**User Manual**

**Log Chart With ODBC**

**for WISE-4000 Series**

CONTENTS

[1. Introduction 4](#_Toc464733192)

[1.1. About This Manual 4](#_Toc464733193)

[1.2. Organization of This Manual 4](#_Toc464733194)

[2. Installations 7](#_Toc464733195)

[3. Configuration of Log Chart 10](#_Toc464733196)

[4. Database generation and viewer 19](#_Toc464733197)

[Appendix A: I/O data header in CSV format 22](#_Toc464733198)

[Appendix B: System data header in CSV format 23](#_Toc464733199)

[Appendix C: Add new data columns in database 24](#_Toc464733200)

**Chapter** **1**

# Introduction

## About This Manual

This document describes the installation and usage of Log Chart sample code for Advantech WISE-4000 series IoT Wireless I/O Module products.

For WISE-4000 series, system log and signal log could be sent to a log server such as Private Server or DropBox. Signal log include I/O value, I/O error code and other information, system log include system error code and other information. These logs are created in csv or json format (Data in csv format is all the same as json format). For parsing csv data, we create Log Chart 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 Log Chart program by using Microsoft Visual Studio 2008 and a database viewer software (DB Browser for SQLite) which shows the detail value in SQLite database file. For data browser in Microsoft SQL server, please refer to Microsoft SQL server web site.

**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** 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 Log Chart.

**Configuration of Log Chart** This section gives the user a walk-through in configuring Log Chart.

**Database viewer**

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

**Chapter** **2**

# Installations

Before running Log Chart 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/>

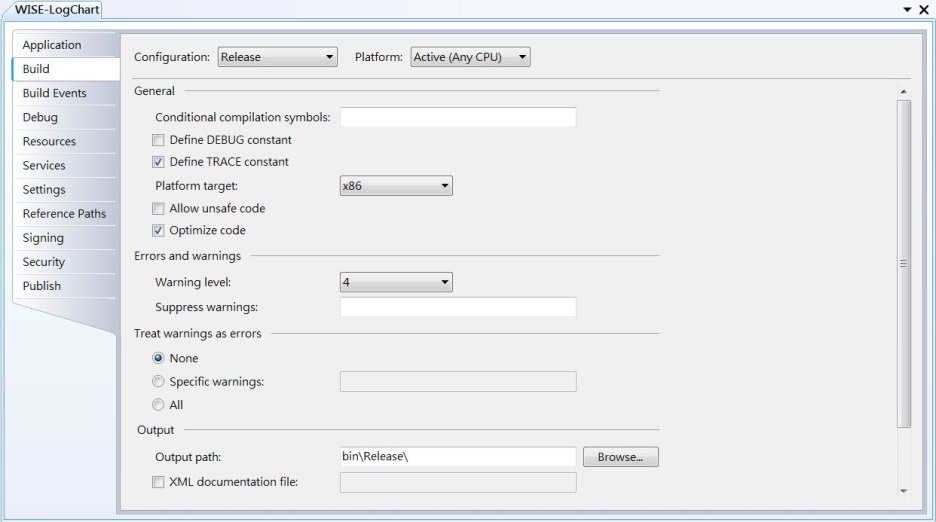
**Note:** Because of build code environment, you have to install SQLite ODBC driver “**sqliteodbc.exe**”, not “**sqliteodbc\_w64.exe**”.

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

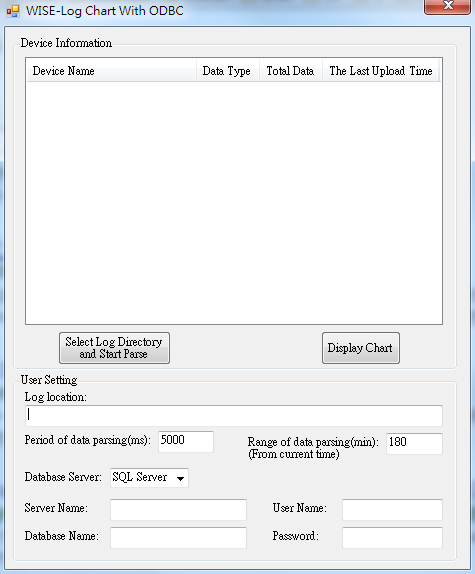
<https://blogs.msdn.microsoft.com/walzenbach/2010/04/14/how-to-enable-remote-connections-in-sql-server-2008/>

Finally, execute Microsoft Visual Studio 2008 or above version on your PC. Please follow below steps to compile Log Chart 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 Log Chart program is located.
3. On the menu bar, choose Project🡺WISE-Log Chart 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.



**Chapter** **3**

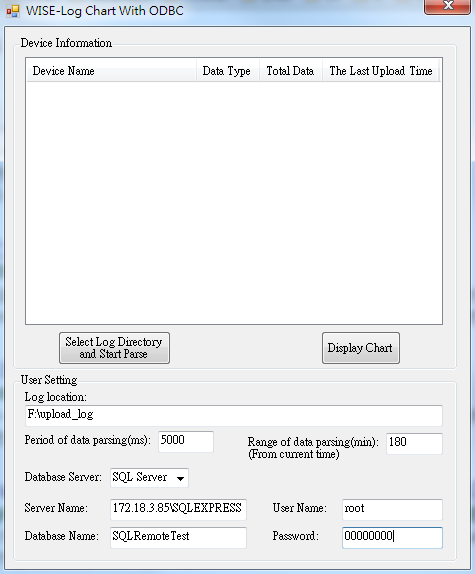
# Configuration of Log Chart

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

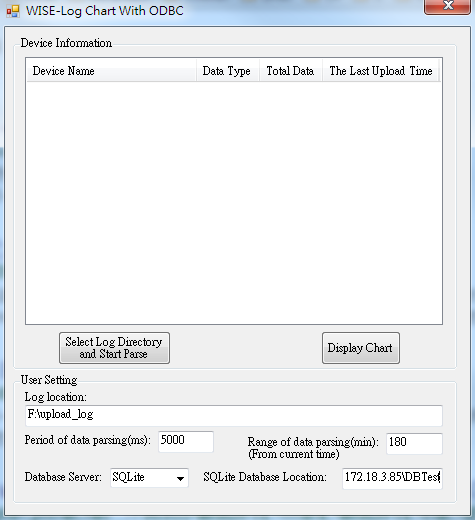
Several parameters and buttons of Log Chart sample program could be modified or execute, please refer to below table:

|  |  |
| --- | --- |
| Variable/Button | Description |
| Select Log Directory and Start Parse/Parse Stop | Start/Stop csv files parsing of log file. |
| Display Chart | Display chart of the log type that is selected. |
| Log location | Directory of “signal\_log” or “system\_log”. |
| 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 Chart 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

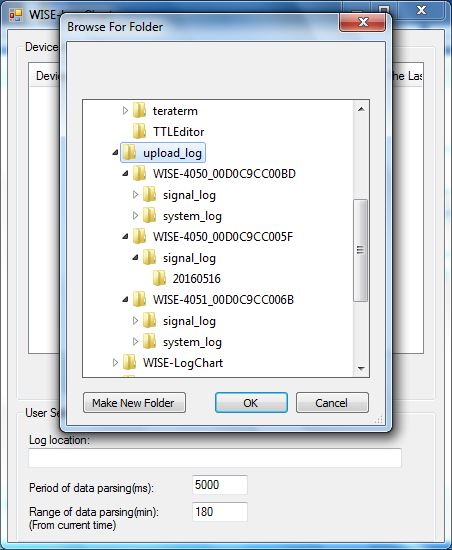


Otherwise, SQLite is selected, you only need to set db file location(SQLite Database Location).

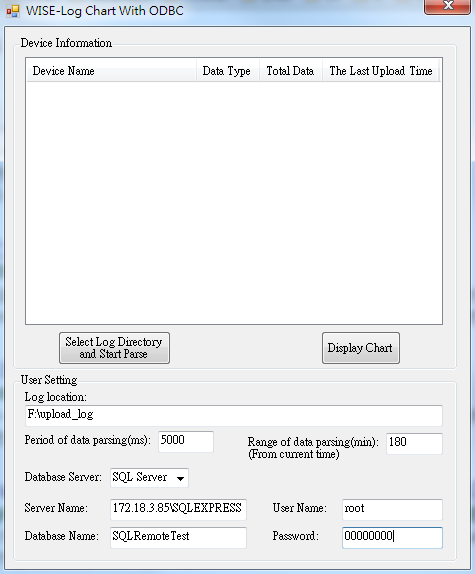


**Note:** if “SQLite Database Location” field is empty, Log Chart sample will store db file in the same folder that Log Chart located.

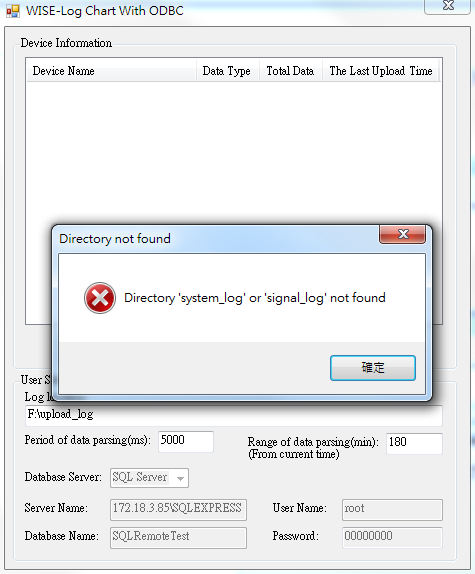
When user setting is finished, press “Select Log Directory and Start Parse” button to set the location of log folder “upload\_log”



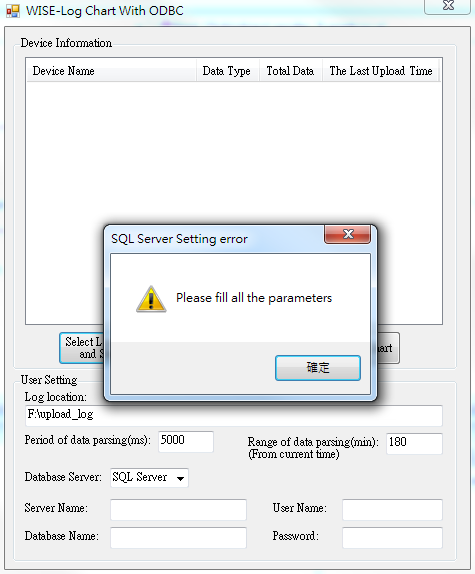
Otherwise, you can also paste string of log directory to “Log location” then press “Select Log Directory and Start Parse”.



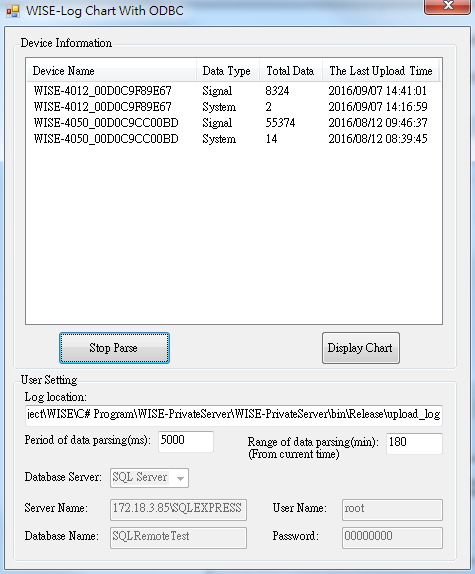
**Note:** If the directory does not have directory: “signal\_log” and “system\_log”, Log Chart sample will show error message box as below.



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

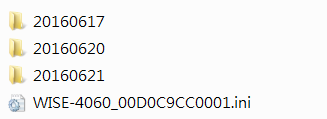


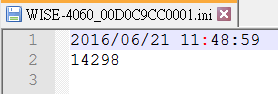
After setting, Log Chart will be automatically start scanning and parsing all the csv files of each log type in your log directory in period, you can see device name, data type, data counts and the latest upload time of csv file in Device Information section.



Log Chart 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 Chart 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.

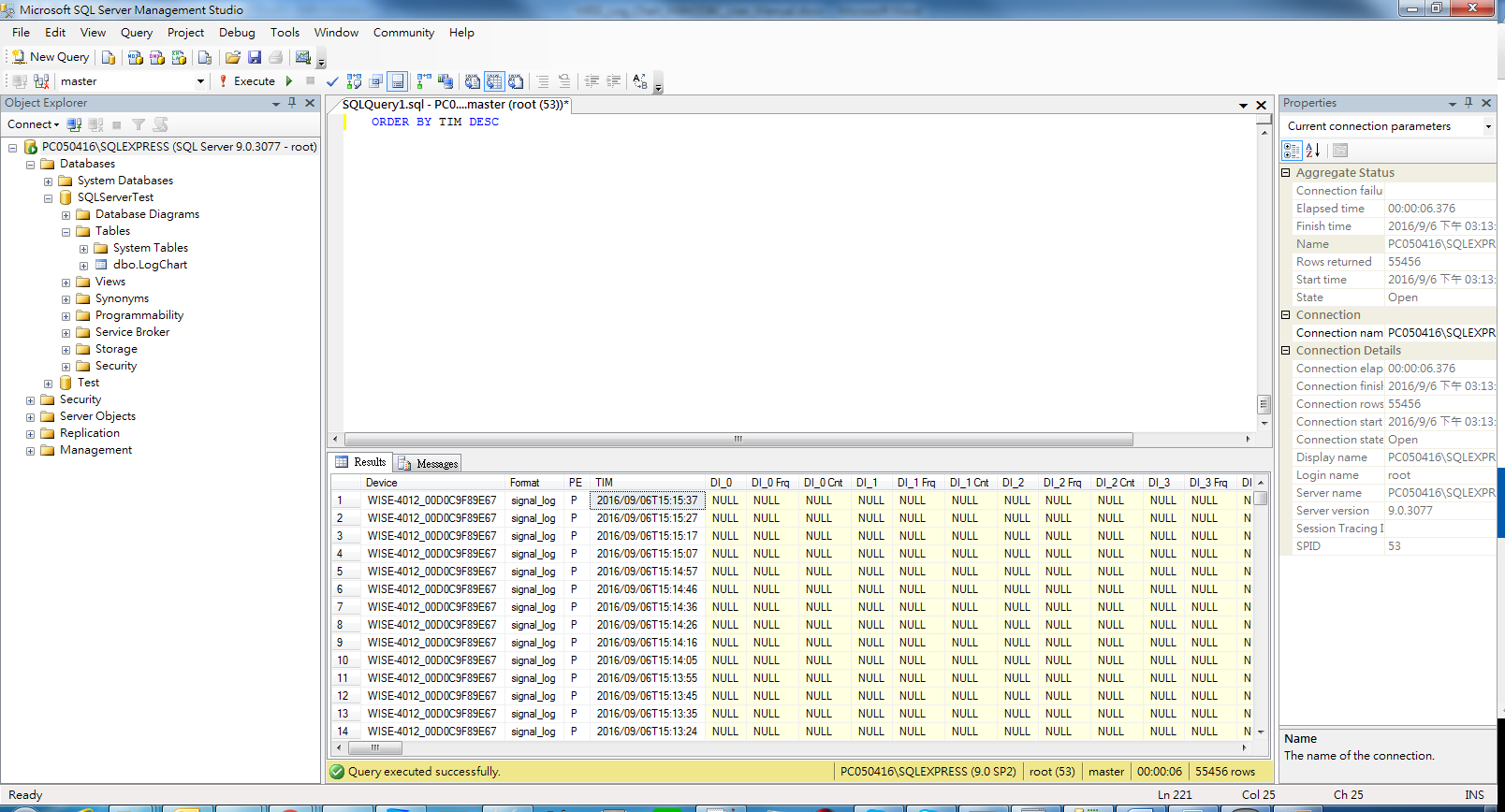




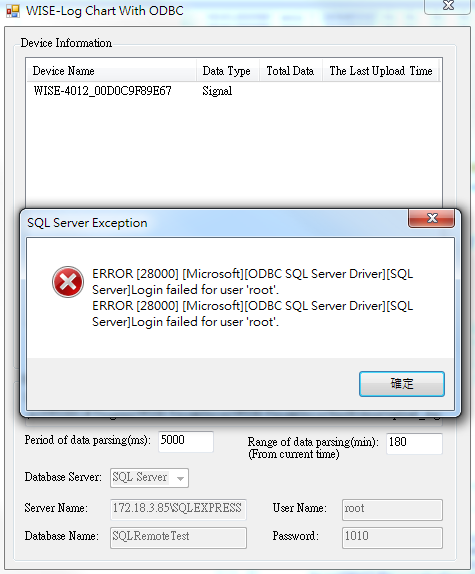
Log data will be stored in data table “LogChart” in SQL server or the database file “LogChartDemo.db3” that is generated in location “SQLite Database Location” of user setting (if you want to see the detail data of db file, please refer ch.4).



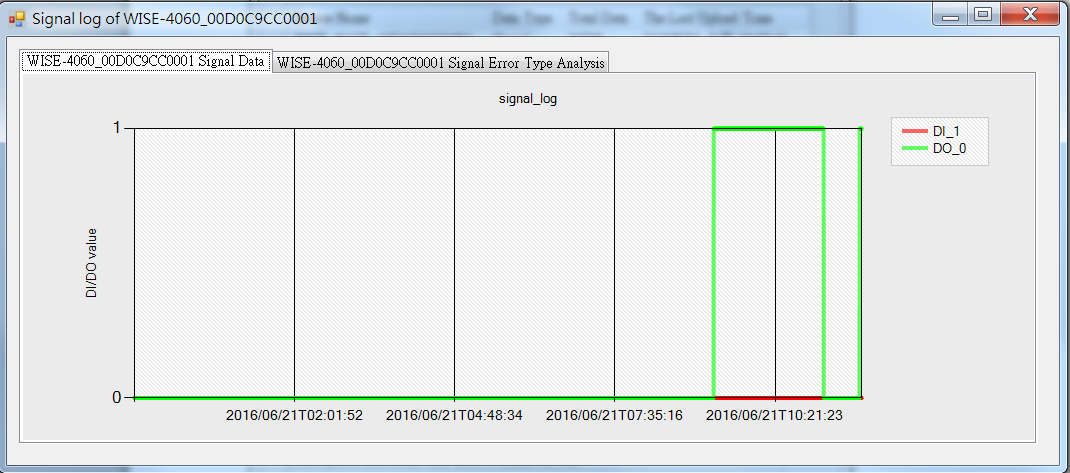
If SQL Server is used, you can see the data table “dbo.LogChart” in your database as below.

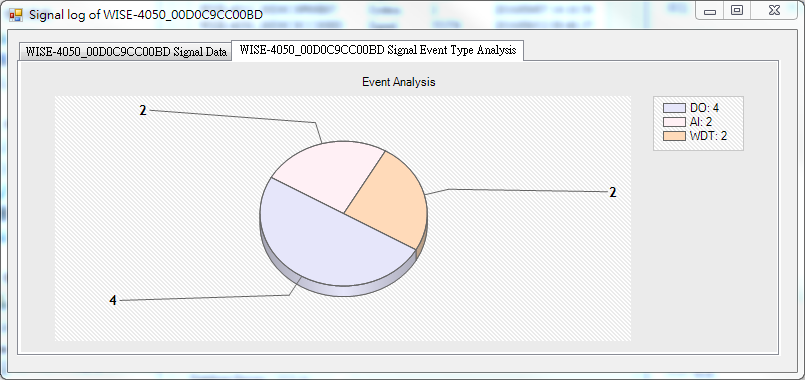


Otherwise, if there is any SQL Server process issue, Log Chart sample will show error message on screen then stop data parse.

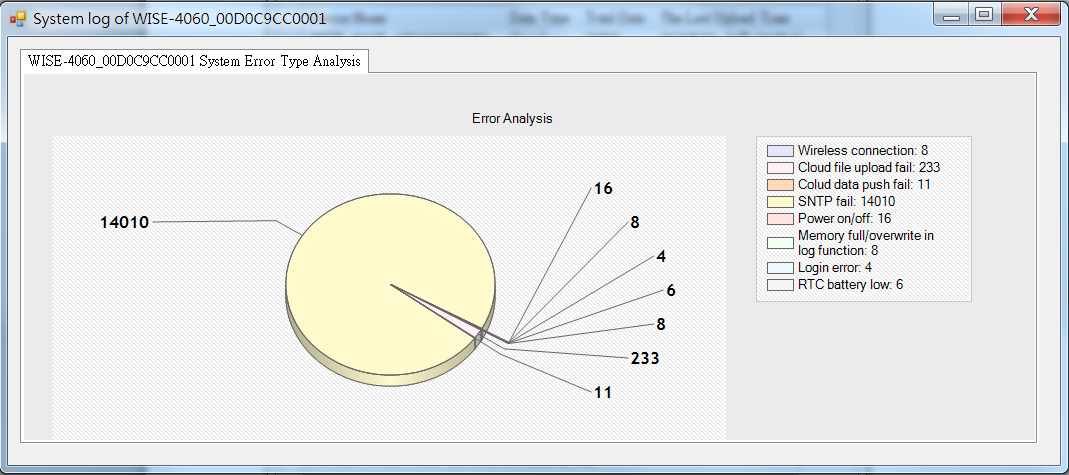


Select a device and press button “Display Chart”, the new real time chart windows will show on screen according to data type, if data type is signal log, chart window will divide into signal I/O value (ex: DI\_1, DO\_0, AI\_0 scaling value) and event type analysis, you can click the tab to switch to another chart.

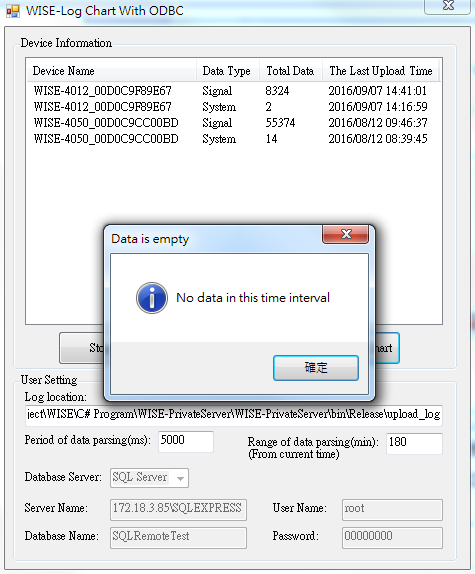
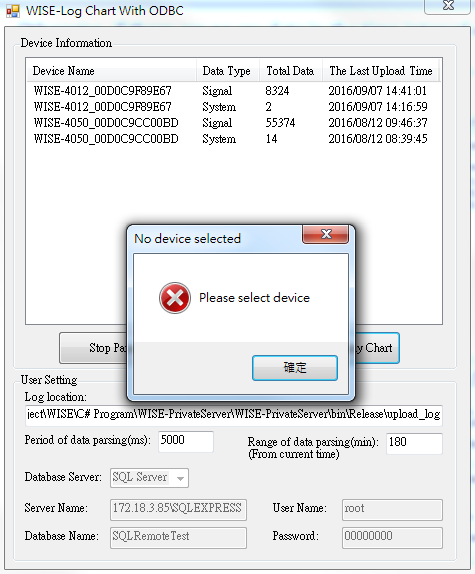




If data type is system, chart window will only show the system error type analysis on screen because there are no I/O data in system log.



Otherwise, if there are no csv data in the time interval or no device selected when you press “Display Chart”, only message box will show on screen.

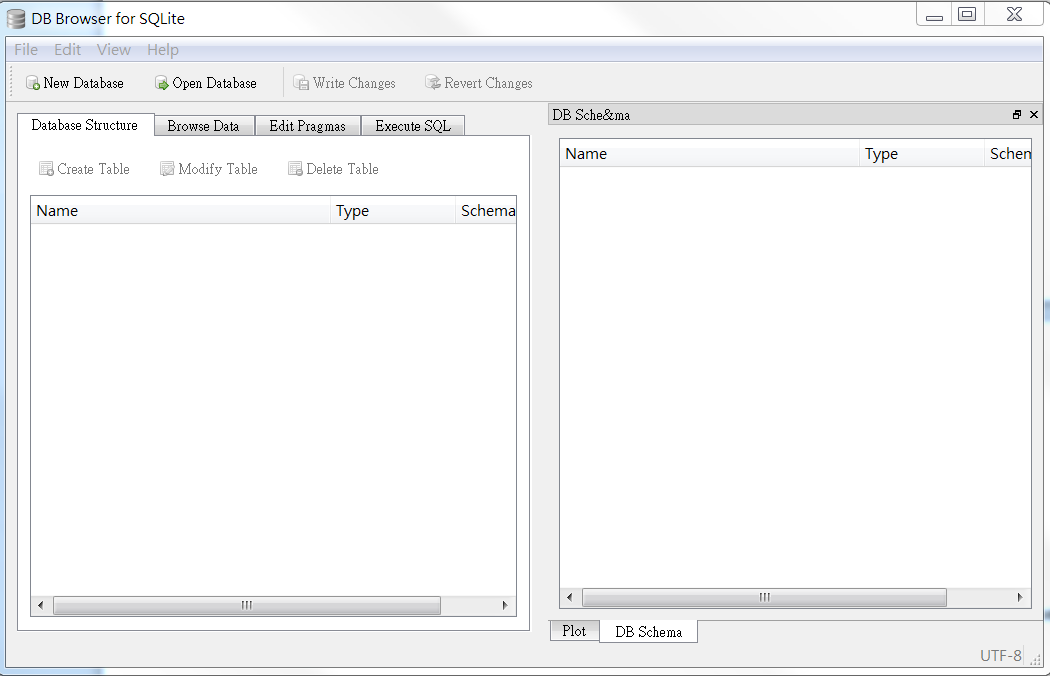
**Chapter** **4**

# Database generation and viewer

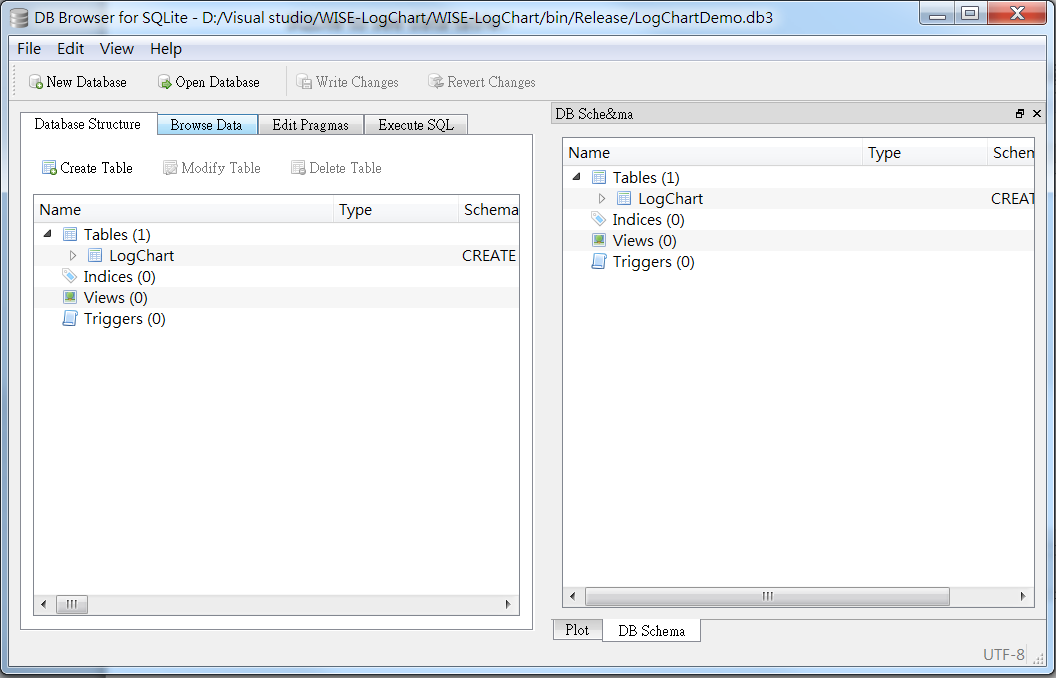
For creating data columns, we defined two variables “TitleArray” and “Dataformat” to generate column name and column format in Log Chart sample code, in this sample, we defined eight groups of each AI, DI and DO, sixteen bit and word data of each Modbus COM, you can refer appendix to know the details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

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

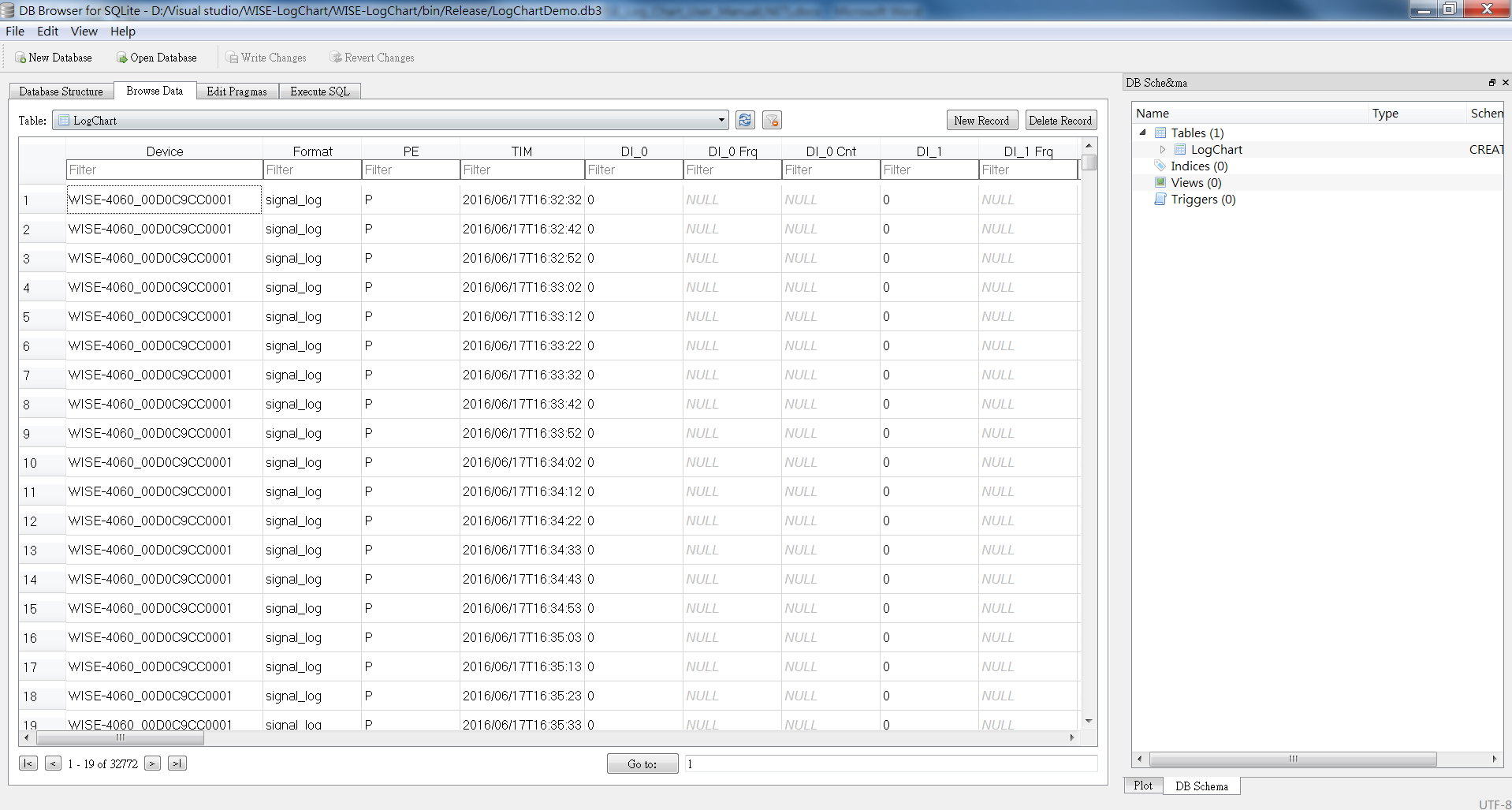
**Note:** You can download DB Browser for SQLite installer at <http://sqlitebrowser.org/>



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

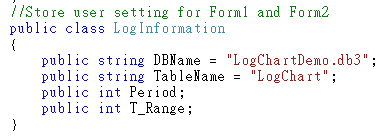


When db file opened, click “Browse Data”, then you can see the detail information of I/O and system data.



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 Form1.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

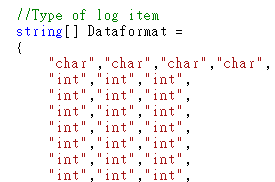
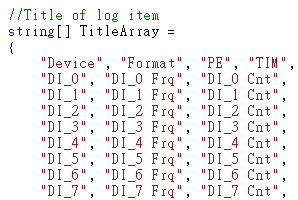
|  |  |  |
| --- | --- | --- |
| I/O data header in CSV format | | |
| Category | Type | Header |
| Periodic/Event | Periodic/Event | PE |
| Timestamp | Timestamp | TIM |
| DI | DI status | DI\_x |
| DI counter | DI\_x Cnt |
| DI frequency | DI\_x Frq |
| DO | DO status | DO\_x |
| Absolute pulse | DO\_x 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

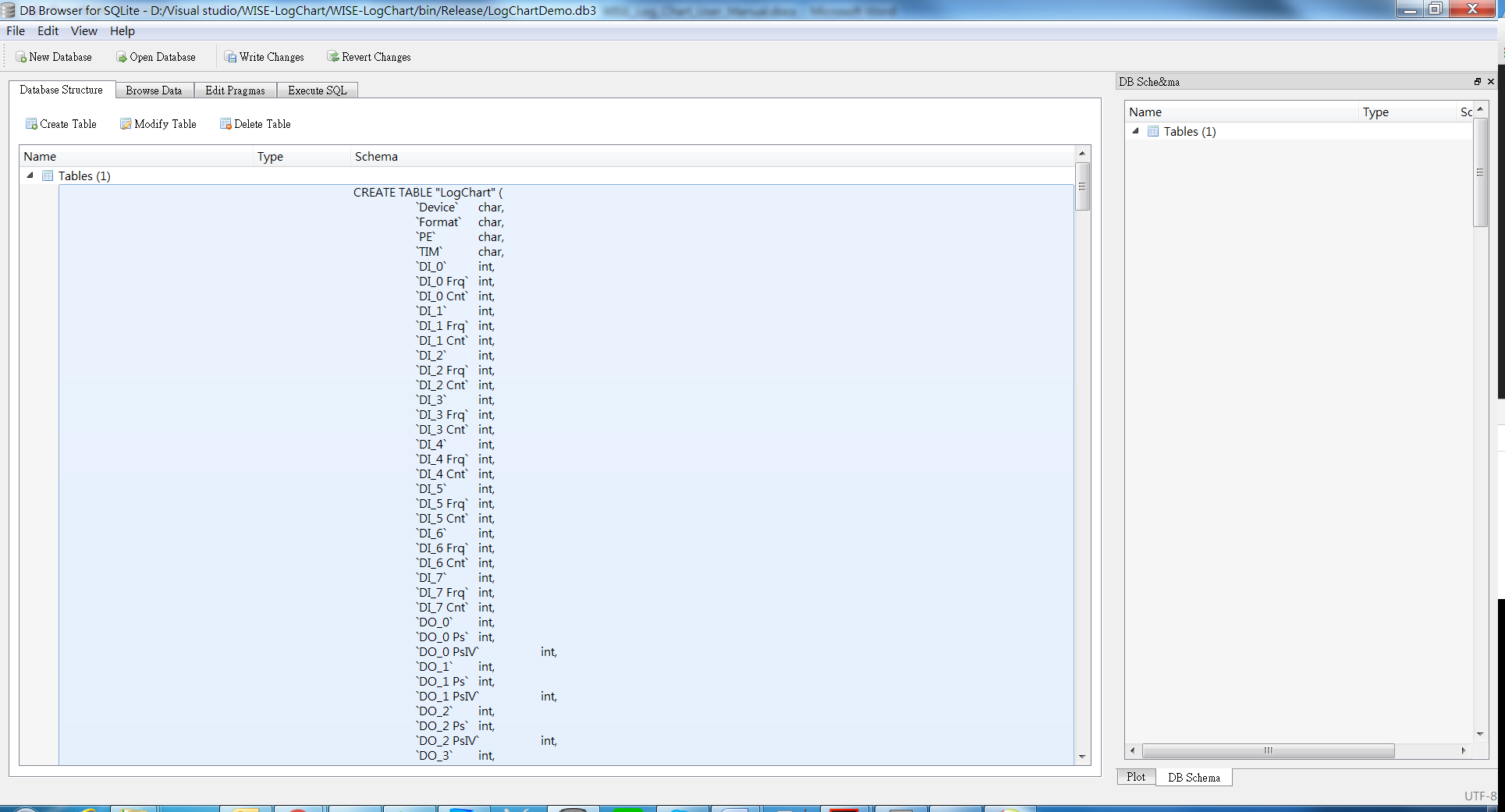
|  |  |  |
| --- | --- | --- |
| System data header in CSV format | | |
| Category | Type | Header |
| Periodic/Event | Periodic/Event | PE |
| Timestamp | Timestamp | TIM |
| Event Data | Event Data | Record |

# Appendix C: Add new data columns in database

If you wanted to add more columns in csv, you can modify variables in Form1.cs of Log Chart 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.

