Communication Driver -- DCON

Support Serial Comm. (**RS-232 / 485**) for PC, WinPAC, ViewPAC, XPAC-8000 , XPAC-8000-CE6 Embedded Controller with ICP DAS DCON modules (I-7K, I-8K, I-87K) and others.

Table of Contents

1. INTRODUCTION 2

2. GENERAL INFORMATION 3

2.1 Device Specifications 3

2.2 Network Specifications 3

2.3 Network Specifications 3

3. Driver Installation 5

3.1 Installing and Selecting Driver 5

3.2 Other Software Requirements 6

4. CONFIGURING THE DRIVER 7

4.1 Configuring the Communication Settings 7

4.2 Configuring the Driver Worksheets 8

4.2.1 Insert a Standard Worksheet 8

4.2.2 Configuring the Station and Header fields 10

4.2.3 Configuring the Tag and Address fields for General Modules 11

4.2.4 Configuring the Tag and Address fields for FRnet Module 13

4.2.5 Configuring the Tag and Address fields for I-8084W 14

4.2.6 Configuring the Tag and Address fields for I-8088W 15

4.2.7 Configuring the Tag and Address fields for DL-100T Series 17

4.2.8 Configuring the Tag and Address fields for HART I/O Module 18

4.2.9 Configuring the Tag and Address fields for DALI Gateway Series 19

4.3 Device Configuration 20

5. Executing the Driver 21

6. Improve DI/AI scan time 22

7. Using Multi-DCON Driver 23

8. Troubleshooting 24

9. ICP DAS InduSoft Resource 27

10. History Version 27

1. INTRODUCTION

The DCON driver enables communication between InduSoft Web Studio and most of ICP DAS modules from DCON families (I-7000, I-8000 and I-87K), in accordance with the characteristics covered in this document.

This document will help you to select, configure and execute the DCON driver, and it is organized as follows:

- **Introduction: Provide an overview of the driver documentation.**
- **General Information: Provide information necessary to identify all the required components (hardware and software) necessary to implement the communication and global characteristics about the communication.**
- **Driver Installation: Explain the procedures that must be followed to install the software and hardware required for the communication.**
- **Configuring the Driver: Provide the required information to configure the communication driver such as the different permutations for configuration and its default values.**
- **Executing the Driver: Explain the steps to test whether the driver was correctly installed and configured.**
- **Enhance DI/AI Scan Timing: Explain the way to enhance the scan time when using DI/AI modules**
- **Using Multi-DCON Driver: Explain how to add another DCON driver in InduSoft**
- **Troubleshooting: Supply a list of the most common error codes for this protocol and the procedures to fix them.**
- **ICP DAS InduSoft Resources: A list of useful links which provide varies demos and documents.**
- **History of versions: Provide a log of all the modifications done in driver.**

Note:

- **This document assumes that you have read the "Development Environment" chapter in Studio's Technical Reference Manual.**
- **This document also assumes that you are familiar with the Microsoft Windows NT/2000/XP/7/8 environment. If you are not familiar with Windows, then we suggest using the Help feature (available3 from the windows desktop Start menu) as you work through this guide.**

2. GENERAL INFORMATION

2.1 Device Specifications

- Manufacturer: ICP DAS Co., Ltd.
- Compatible Equipment - ICP DAS modules from DCON families (I-7000, I-8000 and I-87K).

Note:

- All analog modules must be configured to engineering units.
- This Driver version does not implement the CRC. The equipment must be configured not to use CRC check.

2.2 Network Specifications

- Device communication port**:** RS232 port
- Physical protocol**:** RS232/RS485
- Logic protocol**:** ASCII (Proprietary)
- Device Runtime software**:** DCON driver
- Specific PC Board**:** None

2.3 Network Specifications

- Operation System:
  - Microsoft Windows NT / 2000 / XP / 7 / 8
  - Microsoft Windows Embedded Standard 2009 (ICP DAS XP-8000 embedded controller)
  - Microsoft Windows CE 7.0 (ICP DAS IWS-x201 embedded controller)
  - Microsoft Windows CE 6.0 (ICP DAS XP-8000-CE6 embedded controller)
  - Microsoft Windows CE 5.0 (ICP DAS WinPAC / ViewPAC embedded controller)

The driver is composed of the following files:

- DCON.IN**I:** Internal files of the driver, it should not be modified by the user.
- DCON.MSG**:** This file contains the error messages for each error code. It is an internal file of the driver, and the user should not modify it.
- DCON.PDF**:** This document provides detailed documentation about the driver.

- DCON.DLL**:** This is the compiled library for the driver.

Note:

All the files above must be in the subdirectory /DRV of the Studio's installation direction.

Default Configuration:

- COM Port**:** 1
- Baud Rate**:** 9600
- Data Bits**:** 8
- Parity**:** None
- Stop Bits**:** 1
- Timeout**:** 200
- Checksum**:** 0 (No Checksum)

3. Driver Installation

3.1 Installing and Selecting Driver

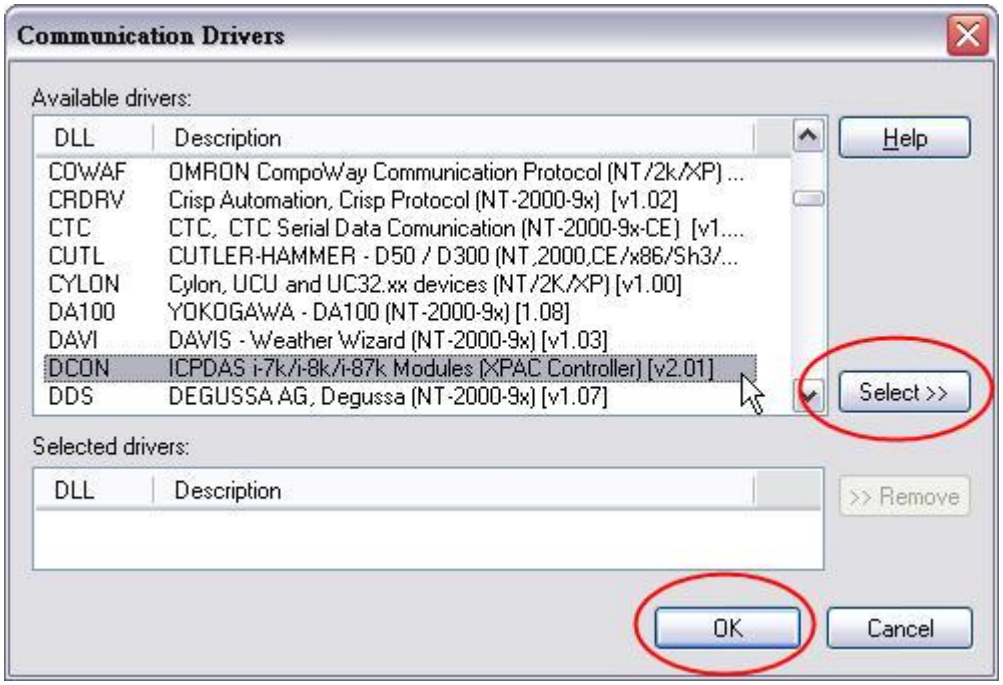
After install InduSoft Web Studio in XPAC-8000 or PC, **the DCON driver of ICPDAS is not installed automatically**. Therefore, users need to install the DCON driver first before using the DCON driver in IWS. The DCON driver setup file can be downloaded from the site: **http://www.icpdas.com/products/Software/InduSoft/indusoft_drivers.htm** .

For WinPAC-8000, ViewPAC-2xWx, and XPAC-8000-CE6 users, there is no need to install the DCON driver. The DCON driver is included inside the runtime folder. Simply download the InduSoft runtime folder from the site: **http://www.icpdas.com/products/Software/InduSoft/indusoft_download.htm** . When the DCON communication driver is installed in IWS, users just need to select the driver in the application and the steps are described as follows:

1. Execute the Studio and select the proper application.

2. From the main menu bar, select **Insert** → **Driver** to open the *Communication Drivers* dialog.

3. Select the **DCON** driver from the *Available Drivers* list, and then click the **Select** button.



4. When the **DCON** driver is displayed in the **Selected Drivers** list, click the **OK** button to close the dialog. The driver is added to the *Drivers* folder, in the *Comm* tab of the Workspace.

3.2 Other Software Requirements

It is not necessary to install any other software in the PC / XP-8000 / XP-8000-CE / WinPAC / ViewPAC to enable the communication between the host and the Device.

Note:

Special cautions must be taken when installing the physical hardware. Refer to the hardware manufacture documentation for specific instructions in this area.

4. CONFIGURING THE DRIVER

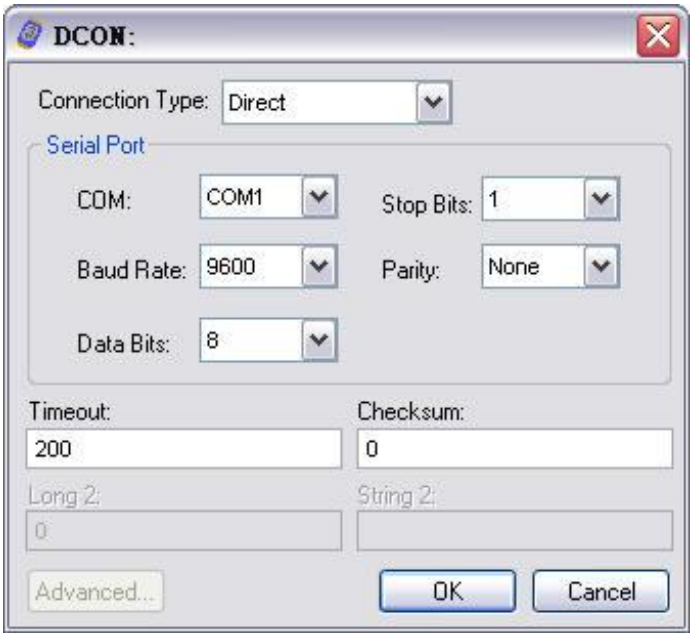
Once you have selected the DCON driver in Studio, you must properly configure it to communicate with your target devices. First, you must set the driver's communication settings to match the parameters set on the device. Then, you must build driver worksheets to associate database tags in your Studio application with the appropriate addresses for the devices.

4.1 Configuring the Communication Settings

The communication settings are described in detail in the "Communication" chapter of the Studio *Technical Reference Manual*, and the same general procedures are used for all drivers. Please review those procedures before continuing.

For the purposes of this document, only DCON driver-specific settings and procedures will be discussed here. To configure the communication settings for the DCON driver:

1. In the *Workspace* pane, select the *Comm* tab and then expand the *Drivers* folder. The DCON driver is listed here as a subfolder.
2. Right-click on the *DCON* subfolder and then select the **Settings** option from the pop-up menu. The *DCON: Communication Parameters* dialog is displayed:



3. Verify the *Serial Port* settings, and change them if necessary.
4. Configure the additional driver-specific settings, as described in the following table:

Parameter	Default Value	Valid values	Description
COM	COM1	COM1 to COM256	Serial port uses to communication with the devices
Baud Rate	9600	110 to 115200bps	Communication rate of data
Data Bits	8	5 to 8	Number of data bits used in the protocol
Stop Bits	1	1 or 2	Number of stop bits used in the protocol
Parity	None	Even, odd, none, space or mark	Parity of the protocol
Timeout	200	0~2147483647	Time unit is 1 ms.
Checksum	0	0 or 1	**0:** No checksum **1:** Checksum

Table 4.1.1

Note: These Parameters must be just the same as the configured on the ICP DAS device.

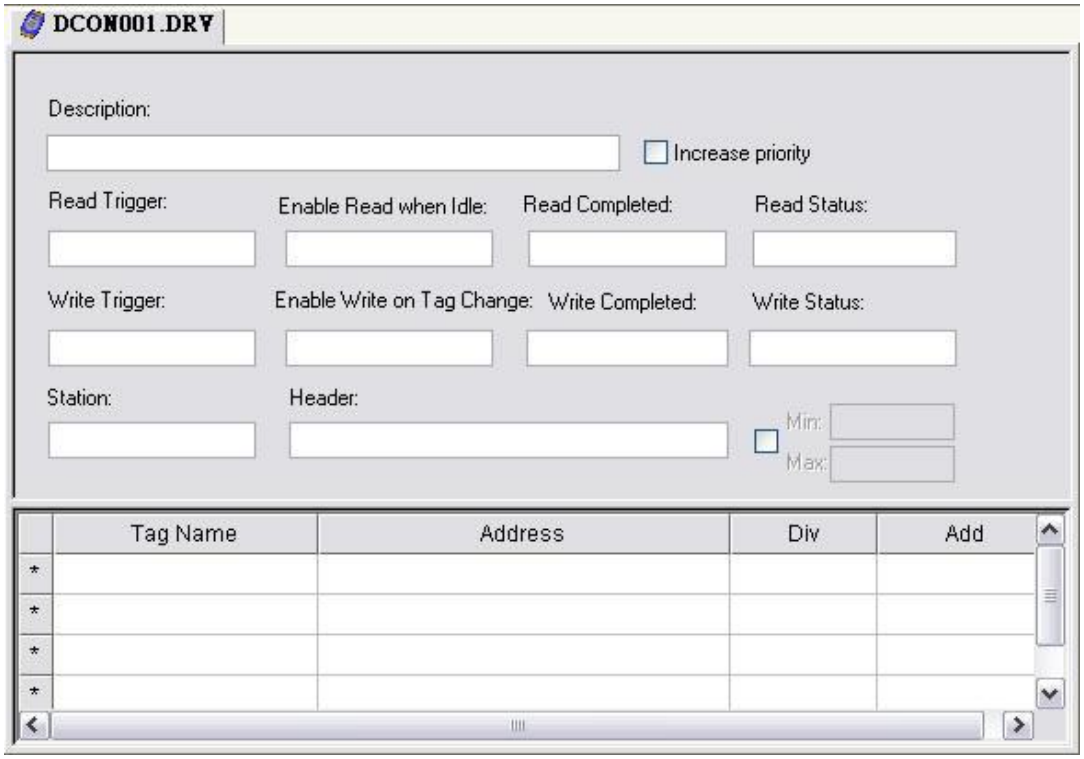
4.2 Configuring the Driver Worksheets

4.2.1 Insert a Standard Worksheet

The configuration of these worksheets is described in detail in the “Communication” chapter of the Studio *Technical Reference Manual*, and the same general procedures are used for all drivers. Please review those procedures before continuing.

1.In the *Comm* tab, open the *Drivers* folder and locate the *DCON* subfolder.
2.Right-click on the *DCON* subfolder, and then select **Insert** from the pop-up menu:

A new DCON driver worksheet is inserted into the *DCON* subfolder, and the worksheet is opened for configuration:



NOTE:

- Most of the field on this sheet are standard for all drivers; see the “Communication” chapter for the Technical Reference Manual for more information on configuring these fields. However, the Station and I/O Address field use syntax that is specific to the DCON driver.
- Users can only apply the tag name up to 100 items in the same worksheet.

4.2.2 Configuring the Station and Header fields

Parameter	Default Value	Valid value	Description
Station			Not used
Header	AI	See next table	Define the type of variable to be read or written from or to the device.

Table 4.2.2.1

- **Station field**: Not used in DCON driver.
- **Header field**: The Header field defines the type of variables that will be read or written from or to the device. It complies with the syntax**:** <Module's Type>. After editing the field Header, the system will check if it is valid or not. If the syntax was incorrect, the default value (AI) will be automatically placed in the field.

You can also specify an indirect tag (e.g. **{header}**), but the tag that is referenced must follow the same syntax and contain a valid value.

Header Field Reference Table			
Type	**Sample of syntax (Header)**	**Valid range of initial Address**	**Comment**
Read Digital Input/Output	DI	Any	
Read/Write Digital Output	DO	Any	
Read Analog	AI:**x**	Any	The ICP DAS device channel must be configured to engineering units
Write/Read Analog	AO:**x**	Any	The ICP DAS device channel must be configured to engineering units
Read/Set Counter	Counter	Any	
Read/Set DI Counter value	DICounter	Any	
Write Command	SendCmd: **x**	**SendCmd String**	Send/Recv string command from/to the device
Communicate with FRnet modules	FRNET	Any	

PWM modules	PWM	Any	
DL-100T Series	DL	Any	
HART I/O Series	HTCmdStr	Any	Send/Recv HART Command String
HART I/O Series	HTCmdData	Any	Send HART Command String and Receive HART Command Data
DALI Gateway	DGW	Any	Access DALI device via DALI gateway (like DGW-521)
XW-Board	XWDI XWDO XW**xxx**AI XW**xxx**AO	Any	For WP-5xxx controller expansion I/O board (like XW-107/110/304…) **xxx** will be as below: 107/110/304/310/310C/…

Table 4.2.2.2

Note:

1. For header type AI or AO

The x symbol is used to configure a module's input/output range in the XP-8000's slot. The following description lists all the x values. The value 0 ~ 3 will be configured as +/- 10V and the value 4 is +/- 20mA. If users just use AI or AO head type without x value, the default x value is 0.

0: +/- 10V

1: +/- 5V

2: +/- 2.5V

3: +/- 1.25V

4: +/- 20mA

2. For header type SendCmd

For Slot Module, x means slot number of module. For COM Port Module, use without x in header field.

Note:

Always creates two different driver worksheets for Input and Output modules.

4.2.3 Configuring the Tag and Address fields for General Modules

- **Tag Name Field**

The body of the driver worksheet allows you to associate each tag to its respective address in the device.

In the column **Tag Name**, you must type the tag from your application database. This tag will receive or send values from or to the device.

■ Address Field

Please fill in module setting value into address field with the following sequence:

< Address > : < Slot > : < Total_In Channel > : < Total_Out Channel > : < Channel No. >

Address (Hex) : Module's Address in the network **(Range from -1 to 0xFF)**.
[1] **-1** for **I-8K** / **XW-board**(For WP-5xxx controller)
[2] **0** for **I-87K** modules in slot.

Slot : The slot number where the Module is plugged in the backplane.
[1] **-1** for **I-7000** modules or **I-87K modules on I-87K expansion unit**.
[1] **0** for **XW-board**(For WP-5xxx controller)

Total_In Channel : The total number of input channel for a module.

Total_Out Channel : The total number of output channel for a module.

Channel No. : Channel's Number to be read/written from/to the module.

The value of Address and Slot in Address Field			
Module Location		**Address_value**	**Slot_value**
Slot Modules	**I-8K**	**-1**	**>=0**
	I-87K	**0**	**>=0**
Com Port Modules		**>0**	**-1**

Table 4.2.3.1

Example of Addressing Configuration		
Slot Modules (XW-Board)	**Header Field**	**Address Field**
Read DI (XW107) **Slot:0 / Ch:0**	**XWDI**	**-1:0:8:8:0**
Write DO (XW310) **Slot:0 / Ch:1**	**XWDO**	**-1:0:3:3:1**
Read AI (XW310) **Slot:0 / Ch:2**	**XW310AI**	**-1:0:4:2:2**
Read AO (XW310) **Slot:0 / Ch:1**	**XW310AO**	**-1:0:4:2:1**
Slot Modules (I-8K)	**Header Field**	**Address Field**

Read DI (I-8053P) Slot:2 / Ch:0	DI	-1:2:16:0:0
Read/ Write DO (I-8057W) Slot:1 / Ch:1	DO	-1:1:0:16:1
Read AI (I-8017HW) Slot:6 / Ch:2	AI	-1:6:8:0:2
Write AO (I-8024W) Slot:7 / Ch:3	AO	-1:7:0:4:3
Slot Modules (I-87K)	**Header Field**	**Address Field**
Read DI (I-87040W) Slot:5 / Ch:31	DI	0:5:32:0:31
Read/ Write DO (I-87041W) Slot:6 / Ch:30	DO	0:6:0:32:30
Read AI (I-87017W) Slot:4 / Ch:4	AI	0:4:8:0:4
Write AO (I-87024W) Slot: 7 / Ch: 0	AO	0:7:0:4:0
Send/Recv Cmd to DO I-87041W Slot:6	SendCmd:6	Command String (like $00M)
ComPort Modules	**Header Field**	**Address Field**
Read DI (I-7041) Addr:1 / Ch:5	DI	1:-1:14:0:5
Read/ Write DO (I-7042) Addr:2 / Ch:6	DO	2-1:0:13:6
Read AI (I-7012) Addr: 4 / Ch: 0	AI	4:-1:1:0:0
Write AO (I-7021) Addr: 3 / Ch: 0	AO	3:-1:0:1:0
Send/Recv Cmd to DO (I-7041) Addr:1	SendCmd	Command String (like $06M)

Table 4.2.3.2

4.2.4 Configuring the Tag and Address fields for FRnet Module

- **Tag Name Field**

The body of the driver worksheet allows you to associate each tag to its respective address in the device. In the column **Tag Name**, you must type the tag from your application database. This tag will receive or send values from or to the device.

- **Address Field**

Please fill in module setting value into address field with the following sequence:

<Address> : <Slot> : <Port Number> : <Group Number> : <Used Channel>

Address (Hex) : Module's Address in the network **(Range from -1 to 0xFF)**. **-1** for I-8K and **0** for I-87K modules in slot. Please refer to table 4.2.3.1.

Slot : The slot number where the Module is plugged in the backplane. **-1** for I-7000 modules or I-87K modules on I-87K expansion unit. Please refer to table 4.2.3.1.

Port Number : The port number of FRnet module in which connects to FRnet **(Range: 0 or 1)**.

Group Number : Group number which is corresponding to an FRnet module **(Range: 0 to 7)**.

Used Channel : Channel's Number to be read/written from/to the module.

Example of Addressing Configuration		
Slot Modules (I-8K)	**Header Field**	**Address Field**
Read FRnet DI Channel via I-8172W Slot:3 / Port:1 / Group:0 / Ch:5	**FRnet**	**-1:3:1:0:5**
Write FRnet DO Channel via I-8172W Slot:2 / Port:0 / Group:5 / Ch:8	**FRnet**	**-1:2:0:5:8**

Table 4.2.4.1

4.2.5 Configuring the Tag and Address fields for I-8084W

- **Tag Name Field**

The body of the driver worksheet allows you to associate each tag to its respective address in the device. In the column **Tag Name**, you must type the tag from your application database. This tag will receive the counter values from the device.

- **Address Field**

Please fill in module setting value into address field with the following sequence:

<Address> : <Slot> : <Channel Mode> : <Unreserved> : <Used Channel>

Address (Hex) : **-1** for I-8K module

Slot : The slot number where the Module is plugged in the backplane **(Range: 0 to 7)**.

Channel Mode : The total number of input channel for a module.

Mode 0: Dir/Pulse Counter
Mode 1: Up/Down Counter
Mode 2: Frequency
Mode 3: Up Counter
Mode 4: A/B Phase Counter

Unreserved : **0** (Not use in FRnet module.)

Channel Number : Channel's Number to be read from the module.

Example of Addressing Configuration		
Slot Modules (I-8K)	**Header Field**	**Address Field**
Read Counter (I-8084W) **Slot: 5 / Channel Mode: 0 / Channel:3**	**Counter**	**-1:5:0:0:3**

Table 4.2.5.1

4.2.6 Configuring the Tag and Address fields for I-8088W

■ Tag Name Field

- **PWM Output Mode:**

The body of the driver worksheet allows you to associate each tag to its respective I-8088W API function. In the column **Tag Name**, you must type the tag from your application database. This tag must be configured as **String Type** and will send its value to an API function according to Address field defined.

The tag value has the following sequence:
<Variable 1> : < Variable 2> : < Variable 3> : < Variable 4>

Variable 1 : The **slot number** where the Module is plugged in the backplane **(Range: 0 to 7)**.
Variable 2 : The number of **channel** the function will be called.
Variable 3 : The value needed to be filled in the third parameter of the respective function. If there is no

parameter needed, please fill **0** instead.

Variable 4 : The value needed to be filled in the forth parameter of the respective function. If there is no parameter needed, please fill **0** instead.

Example of Tag Value in PWM Output Mode	
I-8088W Function	**Tag Value (String Type)**
pac_i8088W_SetPWMDutyDeci( ); **Slot: 2, Channel: 3, Frequency: 10000, Duty Cycle: 10**	**2:3:10000:10**
pac_i8088W_PWMStart( ); **Slot: 1, Channel: 2**	**1:2:0:0**

Table 4.2.6.1

- **Read DI Value from I-8088W:**

  The body of the driver worksheet allows you to associate each tag to its respective address in the device. In the column **Tag Name**, you must type the tag from your application database. This tag will receive the DI values from the I-8088W.

■ Address Field

- **PWM Output Mode:**

  Please fill in the function type ID from **0 ~ 10**.

Type ID and Function Reference Chart	
Function Type ID	**Function**
1	**pac_i8088W_SetPWMDutyDeci( );**
2	**pac_i8088W_SetPWMDutyCoutn( );**
3	**pac_i8088W_SetPWMCountMode( );**
4	**pac_i8088W_BurstCount( );**
5	**pac_i8088W_PWMStart( );**
6	**pac_i8088W_PWMStop( );**
7	**pac_i8088W_SetSyncChannel( );**
8	**pac_i8088W_SyncStart( );**

9	pac_i8088W_SyncStop( );
10	pac_i8088W_SetHardwareTrigChannel( );

Table 4.2.6.2

- **Read DI Value from I-8088W:**

  Please fill in module setting value into address field with the following sequence:

  <Address> : <Slot> : <Channel Mode> : <Total_Out Channel> : <Used Channel>

  Address (Hex) : -1 for I-8088W module.
  Slot : The slot number where the Module is plugged in the backplane (Range 0 ~7).
  Total_In Channel : 0 (not use for I-8088W)
  Total_Out Channel : 0 (not use for I-8088W)
  Channel Number : DI Channel's Number to be read from the module.

Example of Addressing Configuration		
Slot Modules (I-8K)	**Header Field**	**Address Field**
Read DI (I-8088W) **Slot: 5 / Channel:3**	**PWM**	**-1:5:0:0:3**

Table 4.2.6.3

4.2.7 Configuring the Tag and Address fields for DL-100T Series

■ Tag Name Field

The body of the driver worksheet allows you to associate each tag to its respective address in the device. In the column **Tag Name**, you must type the tag from your application database. This tag will receive the counter values from the device.

■ Address Field

Please fill in module setting value into address field with the following sequence:

<Address> : <Data Type>

Address (Hex) : Module's Address in the network **(Range from 0x01 to 0xFF)**.
Data Type : The type of value to be read **(Range: 0 to 2)**.

Type 0: Temperature value in degrees Celsius

Type 1: Temperature value in degrees Fahrenheit

Type 2: Humidity value

Example of Addressing Configuration		
Slot Modules (I-8K)	**Header Field**	**Address Field**
Read DL (DL-100T485) **Address: 5 / Data Type: 0**	**DL**	**5:0**

Table 4.2.7.1

4.2.8 Configuring the Tag and Address fields for HART I/O Module

■ Tag Name Field

The tag value is used to receive the HART device response data.

If the "**Header**" field is filled with "**HTCmdStr**", it means the tag will be the received HART command string.

If the "**Header**" field is filled with "**HTCmdData**", it means the tag will be the received HART command data.

■ Address Field

Please fill in module setting value into address field with the following sequence:

<Address> : <Slot> : <Total_In Ch> : <Total_Out Ch> : <Ch No.> : <HTCmd> : <HTDataIdx>

Address (Hex) : Module's Address in the network **(Range from -1 to 0xFF)**. **-1** for I-8K and **0** for I-87K modules in slot.

Slot : The slot number where the Module is plugged in the backplane. **-1** for I-7000 modules or I-87K modules on I-87K expansion unit.

The value of Address and Slot in Address Field			
Module Location		**Address_value**	**Slot_value**
Slot Modules	**I-8K**	**-1**	**>=0**
	I-87K	**0**	**>=0**
Com Port Modules		**>0**	**-1**

Table 4.2.8.1

Total_In Channel : The total number of input channel for a module.

Total_Out Channel : The total number of output channel for a module.

Channel No. : Channel's Number to be read/written from/to the module.

HTCmd : HART command string for sending and the format is as below.

Preamble_Delimeter_Address_Command_ByteCnt_DataArray

(like: HART Cmd3 => **05_82_16850B0A42_03_00**)

HTDataIdx : HART command data index for receiving. The value is as below.

Example of HTCmd and HTDataIdx configuration		
"Header" Field	**<HTCmd>**	**<HTDataIdx>**
HTCmdStr	**HART send command string**	**0: HART receive command string**
HTCmdData	**HART send command string**	**0: HART receive data0** **1: HART receive data1** **2: HART receive data2** **…** **For example: (For HART Command 3)** **0:Current ; 1:PV ; 2:SV ; 3:TV ; 4: QV**

Table 4.2.8.2

[For Example]

Header = HTCmdData

Address = 0:1:8:0:0:05_82_16850B0A42_03_00:0

<HTCmd> => **05_82_16850B0A42_03_00**

<HTDataIdx> => **0**

=> It will send HART command 3 string and receive the "Current" data in the InduSoft tag.

4.2.9 Configuring the Tag and Address fields for DALI Gateway Series

■ Tag Name Field

The tag value is used to the DALI device.

If **DALI_DataType**=0, it means the value will be **Lamp Power Value**.

If **DALI_DataType**=1, it means the value will be **DALI Command Code**.

■ Address Field

Please fill in module setting value into address field with the following sequence: (DL means DALI)

<Mod_Addr> : <DALI_AddrType> : <DALI_Addr> : <DALI_DataType> : <DALI_CmdIdx>

Mod_Addr (Hex) : Module's Address in the network **(Range from 0x01 to 0xFF)**.

DALI_AddrType : 0=> DALI **single** device address.
1=> DALI **group** device address.
2=> DALI **broadcast** address.

DALI_Addr : if "DALI_AddrType" equals 0, then this value will be **between 0 and 63**.
if "DALI_AddrType" equals 1, then this value will be **between 0 and 15**.
if "DALI_AddrType" equals 2, then this value will be **0**.

DALI_DataType : 0=> The tag value will be **Lamp Power Value**.
1=> The tag value will be **DALI Command Code**.

DALI_CmdIdx : if "DALI_DataType" equals 0, then this value will be **blank (No Use)**.
if "DALI_DataType" equals 1, then this value will be **DALI Command Index**.

Example of DALI gateway configuration		
Module Location	**Header Field**	**Address Field**
Set DALI Single Device Lamp_Power via DGW-521 (Mod_Addr:1 / DALI_Addr:2)	**DGW**	**1:0:2:0**
Set DALI Group Device Lamp_Power via DGW-521 (Mod_Addr:2 / DALI_Group:3)	**DGW**	**2:1:3:0**
Set DALI Broadcast Lamp_Power via DGW-521 (Mod_Addr:5)	**DGW**	**5:2:0:0**
Execute DALI Command 144 via DGW-521 (Mod_Addr:3 / DALI_Addr:4)	**DGW**	**3:0:4:1:144**

Table 4.2.9.1

- **Station Field**

It is used to trig the assigned DALI query command for the reading driver sheet. If the value of the station field is 0, then all query command in reading driver sheet will be executed. Users can use **{rCmdNo}** to execute the assigned DALI query command. (rCmdNo is a integer tag)

4.3 Device Configuration

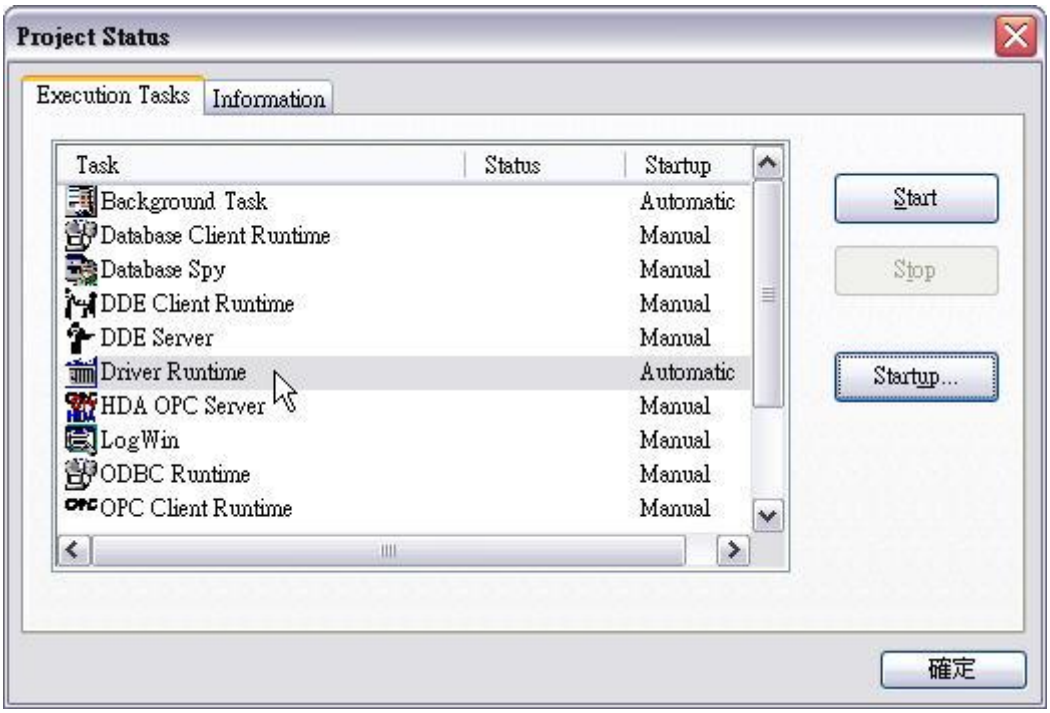
The device parameter (baud rate, stop bits, etc) must be matched with the settings configured in the Communication Parameters of the DCON driver.

5. Executing the Driver

By default, Studio will automatically execute your selected communication driver(s) during application runtime. However, you may verify your application's runtime execution settings by checking the *Project Status* dialog.

To verify that the communication driver(s) will execute correctly:

1. From the main menu bar, select **Project** → **Status**. The *Project Status* dialog displays:



2. Verify that the *Driver Runtime* task is set to **Automatic**.

- If the setting is correct, then proceed to step 3 below.
- If the **Driver Runtime** task is set to **Manual**, then select the task and click the **Startup** button to toggle the task's *Startup* mode to **Automatic**.

3. Click **OK** to close the *Project Status* dialog.

4. Start the application to run the driver.

6. Improve DI/AI scan time

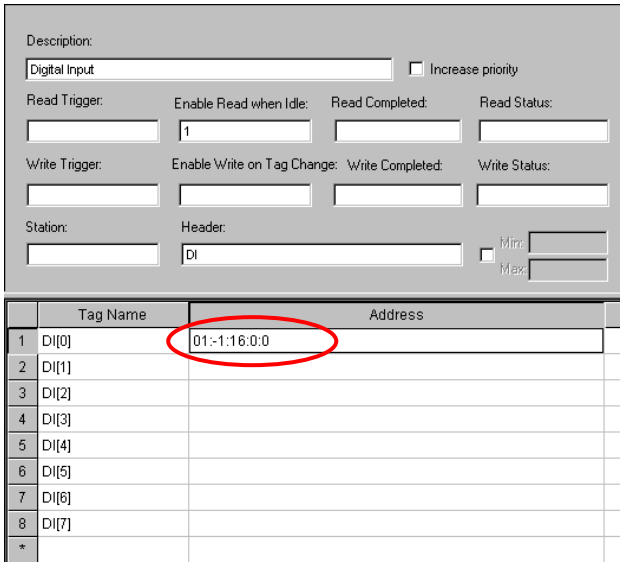
The DCON driver provides a simple method to improve multi-channel scan time for a DI/AI module. If a module has many channels, this method can decrease the communication times to obtain all channels values from this module. Users only need to configure the address for the first channel of this module, and give the others null address as figure below.

(1) In "**Fast mode**" **:**

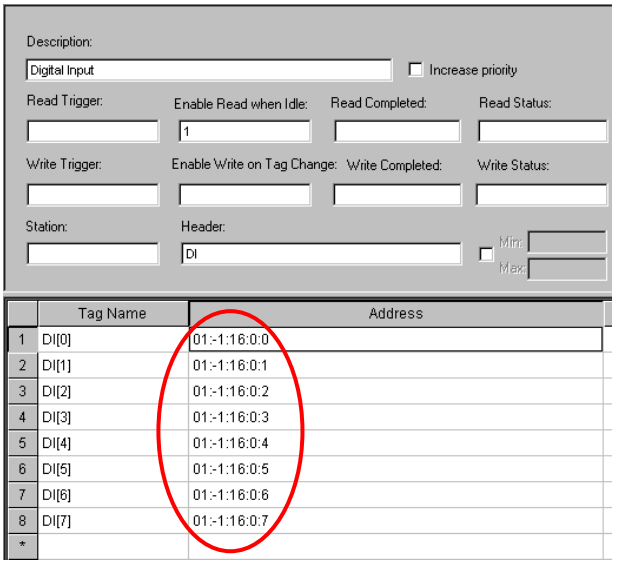
Users just need to configure the first channel of this module and keep the other address field of this module blank. (Refer to Figure 7-1) DCON Driver will obtain all channels' (Ch0 ~ Ch7) data at once.

(2) In "**Normal mode**" **:**

Users need to configure all the channels of this module. (Refer to Figure 7-2) DCON Driver will obtain each channel's value individually.



(**Fast Mode**)



(**Normal Mode**)

7. Using Multi-DCON Driver

When users want to use two or more DCON drivers in InduSoft, please follow the steps list below.

1. Duplicate **DCON.dll**, **DCON.ini**, and **DCON.msg** which located in the Bin folder under the installation path. Ex: C:\Program Files\Indusoft Web Studio v7.1\Drv
2. After these three files duplicated, modify the filename. Please follow rules below for naming the filename of a new driver.
   i. The filename cannot exceed **5 characters**.
   ii. The **last character** of the filename must be **alphabet**.

   For Example:
   - DCONA, DCONB…=> **OK**
   - DCONABC, DCON1, DCON2…=>**NG**

8. Troubleshooting

After each attempt to communicate using this driver, the tag configured in the field **Read Status or Write Status** will receive the error code regarding the kind of failure that occurred. The error messages are:

Error Code of General Module Reference Table			
Error Code	**Description**	**Possible causes**	**Procedure to solve**
0	NoError	Communication without problems.	None required
1	Invalid Header	An invalid Header	Type a valid Header
2	Invalid Address	An invalid Address	Type a valid Address
0x10000+1	Slot registered error		
0x10000+2	Slot not registered error		
0x10000+3	Unknown Module		
0x10000+4	Module doesn't exist		
0x11000+1	EEPROM accesses invalid address		
0x11000+2	SRAM accesses invalid address		
0x12000+1	The input value is invalid		
0x12000+2	The WDT doesn't exist		
0x12000+3	The WDT init error		
0x13000+1	Create interrupt's event failure		
0x14000+1	Uart check sum error		
0x14000+2	Uart read timeout		
0x14000+3	Uart response error		
0x14000+4	Uart under input range		
0x14000+5	Uart exceed input range		
0x14000+6	Uart open filed		
0x15000+1	IO card does not support this API		
0x15000+2	API unsupported this IO card		
0x15000+3	Slot's value exceeds its range		
0x15000+4	Channel's value exceeds its range		
0x15000+5	Gain's value exceeds its range		
0x15000+6	XP-8000 unsupported the interrupt mode		

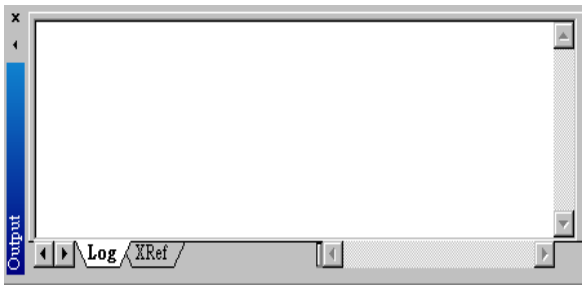
Error Code of I-8084W Reference Table

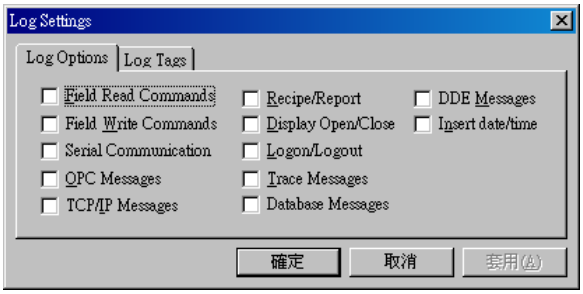
Error Code	Description	Possible causes	Procedure to solve
0	NoError		
-1	ID ERROR		
-2	SLOT OUT RANGE		
-3	CHANNEL OUT RANGE		
-4	SELECT CHANNEL ERROR		
-5	ADDRESS ERROR		
-10	DATA ERROR		
-15	Timeout		

Error Code of I-8088W Reference Table			
Error Code	**Description**	**Possible causes**	**Procedure to solve**
0	NoError		
-1	ID ERROR		
-2	SLOT OUT RANGE		
-3	CHANNEL OUT RANGE		
-4	SELECT CHANNEL ERROR		
-5	HI DUTY OUT RANGE		
-6	LO DUTY OUT RANGE		

Note:

1. The results of the communication may be verified in the **output** Window of the Studio's environment. To set a log of events for **Field Read Commands**, **Field Write Commands** and **Serial Communication** click with the right button of the mouse on the output window and choose the option setting to select these log events.
2. When testing under a Windows CE target, you can enable the log at the unit (Tools/Logwin) and verify the celog.txt created at the target unit.





When you are not able to establish the communication with the PLC, you should first of all establish the communication between the PLC Programming Tool and the PLC. Very frequently the communication it is not possible due to a hardware or cable problem, or due an error or lack of configuration at the PLC. Only after the communication between the PLC Programming Software and the PLC is working fine, you can test again the supervisory. When testing the communication with the Studio, you should first use the application sample described at item 7, instead of the new application that you are creating.

If is required to contact technical support, please have the following information available:

- Operating System (type and version): To find this information use the Tools/System Information option
- Project information: It is displayed using the option Project/Status from the Studio menu
- Driver version and communication log: Available from Studio Output when running the driver
- Device model and boards: please refer to hardware manufacture’s documentation

9. ICP DAS InduSoft Resource

ICPDAS has work with InduSoft for many years and we provide more InduSoft solutions to help users solve InduSoft problems quickly and develop InduSoft project easily. The following InduSoft resources provided by ICPDAS are free, please refer to the ICPDAS InduSoft Web Site：

(1) ICPDAS Indusoft Introduction:
http://www.icpdas.com/products/Software/InduSoft/indusoft.htm

(2) ICPDAS Indusoft News:
http://www.icpdas.com/products/Software/InduSoft/News.html

(3) ICPDAS Indusoft Demos:
http://www.icpdas.com/products/Software/InduSoft/download_DP.htm

(4) ICPDAS InduSoft Package Tool:
http://www.icpdas.com/products/Software/InduSoft/package.htm

(5) ICPDAS InduSoft FAQ:
(English)
http://www.icpdas.com/download/software/InduSoft/InduSoft_FAQ_Eng.pdf
(Chinese)
http://www.icpdas.com/download/software/InduSoft/InduSoft_FAQ_CT.pdf

10. History Version

Ver.	Author	Date	Description of changes
2.01	Eugene Chen	2009/04/25	First driver version
2.02	Eugene Chen	2009/09/15	Add module **I-8172W**, **I-8084W**, and **I-8088W**
2.13	Edward	2014/06/13	Support **I-87H17W** (HART AI module)
2.14	Edward	2014/06/30	Support **DGW-521** (DALI Gateway)
2.16	Ian	2015/01/06	Support **XW-board** for WP-5xxx controller I/O expansion. (Page.11)