HMI softwares resprent the data type and Read/Write authority not one part of the network address. For example, the network address relation between "iFix" and ISaGRAF is as below.

iFix(Decimal)	I-8xx7 (Decimal)
<b>0</b> 0001 (R/W Boolean)	1
10010 (Read Boolean)	10
<b>3</b> 1000(Read Word)	1000
<b>4</b> 2101(R/W Word)	2101

ICP DAS has not been able to test every possible HMI software program or hardware device that has Modbus addressing capability. If you are trying to connect your HMI software program or hardware device with Modbus to an I-8xx7, I-7188EG/X & W-8xx7 controller system, **REMEMBER** that you **may** have to offset the Modus addressing by 1 between these products so they will properly communicate with each other.

Developers who design and write their own software interface programs using Microsoft's Visual Basic or Visual C++ programming language should refer to Chapter 5 of this manual for more information on how to interface the Modbus protocol to these programming languages.

### NOTE:

While communicating with the I-8xx7, I-7188EG/XG, uPAC-7186EG, iPAC-8447/8847 and Wincon-8xx7, **One single Modbus frame** cannot request more than **255 bits** except the Wincon-8xx7 (Max. 1968 bits for W-8xx7), and also cannot request more than **120 words** in one single modbus frame. It should be divided into 2 or more reading frames to achieve it. To write bits to the controllers, **One single Modbus frame** cannot write more than **255 bits**, and also cannot write more than **120 words** in one single modbus frame. It should be divided into 2 or more reading frames to achieve it.

## 4.2:Read/Write Word, Long Word & Float through Modbus

Modbus protocol provides function 3 and 4 for reading multiple words while function 6 and 16 to write words. Please refer to Chapter 5 for more information about the protocol.

The word defined in the Modbus protocol of I-8xx7, I-7188EG/XG, uPAC-7186EG, iPAC-8447/8847 & W-8xx7 controllers is like a signed short integer, which occupies 2 bytes and range from -32,768 (8000 in hexa.) to +32,767 (7FFF in hexa.). It is normally used to describe the behavior of analog I/O channels. For examples, the I-87017 I/O board (please refer to section 3.2)

Range ID	Electrical	Values on t	the channel (	(decimal)
(hexadecimal)	Range	-32768	0	+32767
8 (default)	$\pm 10V$	- 10V	0V	+ 10V
9	± 5V	- 5V	0V	+ 5V
A	± 1V	- 1V	0V	+ 1V
В	± 500mV	- 500mV	0mV	+ 500mV
С	± 150mV	- 150mV	0mV	+ 150mV
D	± 20mA	- 20mA	0mA	+ 20mA

I-87017:

The **long word** defined in the Modbus protocol of I-8xx7, I-7188EG/XG, uPAC-7186EG, iPAC-8447/8847 & W-8xx7 controllers is like a signed long integer, which occupies 4 bytes and range from -2,147,483,648 (8000 0000 in hexa.) to +2,147,483,647 (7FFF FFFF in hexa.). It is normally used to describe the value of internal integer variables declared on ISaGRAF workbench.

All integer variables declared in ISaGRAF are signed 32-bit format however the integer variable, which assigned with a network address will only, occupies 1 word (2 bytes) in the Mudbus transportation format. Since a long word occupies 2 words (4 bytes), to R / W long word through Modbus, the network address assigned to the integer variable must follow rules as below.

<ul> <li>▶ ISaGRAF - SA</li> <li>Eile Edit Iool</li> <li>Booleans Integers</li> <li>Name</li> <li>V1</li> <li>V2</li> <li>V3</li> <li>V4</li> <li>V5</li> <li>V6</li> <li>V7</li> <li>V8</li> </ul>	- Global integers/reals s Options Help Attrib. Addr. (internal,integ 0003 (internal,integ 0003 (internal,integ 0005 (internal,integ 0007 (internal,integ 0008 (internal,integ 0008 (internal,integ 0008 (internal,integ 0008 (internal,integ 0008 (internal,integ 0000	<ul> <li>V1 is assigned to a network address "1". If the network address "2" is not assigned to any other variable, V1 will occupy a long word (4 bytes) in the Modbus transportation formate.</li> <li>However if "2" is assigned to one another variable, V1 will only occupy one word (2 bytes) in the Modbus transportation format.</li> <li>In this example, V1, V2, V3, V6, V7 and V8 will occupy 4 bytes however V4 and V5 only occupy 1 word (Lowest word) in the Modbus transportation formate.</li> </ul>



To read **long word** value of V1 is to read **2 words** by using modbus function 3 or 4 (please refer to section 5.1).

vH vL drcH crcL 02 04 vH vL 10 00 00 00 Req: slv slv 10 00 00 00 02 crcH crcL Ans: Highest word Lowest word

To read / write float (4 bytes) is very similar to read / write long word. The difference is the variable should be declared as "Real" type, and the next network address No. should not be assigned to any other variable.

Integer/Real Varia	ble		×
Name:	A1	Network Address:	1
Comment:			
Unit:		Conversion: (none)	•
Attributes © Interna O <u>I</u> nput O <u>O</u> utput O Const <u>a</u>	nt Initial value: 0	standard)	Store         Cancel         Next         Previous         Extended

There are much available HMI software on the market. You don't need to care about the modbus protocol format. Just be careful to assign the correct network address on ISaGRAF.

# 4.3: Using I-8xx7 As A Modbus I/O Or A Modbus TCP/IP I/O

There are some configurations that the HMI software gathers the I/O data from some called Modbus I/O modules. There I/O modules scan each input channels and refresh the output channels when need. Most of time there are no control logic inside these I/O modules, they are controlled by the HMI. To fit such kind of usage, the I-8417/8817/8437/8837 can be a Modbus I/O module, additionally the I-8437/8837 can be a Modbus TCP/IP I/O module. To do that, follow the following procedures (If you are not familiar with the ISaGRAF programming, recommended to review Chapter 2).

Create a new project

You may refer to section 2.1.1.2 Example:



#### Create an empty program No logic need. Example:

📲 ISaGRAF - CR	EATION - Progr	ams 💶 🗆 🗙
Eile <u>M</u> ake Pr	oject <u>T</u> ools D	Debug Options Help
🖹 🖬 😵 🕮		※ IP ■ 数 条 ■ 24
	Create new pro	gram
	Nour Drogram	
	New Flogram	<u> </u>
	Name:	empty
	Name: Comment:	empty
	Name: Comment:	empty Ouick LD : Ladder Diagram
	Name: Comment: Language:	empty Quick LD : Ladder Diagram
	Name: Comment: Language: Style:	empty Quick LD : Ladder Diagram Begin : Main program

#### Connect I/O modules

You may refer to section 3.1

Example:



Declare Variables associated with the channels of connected I/O modules.

You may refer to section 2.1.1.3

Example:

-	SaGRAF - SIMPLELD - Programs	
File	Make Project Tools Debug Options Help	
	Щ⊗∭ ┣◨曲 ॐ凇፦ ▥淋 옷▣칺	
	مخ Dictionary	
	SaGRAF - SIMPLELD - Global booleans	- <b>D</b> ×
	File Edit Tools Options Help	
	🖆 🔾 🚱 👙 😽 🗈 💰 💐 🖴	
	Booleans Integers/Reals   Timers   Messages   FB instances   Defined words	
	Name 💦 Attrib. Addr. Comment	
	1	
		~
		and the second se

💊 ISaGRAF - S	IMPLELD - Global booleans
File Edit Tools	Options Help
	🖴 🔾 🕼 🤄 🖷 🛩 🗈 🧉 🔛 🖴
Booleans Inter	ers Reals Timers Messages FB instances Defined words
Name	Attrib. Addr. Comment Quick declaration
SW1	[input] 0000 Switch 1
SW2	[input] 0000 Switch 2
SHUT	[input] 0000 Shutdown (Emergency Stop Type) Switch
	Quick declaration
	- Numbering:
	From: I Io: 3 V
SVV1 (* Switch	Digits: 2
@0000 [input]	
	Symbol:
	(Name: OUT ) ##
	A11-3-1-1-1
	Attributes.
	Ointernal Oinput
	Constant COutput
	Other:
	Betain
	Formatr Clatanas C Bast
	romac omeger oneal
	Length:

Link Variables to the associated channels of connected I/O modules.

You may refer to section 3.1.2

Example:



Assign the linked Variable a network address No.

You may refer to section 4.1

Example:

• ISaGRAF	- ST_INTER - Programs					
File Make	Project Tools Debug Options I	Help				
	8 M 🗅 🖬 🍵 🤻 🗶 🕪	💷 🏂 🧏 🛄 🖏				
Begin: Dict	ionary 🖾 ST Inter ST Example	Using Internal Variables				
SISaGRA	- ST_INTER - Global booleans					
File Edit	Tools Options Help	agente contractor				
	Quick declaration	< 🗈 💰 🖄 🖀				
Booleans	Modbus SCADA addressing map	instances Defined words				
Name	Import text	Comment				
D1	Export text	7 Real Input #1 📃 🔺				
D2	Import true/false definitions	Real Input #2				
D3 -		— Real Input #3				
OUT1	Sort	Real Output #1				
OUT2	Renumber addresses	Real Output #2				
OUT3	I/O connection	Real Output #3	+			
	Conversion tables					
D1 (* // Re @0005 [ir	Cross references					

Compile & download the project You may refer to section 2.1.3 & 2.1.5

#### Note:

Make sure the Net ID is set to the proper No. (section 1.3.1) For I-8437/8837, make sure the IP and Mask address is well set (appendix B).

The HMI can access to I/O channels through the associated network address now!

## 4.4: Linking I-8xx7, I-7188EG/XG & W-8xx7 To Touch 500

Touch500 series HMI support below protocols to link to ICP DAS ISaGRAF controllers.

Item	Protocols
Touch-506L	Modbus RTU RS-232 , Modbus RTU RS-485
Touch-506T	Modbus RTU RS-232 , Modbus RTU RS-485
Touch-510T	Modbus RTU RS-232 , Modbus RTU RS-485

Please install "EasyBuilder 500" software (Ver. 2.7.1 or later version) first before you can program the Touch 506L, 506T, 510T HMI. You may download the new released software and manual from below web site

http://www.icpdas.com/download/others/touch/touch.htm "setup.zip" or run "setup.exe" at I-8000 CD-ROM:\napdos\others\touch\500series\setup\

RS-232 Cable Pin assignment of PC to Touch 500 series (For PC to download HMI screen).

PC 9-Pin DSUB Female (RS232)	Touch 500 (PC-232) 9-Pin DSUB Female
RXD 2	8 RXD
TXD 3	7 TXD
GND 5	5 GND

RS-232 Cable Pin assignment between controllers and Touch 500 series.

I-8000 COM1 & I-7188/7186 COM1 (RS232)	Touch 506T/506L/510T (PLC 232)
9-Pin Dsub Male 2 TXD 3 RXD 5 GND	9-Pin Dsub Male 2 TXD 3 RXD 5 GND 7 CTS 8 RTS
Wincon COM2 (RS232)	Touch 506T/506L/510T (PLC 232)
9-Pin Dsub Female 2 RXD 3 TXD 5 GND	9-Pin Dsub Male           2 TXD           3 RXD           5 GND           7 CTS           8 RTS
RS-485 Cable Pin assignment between controllers	and Touch 500 series
I-8417/8817 COM2 (RS485) +* DATA + DATA	Touch 506T/506L/510T (PLC 485)          2 R+          1 R-
Wincon COM3 (RS485) D + D	Touch 506T/506L/510T (PLC 485) 2 R+ 1 R-

### 4.4.1: Program the I-8xx7, I-7188EG/XG & W-8xx7

To make data of the I-8xx7, I-7188EG/XG, uPAC-7186EG, iPAC-8447/8847 & W-8xx7 controller to be accessible to the Touch 510T, variables in the controller should be assigned a network address. Please refer to section 4.1, 4.2. If you are not familiar with the ISaGRAF programming, recommended to review Chapter 2.

s used in this	s example.			
Name	Туре	Attribute	Network address	Others
OUT01	Boolean	Output	0001	-
OUT02	Boolean	Output	0002	-
VAL1	Integer	Internal	000A (10)	-

Variables used in this example.

IO connection:

📷 ISaGRAF - TEST - I/O connection	_ 🗆 ×
<u>File Edit Tools Options Help</u>	
🖴 📼 🗟 🗭 💼 👌 🤣 🕞 👗 🖀	
• ► :::: ref = 10	
1 OUT01	
2 OUT02	
3 3	
4	
5	
6	
7	
🕫 📼 show3led 🛛 🗛 🔸	
9	
10	
11	
12	

A simple LD program to show the "VAL1" to 7-segment LED:



After you finish this project, compile and download it to the I-8xx7 controller.

### 4.4.2: Program the Touch 510T

The "EasyBuilder 500" software can be used to designe many useful pictures for Touch 500 series. This section illustrates a simple example to program a Touch 510T. For more information about programming on the Touch series, please refer to the user manual which is provided with the "Touch" series hardware.

Click on the Windows "Start" button, then click on the "Program" button, then click on the "EasyBuilder" – "EasyBuilder 500" button. The following window will be displayed. Select the proper model for your application.



Click "File" - "New" to create a new project.

EB	EasyBi	uilder -	[ EBP	rjl : Wi	indow 1	0 - Ini	itial Screen	]				_	
EB	<u>F</u> ile	<u>E</u> dit	View	<u>O</u> ption	Draw	Parts	Library	$\underline{T} \text{ools}$	Window	Help		_	<u>a x</u>
Ľ	Ne	ew			Ct	xl+N	- 	ની નાં	111 🖻		• 🕘 💷 🖶	Щ.	Stat
	Q	pen			Ct	rl+O				E A	. 8 m a.	alt F	利用
	<u>C</u> 1	ose						innini la		1- 4			
	<u>S</u> a	ve			Ct	rl+S						- _	믝
99	Sa	ve <u>A</u> s											님
$\overline{\ }$	11	test.epj											
	21	EBPrj1	.epj										
$\overline{O}$	<u>3</u> (	C:\EB5	00_T\Pr	oject\test l	l.epj							<u></u>	
<u>(</u>	4 0	C:\EB5	00_T\Pr	oject\test2	2.epj								
$\frac{A_{\lambda}}{\infty}$	E>	cit					_						
<b>888</b> - 지신		15	7										11
		- 18	3										RP
Ì		- 19											ED
M		4											
PLC											[	•	
FI		🖥 Win-	dows					_			•		
		ど (Ub)	jects	Create a 1	new docu	nent		X	X = 37 Y =	4		///	

Click "Edit" – "System Parameters" to set the communication parameter between the Touch 510 and the ISaGRAF controller.

<b>FR</b> EasyBuilder -	Window Copy	
58 File Edit	<u>P</u> aste Ctrl+V	arw Tools Window Help - RX
	<u>D</u> elete Del	
Font: 16	Layer	- ㅋㅋㅠㅋ ㅌㅎㅋㅠㅠㅠ ㅌ
	Nudge	
📐 🗉 Windo	Align	- 9 월
<u>- 4:</u>	Make Same Size 🔹 🕨	
	Flip <u>V</u> ertical	
	Flip <u>H</u> orizontal	🖌 😪 💥
<u>c</u> 13	Rotate <u>2</u> 0 degree	
<u> </u>	Grown	- in 1
- 15	Lioup	
- 16 .	∪ກ⊲າ⊴ແp	
- 17	Redra <u>w</u> Window	III RP
- 18	Select All Objects	
	✓ Select	
	Colort Mourt Object	
M •	perect Mext Object	
PLC THE Wind	Change Attribute	
🖭 🛛 🎦 Obje	System Parameters	X = 145 Y = 30

PLC type should be set to "**MODBUS RTU**", Serial port set to "RS232", Data bits set to "8 Bits", Stop bits set to "1 Bit", Baud rate set to "19200", Parity set to "None", PLC station No. set to be equal to the Net-ID of the I-8xx7 (set to 1 in this example).

PLC type : MODBUS RTU	<b>•</b>
HMI model : MT510T (640 x 480)	
Serial port I/F : RS232	Baud rate: 19200
Stop bits : 1 Bit	
HMI station No. : 0	PLC station No.:
Multiple HMI : Disable	HMI-HMI link speed : 115200
PLC time out constant (sec) : 3.0	PLC block pack : 0

Note:

 If using Touch506TE 's Ethernet to link to controller, please set PLC type as "MODBUS RTU TCP/IP", PLC I/F port as "Ethernet", Local IP address as Touch506TE 's IP, Server IP address as controller 's IP, PLC station No. as the same Net-ID No. of the controller (default is 1)
 If the cable between the Touch 500 series and the controller is 2-wire RS-485, please set PLC type as "RS-485 2W". Other setting is the same as RS-232. Click on "Text" to add a text. Select the prefered "Color", "Font", "Align" for the text and then enter the "Content". And then place it to the proper position.

<b>B</b> B	EasyBı	uilder -	[ EBPi	cjl : ₩i	ndow 1	0 - In	itial Scree	n]			_ 0 ×	1	
B	Eile	Edit	⊻iew	Option	Draw	Parts	Library	Tools	Window	Help	<u>- 8 ×</u>	1	
		8		122	3	N?	<u>/</u> 3	ાં 📶 ન	100 E		0 🖬 🖥		
	Font	: 16	Ţ	A A E						우리	릭 ㅠ 아	l I	
<b>k</b> Øy		Windo 4:	ws Fa										
$\mathbf{N}$		- б - *10	Create T	lext Objec	t								×
		- 1	Text										
0		- 1	At	tribute									
Ā		- 1		Color :				•	Fo	nt: 32		•	
	Text	- 1		A1:	Cantar			1		- i -			
꼺	T	- 1		Align :	Center								
~		- 1	Con	tent :									
Ē		- 2	Wel	lcome To	Touch 5	10 !							-
M	iks	L A	Buildor	f ERP-	1 . W	indow	10 Ini	ial Server	<u>م</u> ا				
PLC		🕫 Eile	<u>E</u> dit	View	Option	Draw	Parts	Library	Tools	Window	Help	- 8 ×	
E		D			28	8	? <b>\ ?</b>   _	10014		<b>1</b>			
		Ean	L. 33		e ar e				I E F	साम	[[ 오 리	m or	
			U: 3Z										
			- 4: - 6 -*10 - 12	ows : Fa ): Ir 1 2	We	lca	me	To 1	Fouc	h 51	0		
	1	< AL	- 1	3							-		

Click on "Function Key" to add a change-window button. Click on "General", then select "Change Window" and set "Window No." to 11.



Click on "Shape", then select "Use shape" and the click on "Shape library ..."

Create Functi	ion Key Ob	ject			×
General	Shape ]	Label			
	Shane				_
	ыларе	Shape	library	Use shape	
	-Bitman-		T		
	Ditiliap	Bitmap	library	🗖 Use bitmap	
			State : 0	•	
		確定	取消	套用(A)	說明

Select the prefered "Shape library" and then select one item and click on "OK".

hape Library			
Shape library:	button1	Sta	ate : 0 🔹
0: Untitled	1: Untitled	2: Untitled 3:	Untitled
	E: Untitled		
Packground :	5. ontitied		ondied
Backyround :			
Select Lib	New Lib	Unattach Lib.	Delete shape
Place		ОК 🔪	CANCEL

Click on "Label", then select the prefered "Color", "Font", "Align" and set "Content" to "GOTO S11", and **make sure "Use label" is selected**.

Create Function Key Object	×
General Shape Label	
Attribute	
Color : 🔽 Font : 16	<b>.</b>
Align : Center State : 0	<b>_</b>
Content :	
GOTO S11	<u> </u>
	<b></b>
✓ Use label	
<b>確定 取消</b>   <br< td=""><td>説明</td></br<>	説明

Click on "Bit Lamp"



Click on "General", then select "Device type" to "**0x**" (**0x is for boolean variables**), then set "Device address" to 1 (this value is associated with the network address value of the variable in the I-8xx7). And then set "Function" to "Normal".

Create Bit Lamp Object	X
General Shape Label	
Pescription :	
Read address	
Device type : 0x Device address : 1	
	-
Attribute	
Function : Normal	

By the same way as former, select prefered "Shap library".

Create Bit L	amp Objec	et in the second s			×
General	Shape 🔪	Label			
	-Shape-	Shape li	brary	☑ Use shape	
	Bitmap	Bitmap li	ibrary	use bitmap	
	Sha	ipe Library			×
		Shape library:	button1		State: 0 🔹
		16: Untitled	17: Untitled	18: Untitled	19: Untitled
		20: Untitled	21: Untitled	22: Untitled	23: Untitled
		Background :		-	
		Select Lib	New Lib	Unattach Lit	Delete shape
		Place		ОК	CANCEL

And then select "Label", given a "OFF" to "Content" for "State : 0". Make sure "Use label" is choosed.

Create Bit Lamp Object	×
General Shape Label	
Attribute	
Color: Font: 16	<u> </u>
Align : Center State : 0	•
Content :	
OFF	<u> </u>
T	Þ
✓ Use label	
確定 取消 套用(A)	說明

And then change "State" to 1, and given a "ON" to "Content". Make sure "Use label" is choosed.

Create Bit Lamp Object	×
General Shape Label	
Attribute	
Color : Font : 16	•
Align : Left State : 1	
Content :	
ON	
	Þ
Use label Tracking	
確定 取消 套用(A)	說明

By the same way as former, create one another Bit Lamp with a "Device address" = 2.

E

Bit Lamp Object	's At	tribute						×	
General Sha	pe	Label Profile							
Descriptio	n : [							-	
Read address					$\geq$				
Device typ	€:	0x	-	Device addı	ress :  2	)			
·									
-Attribute-									
Functio	n : [	Normal	¥						
	_								
	ER 1	syBuilder - [ EBI File Edit View	Prjl : Wir Ontion	ndow IU - I Drow Porte	nitial Screen.	Toolo	Window	Holm	
	n l		⊻puon Nol⊘l						
							ind ind		
	Fo	ont:16 🚽						阜阜	<u>희 때 아</u>
		🖃 Windows 🔺							鲁뭨
	<u>%</u>	- 4: Fa-	Wel	come	To T	ouc	h 51	0	- <b>™</b> - <b>™</b>
-	긝	*10: Ir							
-	눼	- 11	EK 1	0	_	•			
-	Ť	- 13	GOT	o D S11		6			
	A	- 14					<u>)</u>		<u> </u>
	20	15							┚▦┦╝
4	<u></u>	- 17							
-		- 18							
-		20							
-	M								_
-	PLC	• Windows						►	1
	EL	🐮 Objects	For Help, p	oress F1		X = 236	Y = 142		

Click on "Toggle Switch", then set all "Device Type" to "**0x**", all "Device address" to 1 and select "Switch Type " to "Toggle".



By the same way as former to choose a prefered "shape" and "label".

Create Toggle Switch Object 🛛	
General Shape Label	
Shape	
Shape library 🔽 Use shape	
Bitmap	
Bitmap library 🗖 Use bitmap	
State : 0	
Create Toggle Switch Object	X
General Shape Label	
Attribute	
Color: Font: 16	
Align : Center State : 0 🔽	
Content :	
OFF	<b>A</b>
4	
🔽 Uge label 🗖 Tracking	
EasyBuilder - [EBFTj] : Window IU - Initial Screen]       Image: Screen initial Screen initinitial Screen initial Screen initial Screen initial Sc	╏
	<u>.</u>
	1
	4
🚆 📲 Welcome To Touch 510 📲 🚆	
	1
X = 293 Y = 190	

By the same way as former, create one another "Toggle Switch" however set all "Device address" to 2 and "Switch style" to "Momentary". Click on "save" to save the project.

Toggle Switch Object's Attribute	
General Shape Label Profile	
Description :	
Read address	
Device type :  0x Device address :  2	
Write address	
Device type: Ux Device address : 2	
Attribute	
Switch style : ON	
OFF Taggelo	
Momentary	
💴 EasyBuilder - [ EBPrj1 : Window 10 - Initial Screen]	J×
	빈스
	0]-
Normal Contraction of the second seco	<u>8</u>
👋 📲 🖟 Welcome To Touch 510	<u></u>
	¥+01
	<u>-</u>
	RP
	ED
Image: Section of the secti	

We are going to design another window. Click on "Windows" – "11", then click and hold on the right button of the mouse and drag to "Create".



**Double click** on "Window\_011".

Pile Edit View Option Draw Parts Library Tools Window Help Image: Second sec	EB EasyB	uilder -	[ EBF	rjl : Wi	ndow 1	1 - W	indow_011	]			_ 🗆 🗵
Image: Second	B Eile	<u>E</u> dit	⊻iew	Option	Draw	Parts	Library	Tools	Window	Help	_ 8 ×
Font: 16       Image: Second sec				22	3	<b>N?</b>	<u>/</u> 本	-1	101 🗉		
Windows       4: Fa         -       6         *10: Ir       *11: Window 011         -       12         -       13         -       14         -       15         -       16         -       17         -       18         -       19         -       20         *       *	Font	: 16	-	A A E		5	백 등 등	田田	5 F F	[] 후	의 ㅠ ~
		Windov 4: - 6 -*10 -*11 -12 -13 -14 -15 -16 -17 -18 -19 -200 -200 	WS Fa	ow_011							

Create a change-window "Function Key" as former method to change to "Window No." = 10, and Labeled as "BACK".

Create Function Key Object					×	
General Shape Label						
Description :						
C [ENT] C	[BS]	C [CLR]	O (E	(SC]		
C [ASCII]		C Hard Copy	Attrib	utes		
Change Window		C Return to	Previous	<u> </u>		
C Change Common Wi	ndow	Window No.	. [ 10	$\overline{}$		
C Popup Window		C Close Win	udow		1	
55 EasyBuilder - [ EBF	rjl : Window	w 11 - Window_0	11]		_	
Eile Edit View	Option Dr	aw Parts Library	y Icols Y	Mindow Hel	p _	9 ×
	122 8				<u>&gt; 0</u>	
Font: 16 🔹	A 🗡 🔳	≣ <u>°</u> 8 9			후 引 :	아
Windows	FK 0 BAC	<b>ж</b>				
E Objects	For Help, press	F1	X = 181 Y	7 = 99		

Click on "Set Word", then set "Device Type" as "4x" (4x is for short integer, 4L is for long integer), set "Device address" to 10, "BIN", and "Set style" to "Set Constant", and "Set value" = 100. And then select the prefered "shape", and set "label" to "Set to 100".



Click on "Numerical Data", set "Device Type" to "4x" (**4x is for short integer, 4L is for long integer**), "Device address" to 10, "BIN", "Number of words" to 1, "No. above Dec" to 7, "No. below Decimal" to 0, "Input low" to -32768, "Input high" to +32767. And then select the prefered shape.

🖪 EasyBuilder - [ EBPrj1	: Window 11 - Window_011]
Eile Edit View Q	ption Draw Parts Library Tools Window Help _ 🗗 🗙
Font: 16 • A	
Image: Windows       -         Image: Windows       - <td>FK_0   BACK     Set to   100     Numeric     Font     Numeric     Font     inption :     idress   expre:   4x   Device address :   10   No. of words :</td>	FK_0   BACK     Set to   100     Numeric     Font     Numeric     Font     inption :     idress   expre:   4x   Device address :   10   No. of words :
	Create Numeric Data Object
	General Numeric Font
	Display
	© Decimal C Hex Binary C Mask
	© Raw data display C Do conversion
	Numeric
	Mo. above Dec. :         7          No. below Dec. :         0            Input low :         -32768         Input high :         32767
	<b>確定 取消</b> 套用(A) <b>説明</b>

Now we are going to add one another "Numerical Data" with conversion.

Click on "Numerical Data", set "Device Type" to "4x", "Device address" to 10, "BIN", "Number of words" to 1, "No. above Dec" to 5, "No. below Decimal" to 0, "Input low" to -32768, "Input high" to +32767, check "Do conversion", set "engineering low" to -10, "engineering high" to +10 (**Convert** [-32768,+32767] to [-10,+10]). And then select the prefered font.

Image: Second secon
Windows
General Numeric Font
Description :
Device type: 4x   Device address : 10
BIN No. of words : 1
Create Numaria Data Obiest
General Numeric Font
Display
💿 Decimal 🔿 Hex 🔿 Binary 🔿 Mask
C Single float C Double float
C Raw data display C Do conversion
Numeric
No above Dec : 5 No below Dec : 2
Input low :  -32768 Input high :  32767
Engineering low: -10 Engineering high : 10

Click on "Numerical Input", set "Device Type" to "4x", "Device address" to 10, "BIN", "Number of words" to 1, **"Trigger Device Type" to "LB", "Trigger Device address" to "9000",** "No. above Dec" to 7, "No. below Decimal" to 0, "Input low" to –32768, "Input high" to +32767. And then select the prefered shape.

BesyBuilder - [ EBPrj1 : Window 11 - Window_011]
Image: Windows     Image: Window
General Numeric Shape Font
Description :
Device type: 4x
BIN No. of words : 1
Trigger address
Device type : LB   Device address : 9000
Create Numeric Input Object
General Numeric Shape Font
Display
© Decimal © Hex © Binary © Mask © Single float © Double float
Numeric
No. above Dec. : 7 * No. below Dec. : 0 *
Input low : -32768 Input high : +32767
確定 取消 套用(A) 説明

Font : 1	dit View dit View 6 - indows - 4: Fa - 6 -*10: In Compiling		ndow II - Draw B B <b>?</b>	- Window_01 arts Library 2 ▲	I Tools <u>Com</u> On-l: Off-I Dow	Window pile ine Simula line Simula nload	Help tion 🔊		X
	Co	Proje ompile fi	ct name le name	: C:\EB50 : C:\EB50	10\Proje 10\Proje	ctitest1. ctitest1.	.epj .eob		
				Co	ompile			Close	

Click "Tools" – "Compile ..." to compile this project.

To download the project to the Touch 510, click on the Windows "Start" button, then click on the "Program" button, then click on the "EasyBuilder" – "EasyManager" button. The following window will be displayed. Choose the correct COM No. on your PC (Normally is COM1), "115200 bps".

Connect the RS232 download cable (refer to section 4.4) between PC and Touch 500.



Click on "Jump To RDS" first, if OK., you can see the screen of the Touch 500 will change and wait for project download. Click on "Download" to start to download the MMI picture to the Touch 500.



If downloading is OK, You may choose to click on "Jump To Application" or reset the Touch 510T, and then connect another RS232 cable between Touch 510 and the I-8xx7 (refer to section 4.4).

Now, you may touch each icon on the Touch 510 to test. Have a good luck !



## 4.5: Access To Word & Integer Array Via Modbus

User can use the below functions to read/write word & integer arrays inside the ISaGRAF project. For more information about these functions, please refer to Appendix A.4.

ARY_N_R	Read one integer(4 byte, signed) from an integer array
ARY_N_W	Write one integer(4 byte, signed) to an integer array
ARY_W_R	Read one word(2 byte, signed) from an word array
ARY_W_W	Write one word(2 byte, signed) to an word array

Word and integer arrays built in the I-8xx7, I-7188EG, I-7188XG, uPAC-7186EG, iPAC-8447/8847 & Wincon-8xx7 controller occupy the same memory area, please use them carefully. Other softwares (HMI, OPC server, ...) running on the PC can access to these word and integer arrays via Modbus protocol. The valid network address for these arrays is from 5001 to 8072 for I-8xx7, I-7188EG & I-7188XG, while 10,001 to 19,216 for the W-8xx7 and their relation is listed in below table.

For the I-8xx7, I-7188EG, I-7188XG, uPAC-7186EG, iPAC-8447/8847:

Network Address (Decimal)	Word Array	Integer Array
5001	(1,1)	(1,1)
5002	(1,2)	
5003	(1,3)	(1,2)
5004	(1,4)	
8071	(12,255)	(6,256)
8072	(12,256)	

For the W-8xx7:

Network Address (Decimal)	Word Array	Integer Array
10001	(1,1)	(1,1)
10002	(1,2)	
10003	(1,3)	(1,2)
10004	(1,4)	
	•••	
19215	(36,255)	(18,256)
19216	(36,256)	

Note:

1. Network address 1 to 4095 for I-8xx7, I-7188EG/XG, uPAC-7186EG & iPAC-8447/8847, while 1 to 8191 for W-8xx7, can be defined by users, please refer to Section 4.1.

2. **Modbus address** in the physical transmission format is equal to **Network address** minus one (please refer to Chapter 5). So the valid Modbus address for word & integer arrays is from 5000 to 8071 for I-8xx7, I-7188EG/XG, uPAC-7186EG and iPAC-8447/8847, and 10000 to 19215 for W-8xx7.