**User Manual**

**Private Server To ODBC for WISE-4000 Series**

**(.NET version)**

CONTENTS

[1. Introduction 4](#_Toc468291298)

[1.1. About This Manual 4](#_Toc468291299)

[1.2. Organization of This Manual 4](#_Toc468291300)

[2. Installations 7](#_Toc468291301)

[3. Configuration 10](#_Toc468291302)

[3.1 Private Server 10](#_Toc468291303)

[3.2 Log To ODBC 13](#_Toc468291304)

[4. Troubleshooting 20](#_Toc468291305)

[4.1. Private Server Start-up error 20](#_Toc468291306)

[4.2. No log received in Private Server 26](#_Toc468291307)

[5. Database generation and viewer 27](#_Toc468291308)

[Appendix A: I/O data header in CSV format 31](#_Toc468291309)

[Appendix B: System data header in CSV format 32](#_Toc468291310)

[Appendix C: Add new data columns in database 33](#_Toc468291311)

**Chapter** **1**

# Introduction

## About This Manual

This document describes the installation and usage of sample Private Server To ODBC (.NET version) for Advantech WISE-4000 series IoT Wireless I/O Module products.

For WISE-4000 series, system log could be sent to a private log server instead of a public log server (for example: DropBox). A private log server is a web server which accepts a client’s log via http or https protocol. These logs include Push Notification Logs(IO data and System Event, both in JSON format) and Upload files(IO data and System Event, both in csv format). For parsing csv data, we create Log To ODBC sample code that parses csv files and store data value in database, for syntax consistency‬, we use ODBC (Open Database Connectivity) to store data value in SQLite or Microsoft SQL server. In this document, we will describe the steps of building a simple Private Log Server and Log Database by using Microsoft Visual Studio 2008.

**Note:** Only .csv format file is parsed in this sample, other format (ex: json) is not processed.

## Organization of This Manual

This user manual is divided into the following sections:

* Introduction
* Installations
* Configuration of Private Server
* Troubleshooting

**Introduction** This section gives the user a basic idea of this manual.

**Installations** This section provides instructions on ODBC preparation and how to compile and install Private Server To ODBC.

**Configuration**

This section gives the new user a walk-through in configuring Private Server To ODBC.

**Troubleshooting**

This section provides instructions on how to troubleshooting for operation mistakes or errors.

**Database viewer**

This section gives the database view that can see the generation SQLite db file of Log

To ODBC.

**Chapter** **2**

# Installations

Before running Private Server To ODBC sample program, please follow below steps to check your PC has SQL server ODBC driver or SQLite3 ODBC Driver.

1. On the Start menu, click Control Panel.
2. In Control Panel, click Administrative Tools.
3. In Administrative Tools, click Data Sources (ODBC).

In Data Sources page, you will see ODBC driver list in driver tab, if driver “SQL Server” is not exists, here is the ODBC driver download page for SQL server.

<https://msdn.microsoft.com/library/mt703139.aspx>

If “SQLite3 ODBC Driver” is not exists, here is ODBC driver for SQLite.

<http://www.ch-werner.de/sqliteodbc/>

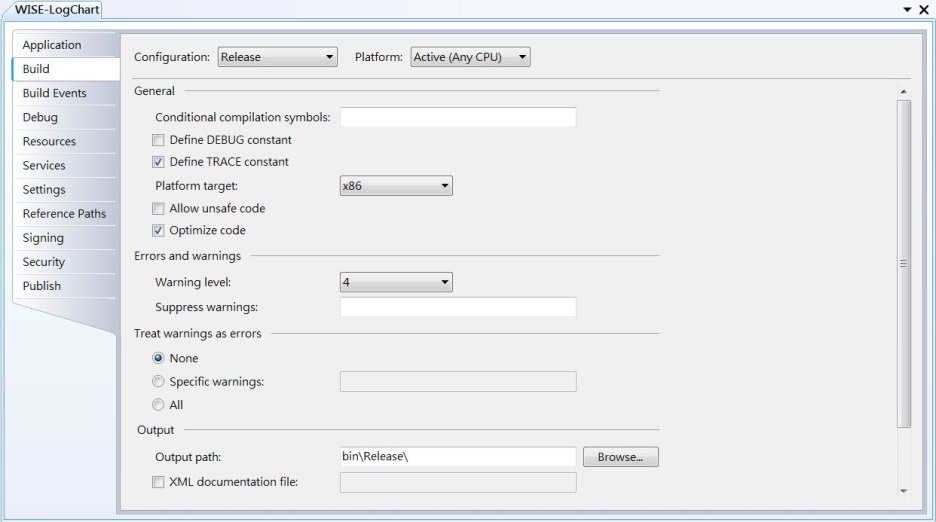
**Note:** Because of build code environment, you have to install SQLite ODBC driver “**sqliteodbc.exe**”, not “**sqliteodbc\_w64.exe**”.

You also have to check the remote connection has enabled on remote SQL server, there are more details about remote connections setting as below.

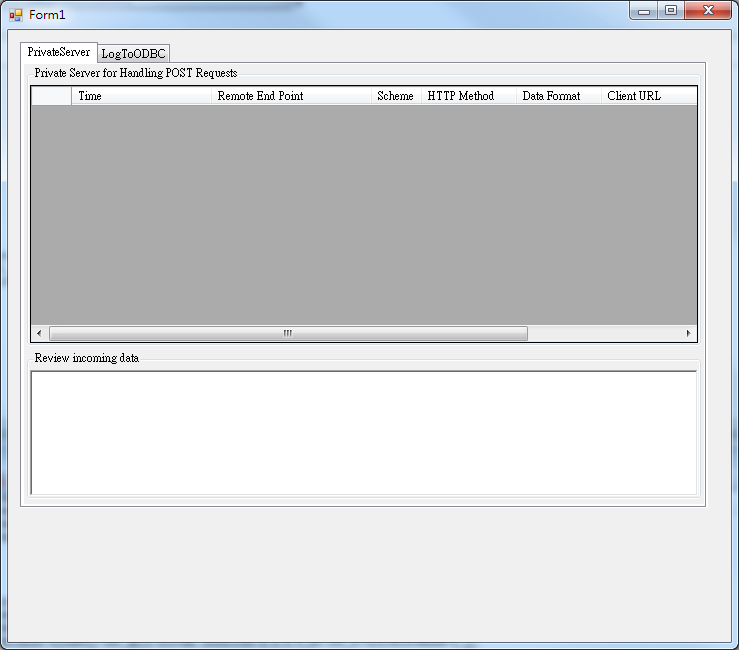
<https://blogs.msdn.microsoft.com/walzenbach/2010/04/14/how-to-enable-remote-connections-in-sql-server-2008/>

Finally, execute Microsoft Visual Studio 2008 or above version on your PC. Please follow below steps to compile Private Server And ODBC sample program.

1. Execute Microsoft Visual Studio 2008.
2. On the menu bar, choose File🡺Open🡺Project/Solution and navigator to the folder where your Private Server And ODBC program is located.
3. On the menu bar, choose Project🡺WISE- Private Server And ODBC Properties🡺 Build🡺 Platform target, please set “x86”(This is because we use SQLite x86 version dll library), you can refer below figure



1. On the menu bar, choose Build🡺 Build Solution.
2. Execute program by choose Debug🡺Start Debug.
3. You will see below figure.



Note: If you have problem when executing program, please refer Chapter 4 for trouble shooting.

**Chapter** **3**

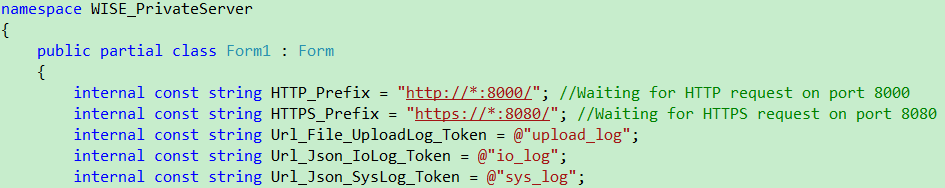
# Configuration

## Private Server

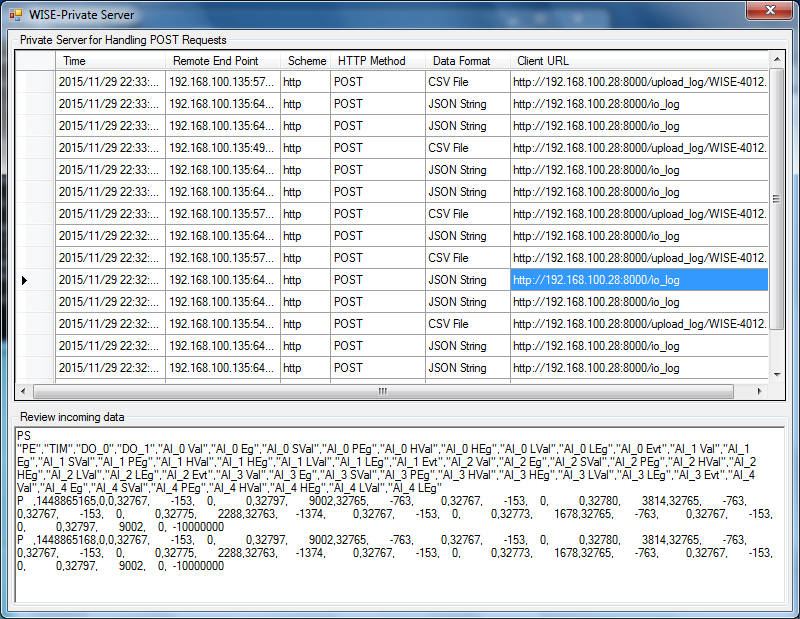
If you finish installing software described in Chapter 2, please proceed following steps to configure sample private server.

Several parameters of Private Server sample program could be modified, please open Form1.cs and refer to below table and figure:

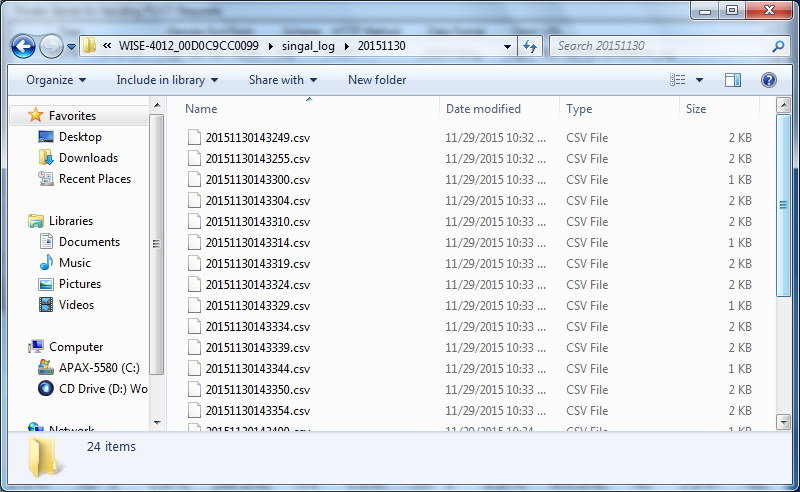
|  |  |
| --- | --- |
| Variable | Description |
| HTTP\_Prefix | Listen Port of Private Server HTTP Web (default 8000). |
| HTTPS\_Prefix | Listen Port of Private Server HTTPS Web (default 8080). |
| Url\_File\_UploadLog\_Token | Private Server Web URL which receives WISE device’s data file (csv format) upload. |
| Url\_Json\_IoLog\_Token | Private Server Web URL which receives WISE device’s push notification of IO data (JSON format). |
| Url\_Json\_SysLog\_Token | Private Server Web URL which receives WISE device’s push notification of System event (JSON format). |



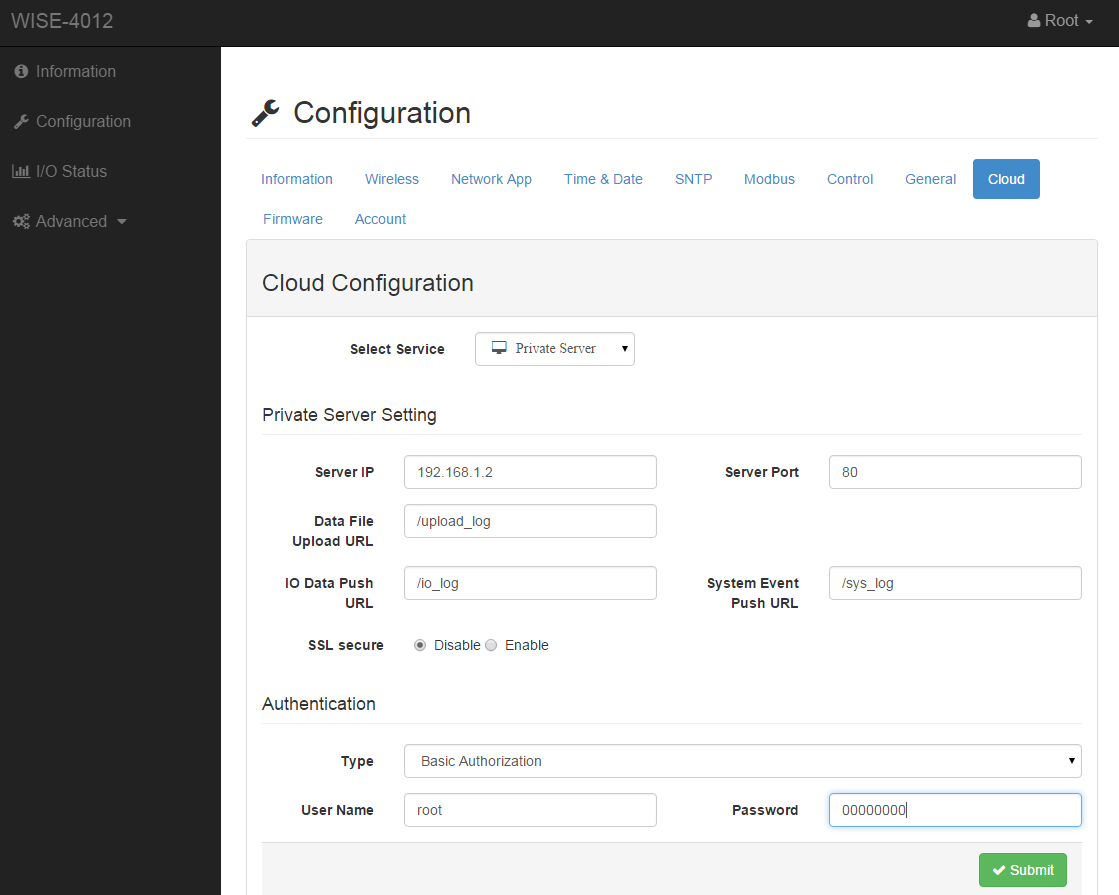
If your configuration on WISE device is correct, you will see below figure which shows logs are received correctly.



The Push notification log will be saved in the path where sample program is located. Following figure shows Push notification files received by Private Server.



Please make sure you have performed the corresponding Private Server configuration in WISE device. Please refer to the following figure:



Note: If you want to execute Private Server with HTTPS enabled, please refer to 「How to configure a port with an SSL certificate」document which is located in “WISE-PrivateServerAndLogToODBC\SSL Cert” folder.

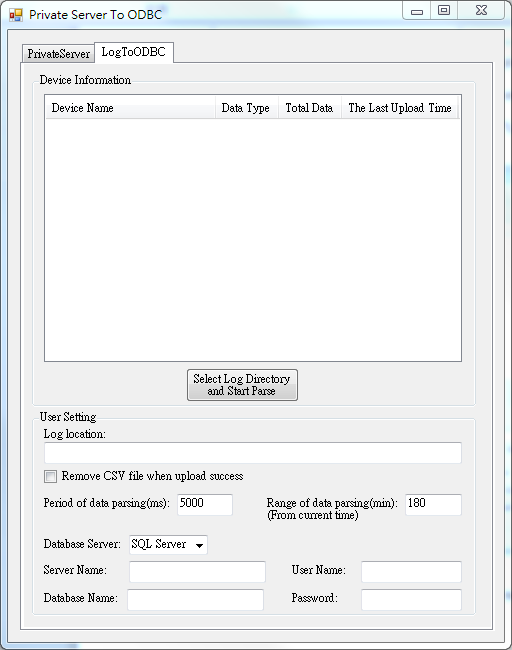
## Log To ODBC

When you finish installing software described in Chapter 2, please proceed following steps to configure sample Log To ODBC.

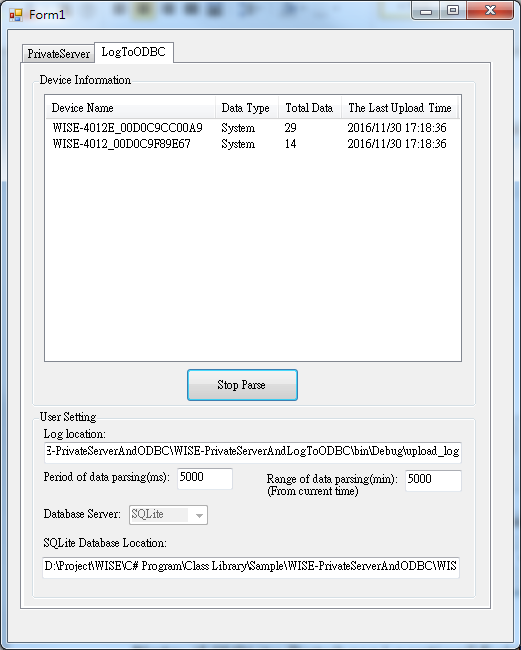
Several parameters and buttons of Log To ODBC sample program could be modified or execute, please refer to below table:

|  |  |
| --- | --- |
| Variable/Button | Description |
| Select Log Directory and Start Parse/Parse Stop | Start/Stop csv files parsing of log file. |
| Log location | Directory of “signal\_log” or “system\_log”. |
| Remove CSV file when insert into DB success | If enable, Log To ODBC will delete the CSV file when upload success. |
| Period of data parsing (ms) | Interval of scan csv files in milliseconds.  (0~999,999,999, default is 5,000ms). |
| Range of data parsing (min) (From current time) | Parsing csv data within N minutes from current time.  (0~999,999,999, default is 180mins) |
| Database Server | Select database which data value store in. |
| SQLite Database Location | Directory that SQLite database file store in. |
| Server Name | SQL server name. |
| Database Name | Database name of SQL server. |
| User Name | User name for login SQL server. |
| Password | Password for login SQL server. |

When Log To ODBC sample is opened, you have to finish user setting, in user setting, you can click the combo box “Database Server” to select database, if SQL server is used, please fill “Server Name”, “Database Name”, “User Name”, “Password” as below

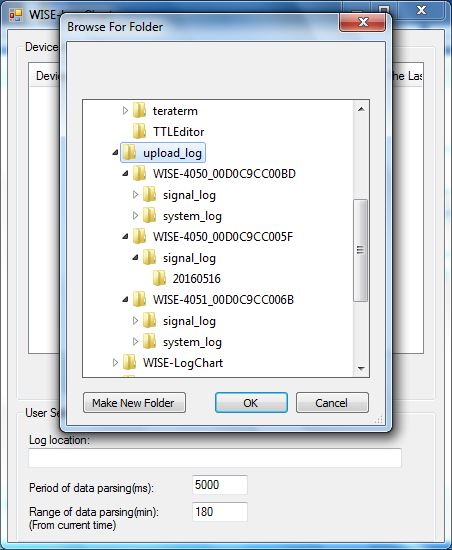


Otherwise, SQLite is selected, you only need to set db file location(SQLite Database Location).

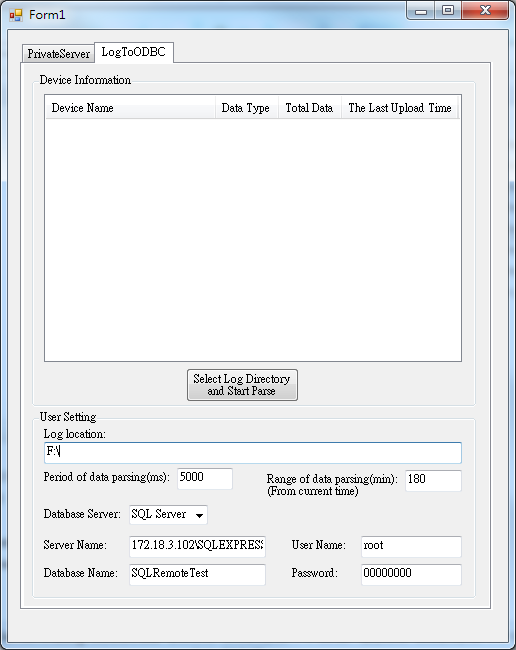


**Note:** if “SQLite Database Location” field is empty, Log To ODBC sample will store db file in the same folder that Log To ODBC located.

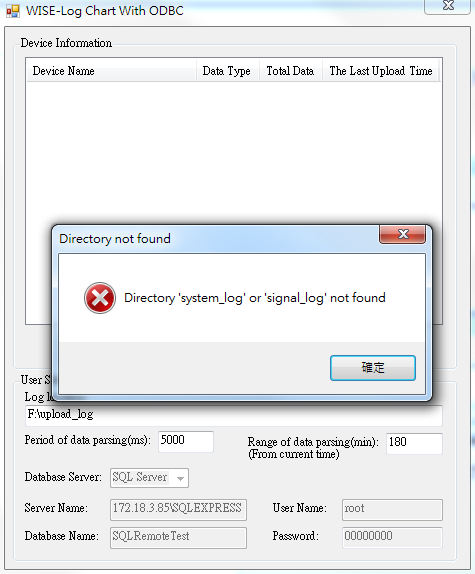
When user setting is finished, press “Select Log Directory and Start Parse” button to set the location of log folder “upload\_log”



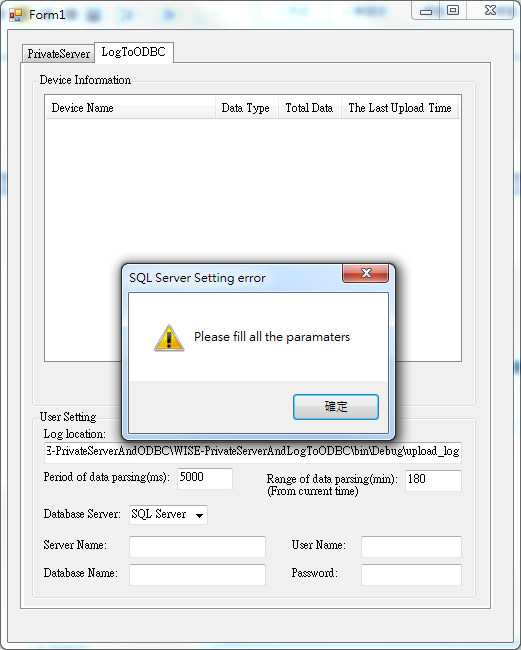
Otherwise, you can also paste string of log directory to “Log location” then press “Select Log Directory and Start Parse”.



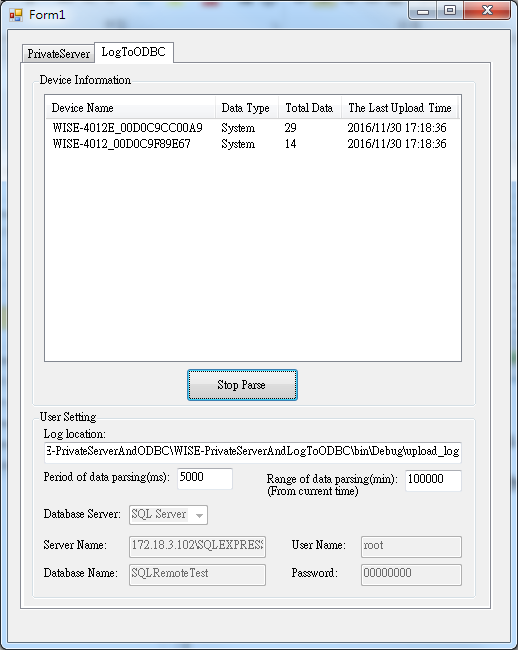
**Note:** If the directory does not have directory: “signal\_log” and “system\_log”, Log To ODBC sample will show error message box as below.



**Note:** If there is any empty value of user setting in SQL Server, Log To ODBC sample will show a setting error message as below.

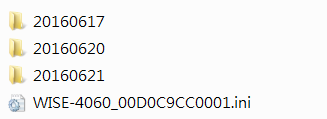


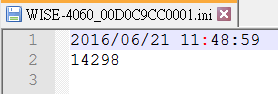
After setting, Log To ODBC will be automatically start scanning and parsing all the csv files of each log type in your log directory in period, you can see device name, data type, data counts and the latest upload time of csv file in Device Information section.



Log To ODBC will also generate an ini file that store data count and the latest upload time to avoid rescan the same data when next time you start Log To ODBC program.

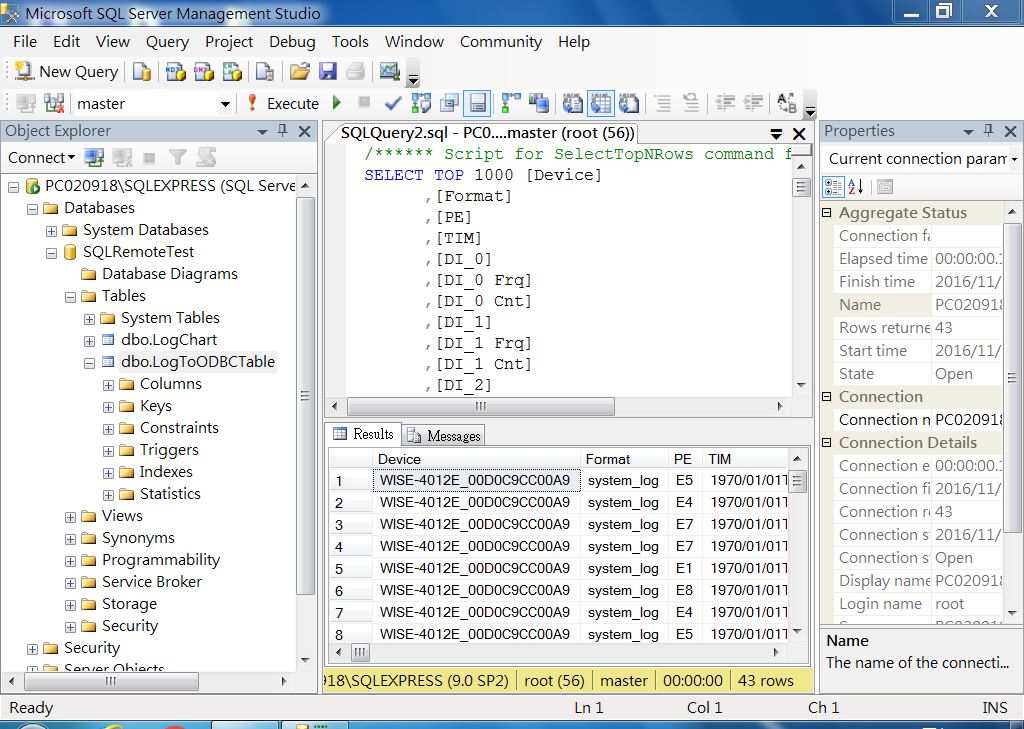
**Note:** If you wanted to rescan the old csv file that had been scanned, you have to delete ini file under “system\_log” and “signal\_log” directory, when CVS file insert into database success, Log To ODBC will remove those file if “Remove CSV file when insert into DB success” is enabled.



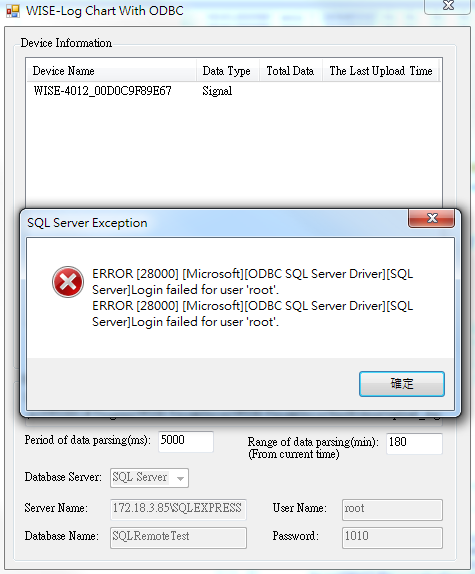


Log data will be stored in data table “LogToODBCTable” in SQL server or the database file “LogToODBCDemo.db3” that is generated in location “SQLite Database Location” of user setting (if you want to see the detail data of db file, please refer ch.4).



If SQL Server is used, you can see the data table “dbo.LogToODBC” in your database as below. 

Otherwise, if there is any SQL Server process issue, Log To ODBC sample will show error message on screen then stop data parse.

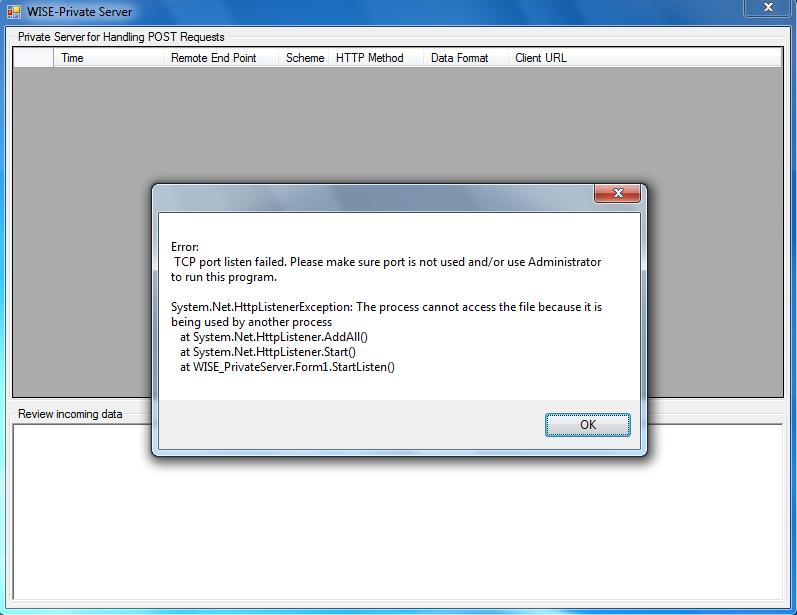


**Chapter** **4**

# Troubleshooting

## Private Server Start-up error

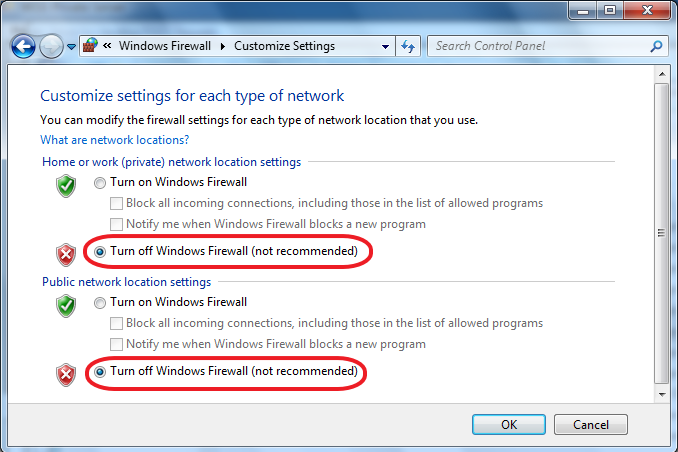
You may see below error message box when start-up private server:



Due to different settings of each computer, there are several possibilities that cause this error. Please check following methods:

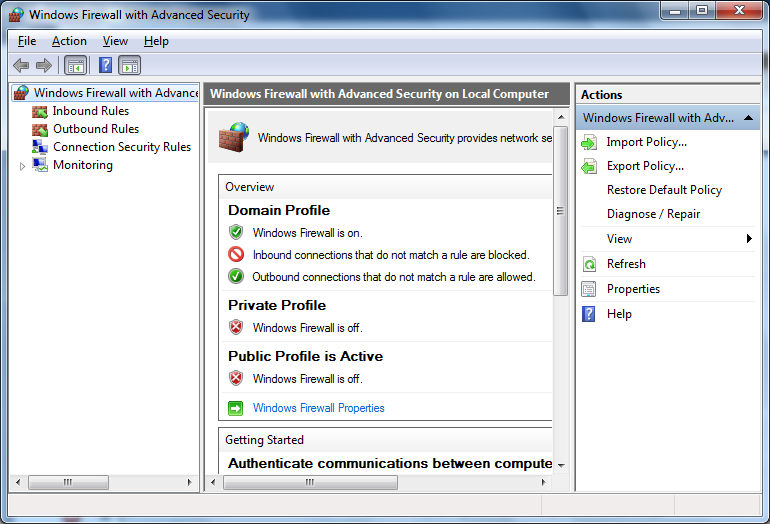
* Method1: The listen port of Private Server may be used by other program on your computer. Please close any program that uses port 8000/8080 or choose another port number (e.g. 8080) for your own program.
* Method2: Use administrator privilege to open Microsoft Visual Studio 2008 and execute Private Server sample program. This could be achieved by clicking mouse right button and choose “Run as administrator”.
* Method3: The listen port of Private Server may be blocked by Windows Firewall. Please choose one of following methods to un-block Windows Firewall.
  + Method 1: Disable Windows Firewall:

In Windows Startup Menu, choose 「Control Panel」🡺 「System and Security」🡺「Windows Firewall」. Select “Turn Windows Firewall on or off” in left menu and choose “Turn off Windows Firewall” in Home or work network location settings and Public network location settings.

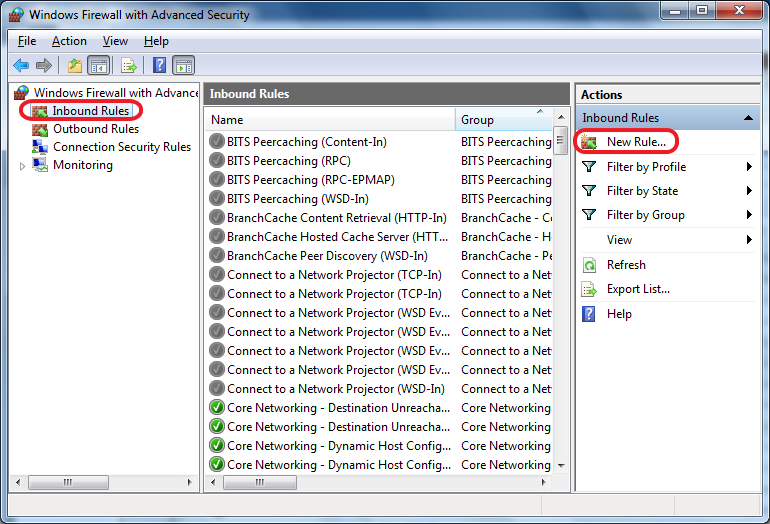


* + Method 2: Add Windows Firewall Rule:

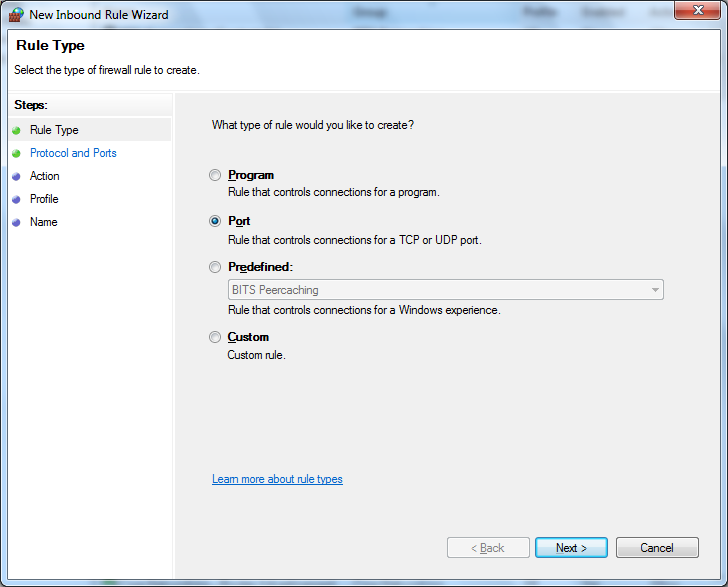
1. In Windows Startup Menu, choose 「Control Panel」🡺 「System and Security」🡺「Windows Firewall」. Select “Advanced settings” in left menu and you will see below figure:



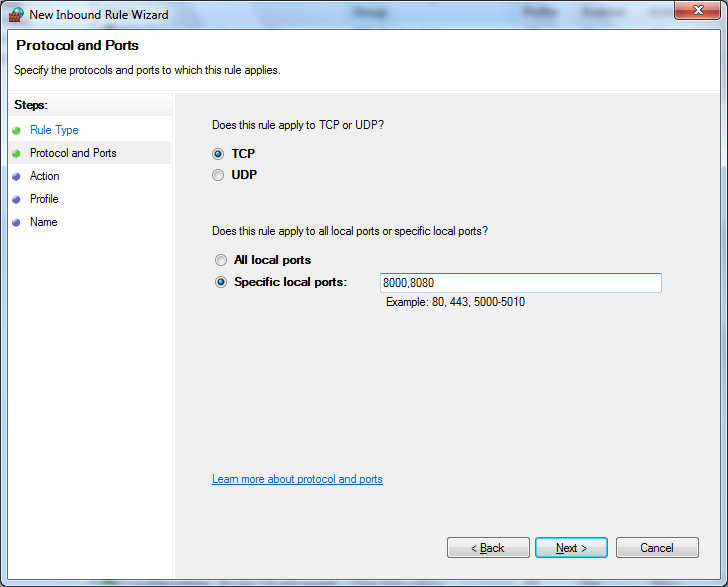
1. Please choose Inbound Rules and click “New Rule” in right menu.



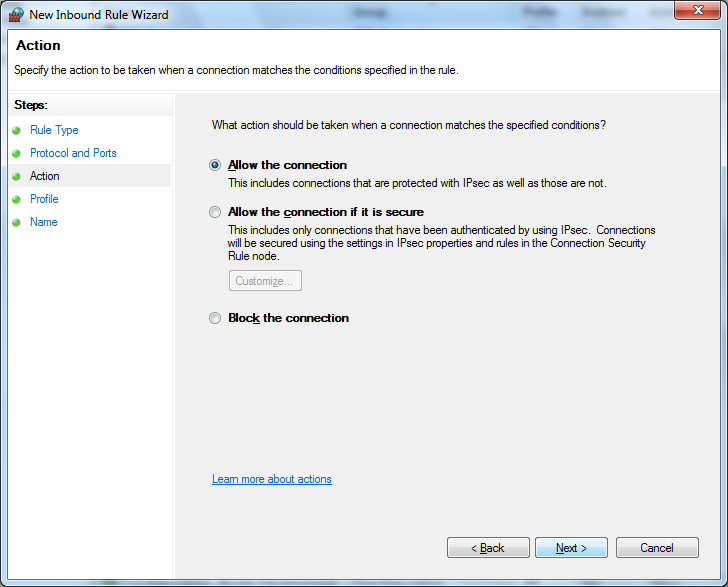
1. In New Inbound Rule Wizard, choose Port and press “Next”.



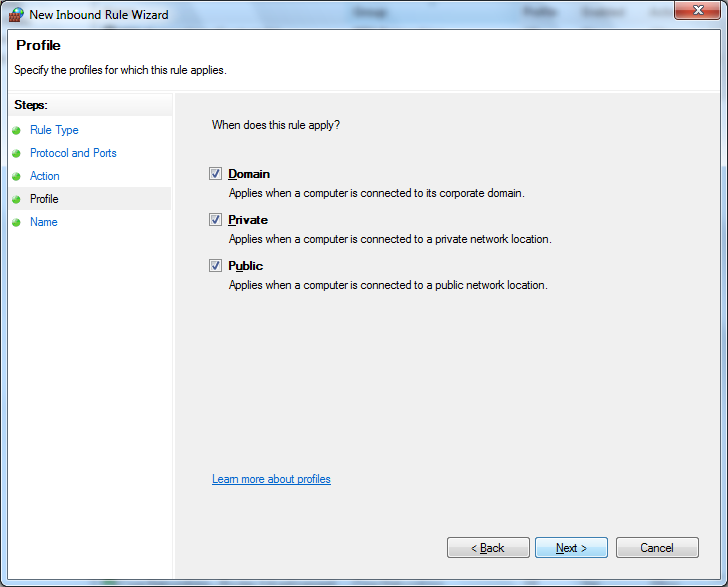
1. Choose TCP and add Specific Local ports: 8000, 8080 for Private Server.



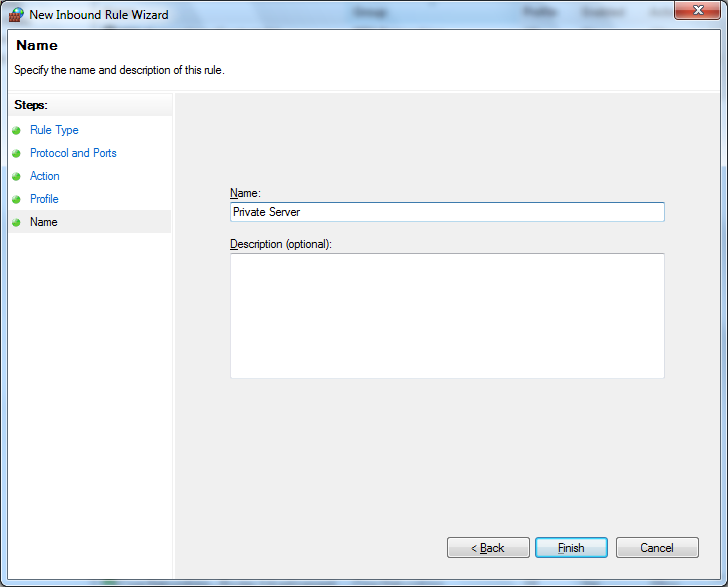
1. Choose Allow the connection.



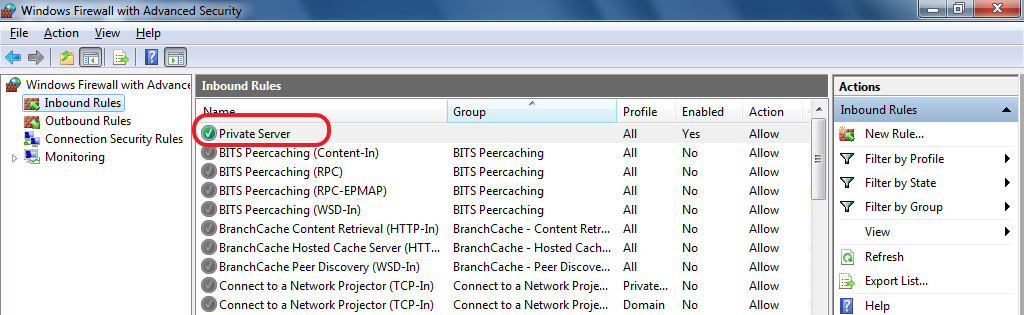
1. Press next.



1. Name this firewall rule as Private Server and press “Finish”.

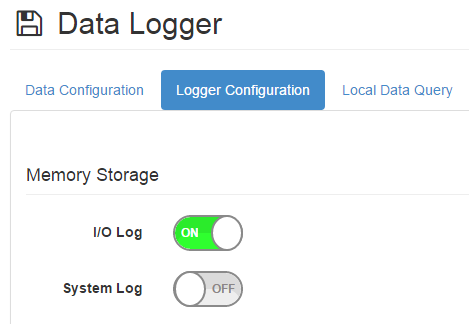


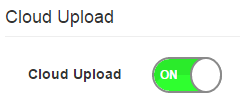
1. You will see added rule in Inbound Rules. This rule will allow Private Server to receive incoming WISE logs.



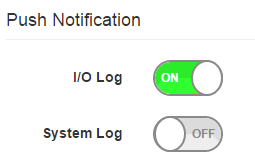
## No log received in Private Server

Step 1: If you want to receive upload csv file, please make sure IO log or System Log in WISE is turned on in Memory Storage of Data Logger configuration. Then turn on Cloud Upload in Data Logger configuration.





Step 2: If you want to receive Push Notification of IO Log or System Log, please make sure the corresponding items are turned on in Data Logger configuration.

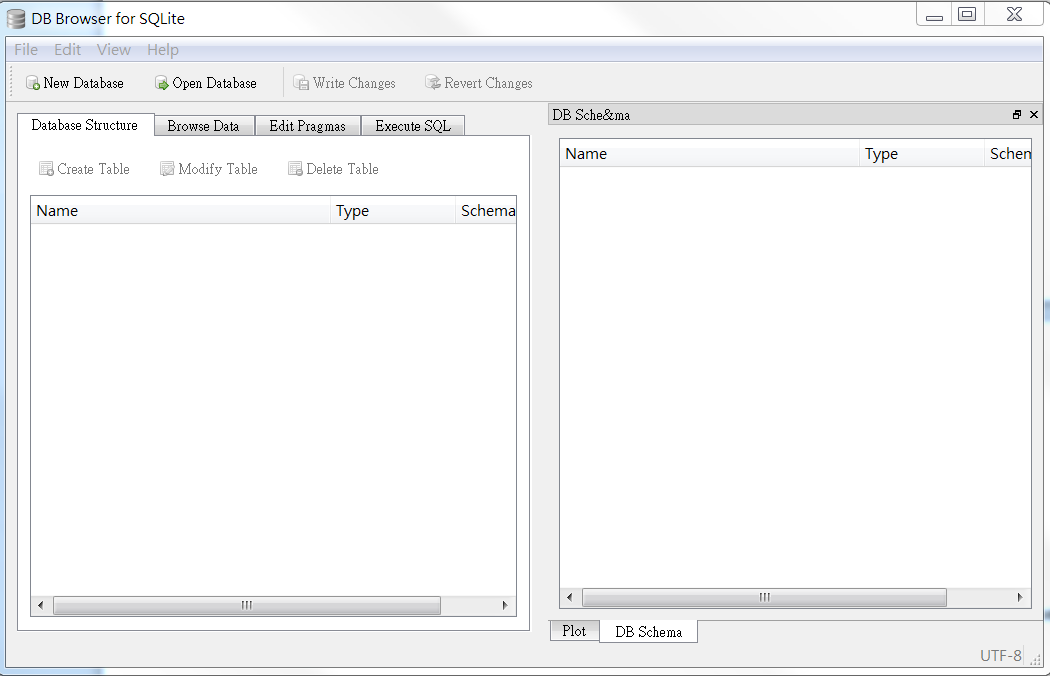


# Database generation and viewer

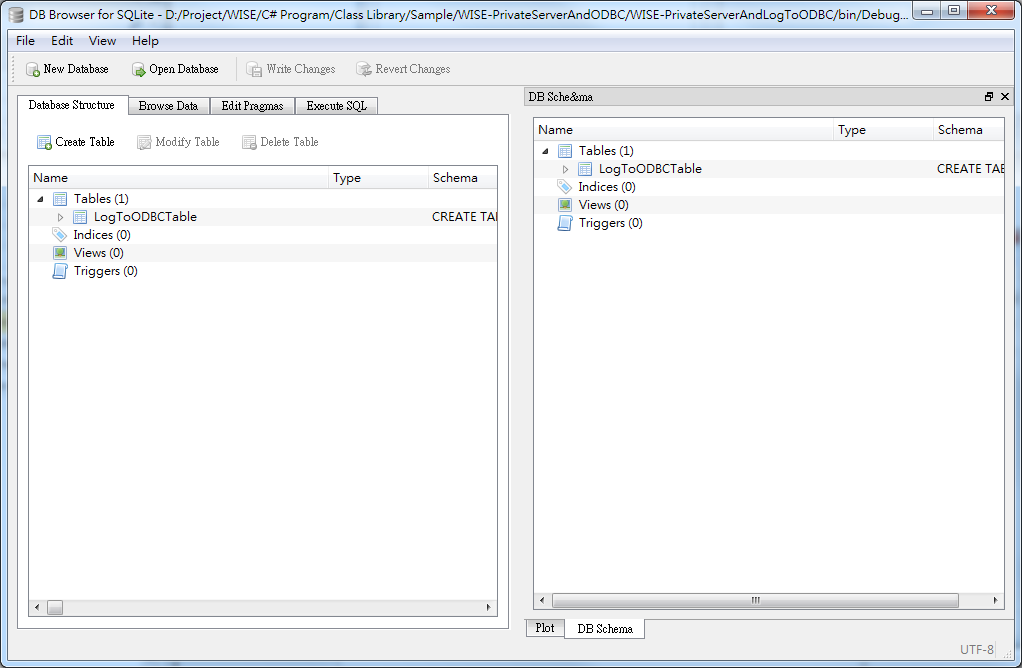
For creating data columns, we defined two variables “TitleArray” and “Dataformat” to generate column name and column format in Log To ODBC sample code, in this sample, we defined eight groups of each AI, DI and DO, sixteen bit and word data of each Modbus COM, you can refer appendix to know the details of column name.

If you choose to use SQLite but do not specify SQLite file location,, when Log To ODBC starts parsing, the database file will be auto-generated, you can use SQLite tools such as “DB Browser for SQLite” to see data table.

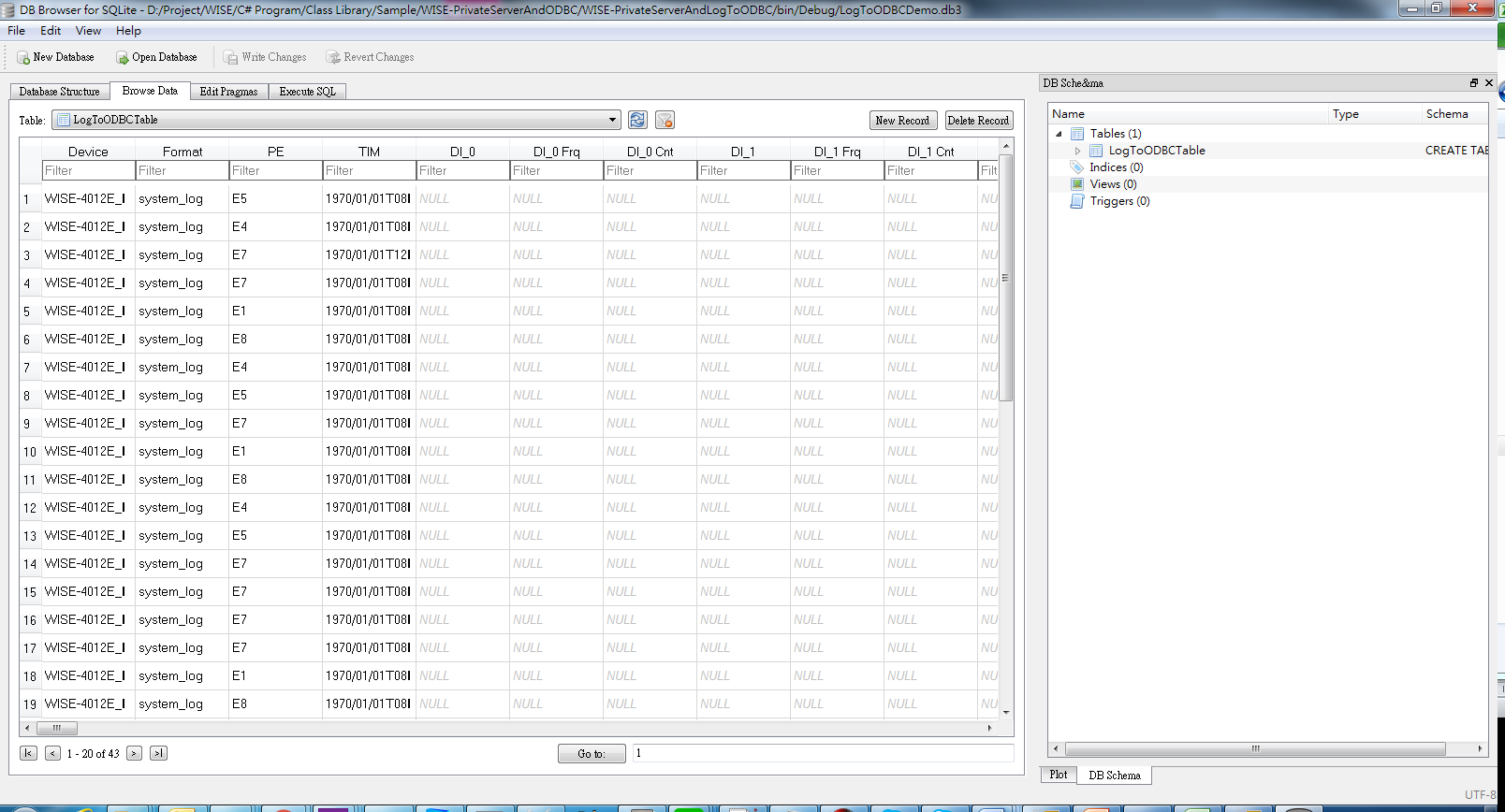
**Note:** You can download DB Browser for SQLite installer at <http://sqlitebrowser.org/>



When you open the program, click “Open Database” to load the db file.

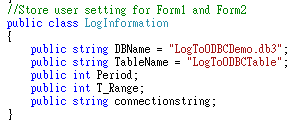


When db file opened, click “Browse Data”, then you can see the detail information of I/O and system data.



If you want to use SQL command to filter data, you can click “Execute SQL” then input some command that SQLite has support.

**Note:** If you want to change the file or table name of database, you can modify the code (in the end of LogToODBC.cs, DBName means SQLite file name, TableName means table name), otherwise, if you want to add new data columns (such as user define tag) in database, you can refer “Appendix C”.



**Appendix**

# Appendix A: I/O data header in CSV format

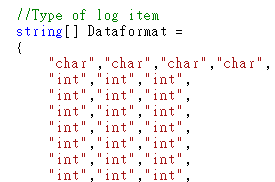
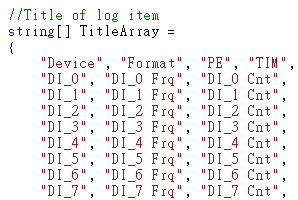
|  |  |  |
| --- | --- | --- |
| I/O data header in CSV format | | |
| Category | Type | Header |
| Periodic/Event | Periodic/Event | PE |
| Timestamp | Timestamp | TIM |
| DI | DI status | DI\_x |
| DI counter | DI\_x Cnt |
| DI frequency | DI\_x Frq |
| DO | DO status | DO\_x |
| Absolute pulse | DO\_x Ps |
| Incremental pulse | DO\_x PsIV |
| AI | AI raw value | (SLx) AI\_x Val |
| AI Max value | (SLx) AI\_x HVal |
| AI Min value | (SLx) AI\_x LVal |
| AI value after scaling | (SLx) AI\_x SVal |
| AI engineering value | (SLx) AI\_x Eg |
| AI Max engineering value | (SLx) AI\_x HEg |
| AI Min engineering value | (SLx) AI\_x LEg |
| AI status | (SLx) AI\_x Evt |
| AI physical value | (SLx) AI\_x PEg |
| AI engineering value in floating type | (SLx) AI\_x EgF |
| AI Max engineering value in floating type | (SLx) AI\_x HEgF |
| AI Min engineering value in floating type | (SLx) AI\_x LEgF |
| AI physical value in floating type | (SLx) AI\_x PEgF |
| Modbus/RTU | Bit data | COM\_x Bit\_x |
| Bit error code | COM\_x Bit\_x Evt |
| Word data | COM\_x Wd\_x |
| Word error code | COM\_x Wd\_x Evt |

# Appendix B: System data header in CSV format

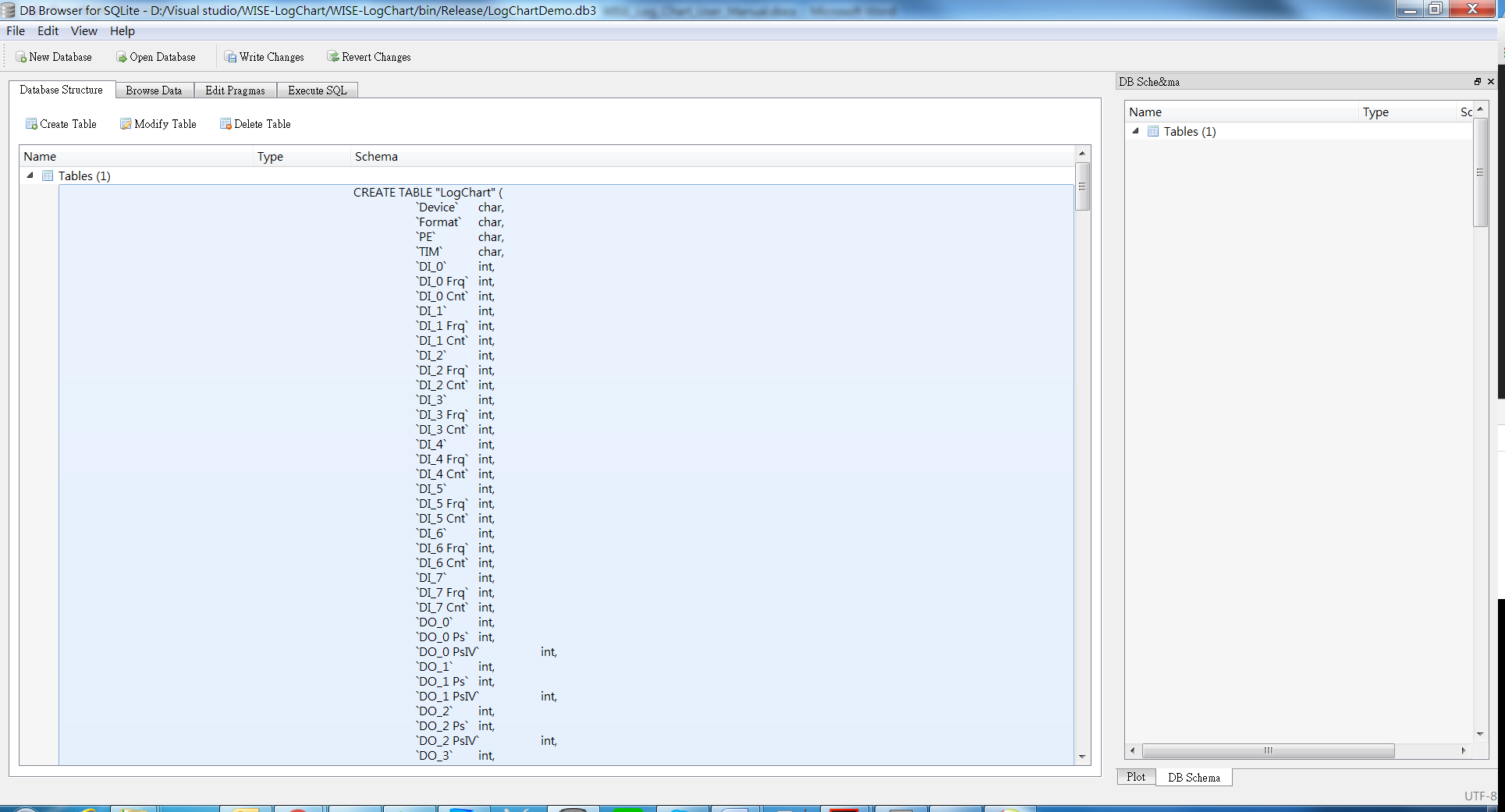
|  |  |  |
| --- | --- | --- |
| System data header in CSV format | | |
| Category | Type | Header |
| Periodic/Event | Periodic/Event | PE |
| Timestamp | Timestamp | TIM |
| Event Data | Event Data | Record |

# Appendix C: Add new data columns in database

If you wanted to add more columns in csv, you can modify variables in LogToODBC.cs of Log To ODBC sample, add tag title into “TitleArray”, add tag format into “Dataformat” (For example: your data is integer, tag format is “int”, if your data is string, tag format is “char”), you also have to add title and format in db file.



You also have to add data title and format into db file, after opening db file by DB Browser for SQLite, click content of Tables, then click “Modify Table”.



You will see the window “Edit table definition” as below, click “Add field” to create new field, change field name to your new tag title and field type to new tag format, then press “OK” to update db table.

