



User Manual

Version 1.0.2 June 2019

NB-DA Server

IoT Service Engine

The screenshot displays the NB-DA Server configuration window. It includes sections for System Config, Database Config, and MQTT Config, along with a table showing the status of various stations.

System Config

- Total Stations: 3
- Buttons: Add Station, Delete Station, Start Station, Stop Station, Start All Stations, Stop All Stations
- Station ID: 0
- UDP Server Port: 5384
- Modbus Server Port: 502
- Session alive time (s): 120
- Save Log Info:

Database Config

- Enable:
- SQL Type: mysql
- SQL IP: 127.0.0.1
- SQL Data Base: gnp-540m-nb
- SQL User Name: rntnot
- SQL Password: *****
- SQL Size Alarm (MB): 0
- SQL Size Limit (MB): 0

MQTT Config

- Enable:
- MQTT Broker: iot.eclipse.org
- MQTT Port: 1883
- MQTT Subscribe: .cloud.ICPDAS.USEI
- MQTT Publish: .cloud.ICPDAS.SERV
- MQTT User Name:
- MQTT Password:

Station	UDP Port	Modbus Port	MQTT	Database	Status	Sessions	Log View
0	5394	502	Disable	Enable	Online	7	Open
2	5396	504	Disable	Enable	Online	0	Open
99	5493	601	Enable	Enable	Online	0	Open

Table of Contents

1. Introduction.....	4
1.1 Features.....	4
1.2 RUN-TIME LICENSE.....	4
2. Hardware Requirement.....	5
2.1 Software architecture	5
2.2 Application architecture	6
2.3 System requirement	7
2.4 Product Support.....	7
3. NB-DA Server operation	8
3.1 Main menu	8
3.2 Manage Stations	12
3.2.1 Add Station	12
3.2.2 Update Station	13
3.2.3 Delete Station	14
3.3 Get Data from NB-DA Server	15
3.3.1 Database	15
3.3.2 MQTT Client.....	16
3.3.3 Modbus TCP Server	17
3.3.4 RAM Disk (option)	18
3.4 Control Remote Device I/O	19
3.4.1 Modbus TCP Command.....	19
3.4.2 Publish MQTT Message	19
Appendix A. Revision History	20

Important Information

Warranty

All products manufactured by ICP DAS are under warranty regarding defective materials for a period of one year, beginning from the date of delivery to the original purchaser.

Warning

ICP DAS assumes no liability for any damage resulting from the use of this product. ICP DAS reserves the right to change this manual at any time without notice. The information furnished by ICP DAS is believed to be accurate and reliable. However, no responsibility is assumed by ICP DAS for its use, not for any infringements of patents or other rights of third parties resulting from its use.

Copyright

Copyright © 2019 by ICP DAS Co., Ltd. All rights are reserved.

Trademark

Names are used for identification purpose only and may be registered trademarks of their respective companies.

Contact us

If you encounter any problems while operating this device, feel free to contact us via mail at: service@icpdas.com. We guarantee to respond within 2 working days.

1. Introduction

The NB-DA Server provided by ICP DAS is an IoT (Internet of Things) management software that has a strong core technology for handling data and lets the user save the trouble of dealing with large IO data. The NB-DA Server support RTU-540P-NB, GRP-540M-NB, etc., in ICP DAS that allow users to manage these devices remotely. It can monitor the local I/O data, local GPS data, and I/O data of Modbus devices. The NB-DA Server also support different protocol like UDP, Modbus and MQTT. Users can establish the remote system with MQTT broker or other SCADA software, even you could monitor and control the remote devices with Web form your mobile devices. That provides a multi-way to complete user's project.

1.1 Features

- ◆ Supports multiple stations. Every station can accept 2000 sessions to connect
- ◆ Can check live sessions and log messages in every station
- ◆ Supports multiple communication ports. Up to 2000 connections for one port
- ◆ Supports 6 data types (DI/DO/AI/AO/GPS/DEVINFO)
- ◆ Built-in Modbus TCP server
- ◆ Supports forwarding data to MQTT broker
- ◆ Supports MySQL/MariaDB database.
- ◆ Can log data automatically
- ◆ Supports UDP packet transmission
- ◆ Can store I/O data (.csv file) in RAM Disk

1.2 RUN-TIME LICENSE

RUN-TIME LICENSE	
Free Version	Up to 10 sessions can be managed in one NB-DA Server software.
Official version	Unlimited sessions can be managed in one NB-DA Server software.

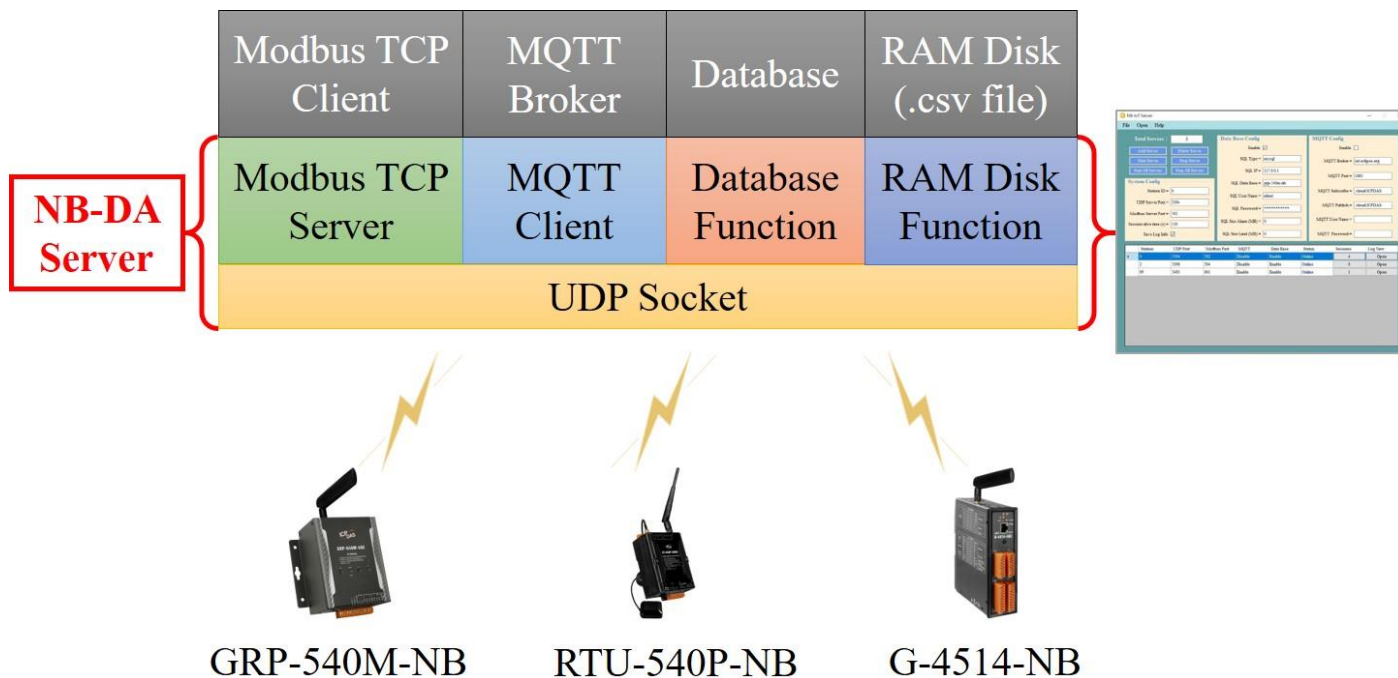
To order NB-DA Server license, please contact your distributor.

2. Hardware Requirement

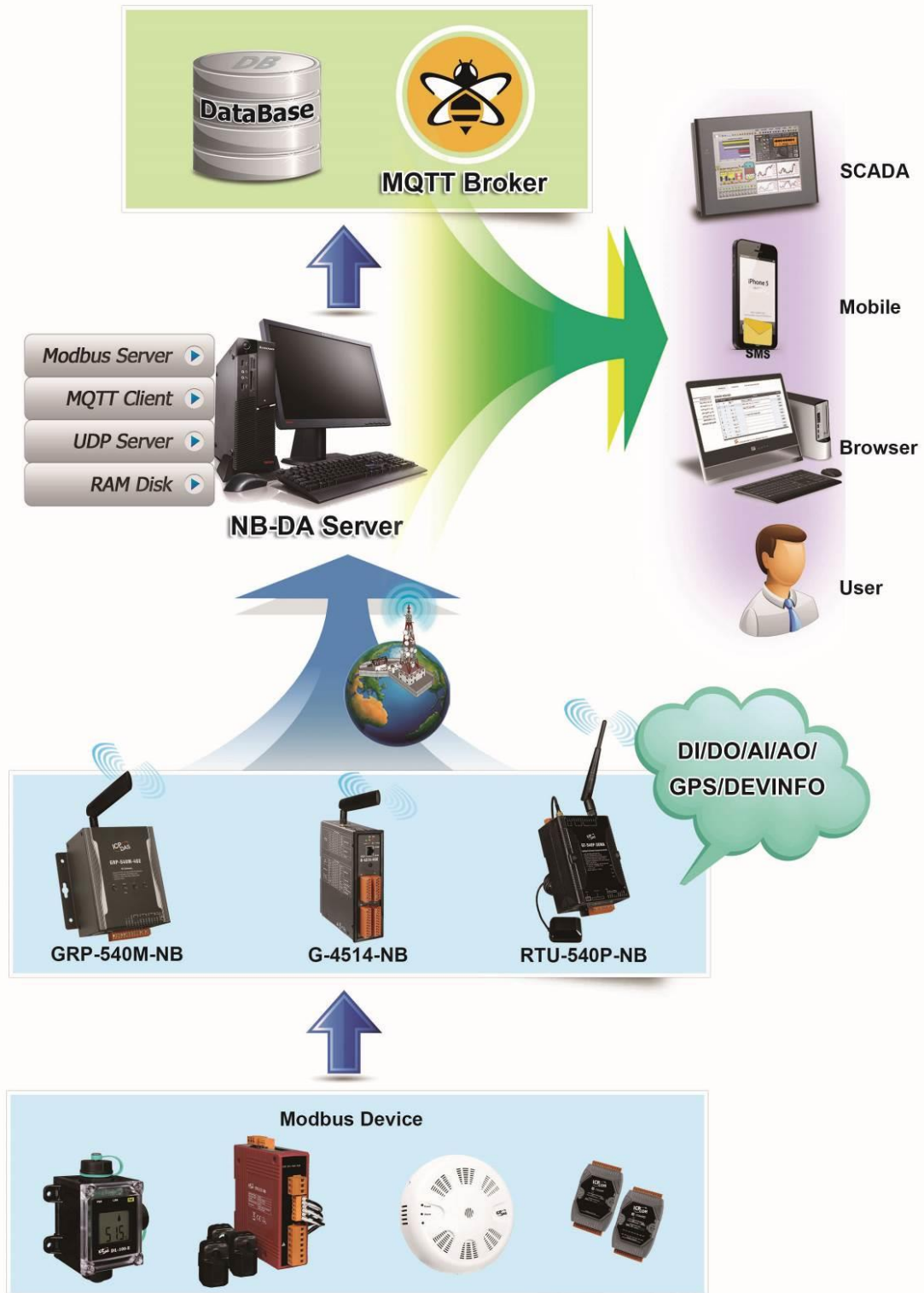
2.1 Software architecture

The NB-DA Server provides four interfaces—Modbus TCP, MQTT, Database, and RAM Disk (.csv file) to let user gets the data. Besides, user can use Modbus TCP Command, publish MQTT Message or store DO/AO control message in RAM Disk to control remote I/O. In the NB-DA Server, user can set multiple stations to collect data in different applications and every station can support 2000 devices online.

Device	Description
.NET Framework software	Require Microsoft .NET Framework 4.5 or higher
Microsoft Visual C++	Require 2008 Feature Pack Redistributable Package
ImDisk Virtual Disk Driver (option)	Used to create 100 MB RAM Disk. Every sessions' I/O data will be stored in ".csv" file.



2.2 Application architecture



2.3 System requirement

- **Software requirement:**

Operating system Version	Support
Windows XP	✘
Windows 7	✓
Windows 10	✓

- **Hardware requirement:**

Hardware Items	Requirement
CPU	1.0 GHz or higher
Memory	1.0 GB or higher
Hardware space	200 MB or higher

2.4 Product Support

Product Type	Description
RTU-540P-NB CR	Intelligent NB-IoT Remote Terminal Unit with GPS (RoHS)
GRP-540M-NB CR	LTE NB-IoT Gateway (RoHS)
G-4514-NB CR	NB-IoT Power Saving PAC with Solar charger (RoHS)

3. NB-DA Server operation

3.1 Main menu

Station ID **UDP Port** **Modbus Port** **MQTT Status** **Database Status** **Station Status** **Sessions** **Log View**

Station ID	UDP Port	Modbus Port	MQTT Status	Database Status	Station Status	Sessions	Log View
0	5394	502	Disable	Enable	Online	6	Open
2	5396	504	Disable	Enable	Online	0	Open
3	5397	505	Disable	Enable	Online	0	Open
4	5398	506	Enable	Enable	Online	0	Open
5	5399	507	Disable	Enable	Offline	0	Open
6	5400	508	Disable	Enable	Offline	0	Open
7	5401	509	Disable	Enable	Offline	0	Open
8	5402	510	Disable	Enable	Offline	0	Open
9	5493	601	Enable	Enable	Online	0	Open
10	5403	511	Enable	Enable	Online	0	Open
11	5404	512	Enable	Enable	Offline	0	Open
12	5405	513	Enable	Enable	Offline	0	Open

● Tool Menu:

- File: User can store the config as default from the window or read default config.
- Open: Can open “configure”, “gps”, “ image”, or “ logger” folder.
- Help: Show the info about this NB-DA Server.

● Function Button:

- Add Station: Set the “Station Config” and click this button to add station.
- Delete Station: Choose the station that need to be deleted in “All Station Info” area, then stop the station and click this button to delete the station’s config.

- Start Station: Choose the station that need to be started, then click this button to start the station.
- Start All Stations: Click this button to auto start all stations that are in the “All Station Info” area.
- Stop Station: Choose the station that need to be stopped, then click this button to stop the station.
- Stop All Stations: Click this button to auto stop all stations that are in the “All Station Info” area.

● Station Config:

■ System Config:

- Station ID: Used to identify the data that belongs to which station. (support 0~9 for now)
- UDP Server Port: The UDP server of every stations must have different server port.
- Modbus Server Port: The Modbus server of every stations must have different server port.
- Session alive time (s): Used to judge the session of this station is living or not.
- Save Log Info: Enable to store exceptional messages to the “logger” folder. (save by date)

■ Database Config:

- Enable: Enable Database function.
- SQL Type: Default as “mysql”. (no need to change it)
- SQL IP: The IP address of MySQL Server.
- SQL Data Base: The Database that will be used to store data. (please use different database in different station)
- SQL User Name: The user name to access the Database.
- SQL Password: The password to access the Database.
- SQL Size Alarm (MB): If the space of database over this value, show alarm in the “Database Status” and log message. (default is zero, means not enable this function)
- SQL Size Limit (MB): If the space of database over this value, the station won’t store any data. (default is zero, means not enable this function)

If set up SQL Size Alarm and SQL Size Limit, every stations will wait for a while in the beginning for checking Database size. The progress of checking database size will also show up in the Station Status field.

■ MQTT Config:

- Enable: Enable MQTT function.
- MQTT Broker: The Domain Name of MQTT Broker.
- MQTT Port: The port of MQTT Broker. (MQTT’s default port is 1883)
- MQTT Subscribe: The topic that used to receive DO/AO control message.
- MQTT Publish: The topic that used to publish DEVINFO/GPS/DI/AI message.
- MQTT User Name: The user name to access the MQTT Broker. (empty it if no user name)
- MQTT Password: The password to access the MQTT Broker. (empty it if no password)

● **All Station Info:**

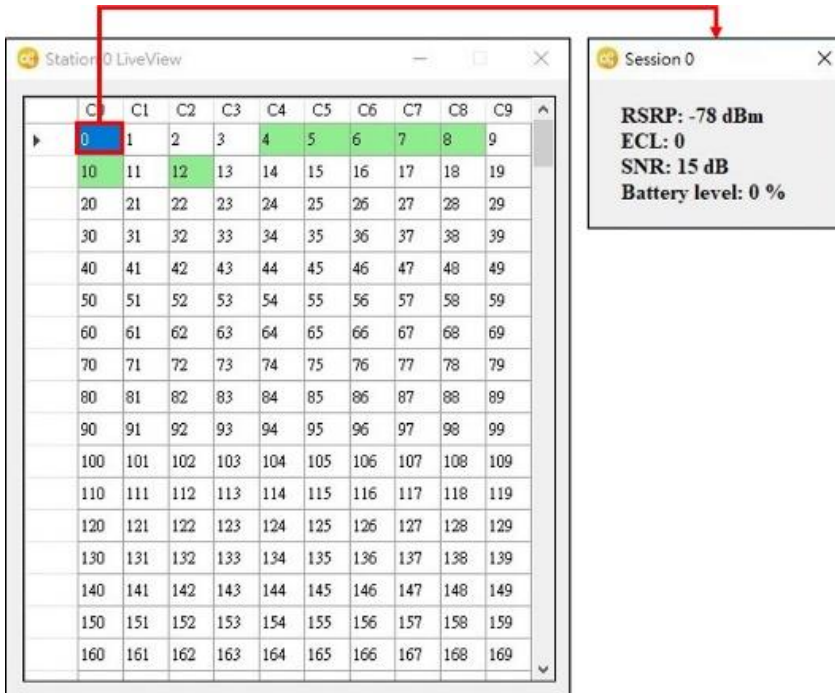
- Station ID: Every station's unique ID.
- UDP Port: The port that used to create UDP Server by every stations.
- Modbus Port: The port that used to create Modbus Server by every stations.
- MQTT Status: Show the MQTT function is enabled or not in every stations.

Message	Color of Message	Occurrence Time	Influence
Enable/Disable	Black	General	No
Overflow	Red	When the Buffer of MQTT message is full.	The Buffer won't store any messages before all messages being sent.

- Database Status: Show the Database function is enabled or not in every stations. (it will show up the database's usage if the station enabled SQL Size Alarm and SQL Size Limit)

Message	Color of Message	Occurrence Time	Influence
Enable/Disable	Black	General (if SQL Size Alarm and SQL Size Limit are zero)	No
Enable (XX%)	Black	General	No
DB Alarm (XX%)	Green	When the Database size is between SQL Size Alarm and SQL Size Limit.	No
DB Full (100%)	Red	When the Database size is over SQL Size Limit.	All messages in Buffer will be cleared and won't store any messages. It will check entire Database size every 60 mins.
Overflow	Red	When the Buffer of Database message is full.	The Buffer won't store any messages before all messages are sent.

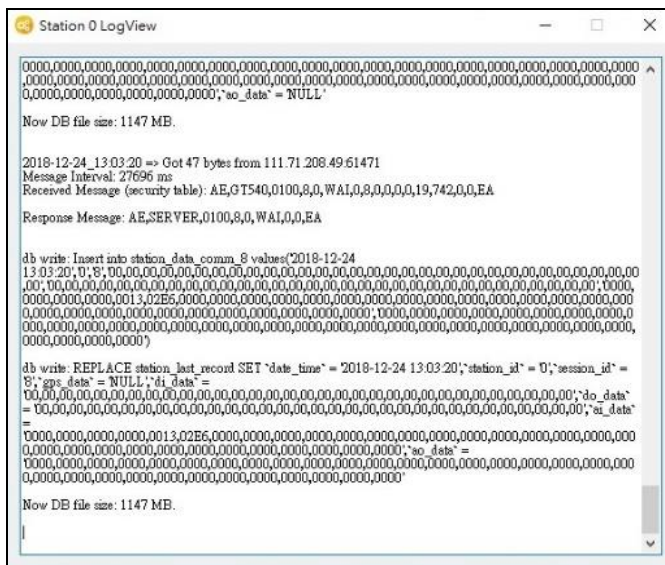
- Station Status: Show the station is running or stopping.
- Session Status: A button to open Live View window. Total living sessions will show on button.



Data Type	Data Range
RSRP	-140 ~ -44 dBm
ECL	0 ~ 2
SNR	-20 ~ 30 dB
Battery level	0 ~ 100 %

The living status of all sessions will show on this window. If sessions are online, they will have green background color. If sessions are offline, they will have white background color. If double-click one of the online sessions, it will show up the session's info in another window.

➤ Log View: A button to open Log View window.



3.2 Manage Stations

3.2.1 Add Station

The screenshot shows the NB-DA Server interface with the following components:

- Total Stations:** 4
- Buttons:** Add Station, Delete Station, Start Station, Stop Station, Start All Stations, Stop All Stations.
- System Config:** Station ID = 3, UDP Server Port = 5397, Modbus Server Port = 505, Session alive time (s) = 60, Save Log Info
- Database Config:** Enable , SQL Type = mysql, SQL IP = 192.168.12.2, SQL Data Base = nb_db, SQL User Name = root, SQL Password = *****
- MQTT Config:** Enable , MQTT Broker = iot.eclipse.org, MQTT Port = 1883, MQTT Subscribe = .cloud.ICPDAS.USEI, MQTT Publish = .cloud.ICPDAS.SERV, MQTT User Name = , MQTT Password =
- Table:**

Station	UDP Port	Modbus Port	MQTT	Database	Status	Sessions	Log View
0	5394	502	Disable	Disable	Online	0	Open
1	5395	503	Disable	Enable	Offline	0	Open
2	5396	504	Disable	Disable	Offline	0	Open
3	5397	505	Disable	Disable	Online	0	Open

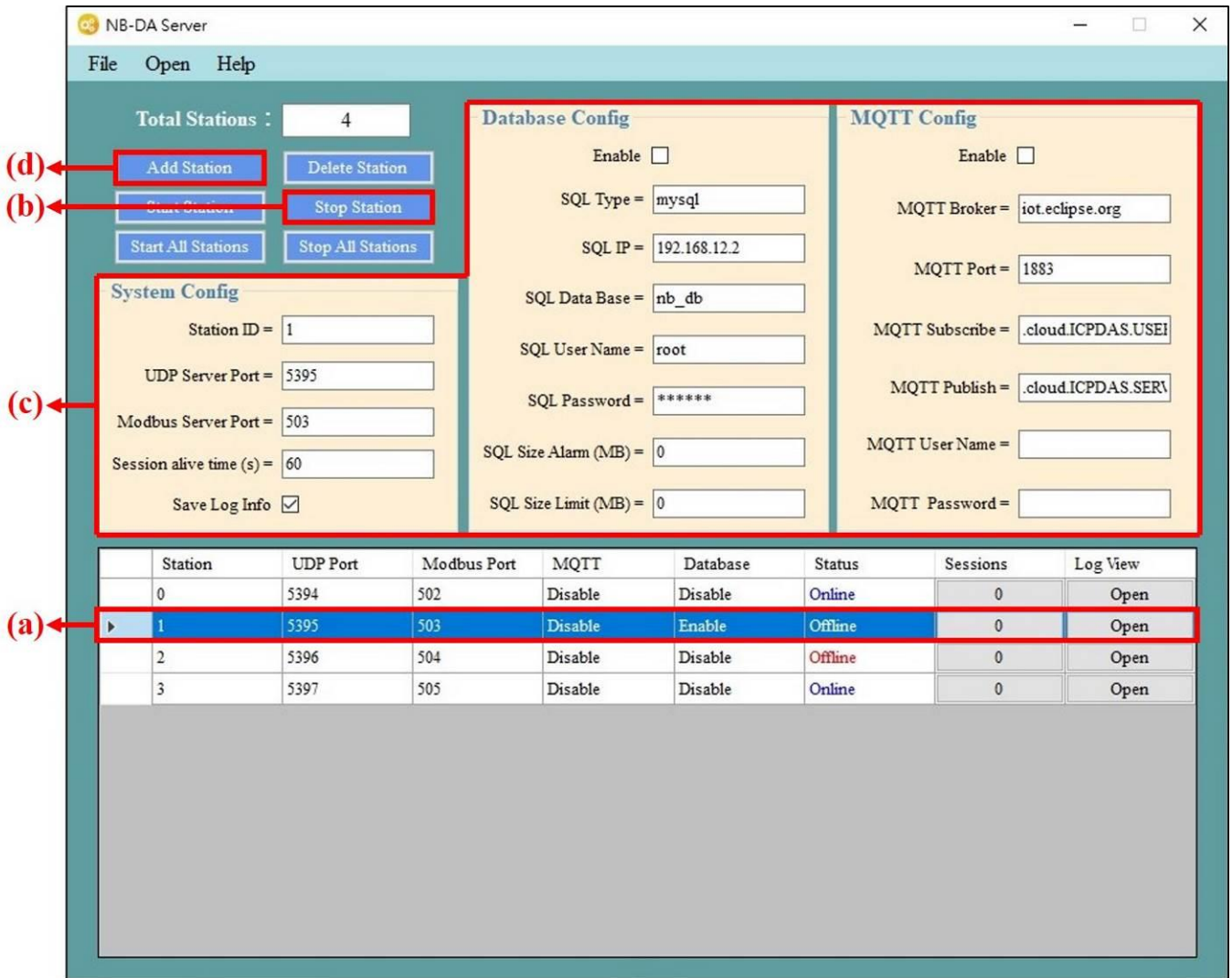
(a) Config parameters for station.

(b) Click “Add Station” button and it will create a station in the below area. “Status” will be “Offline”.

(c) Choose this new station and it will change background color to blue.

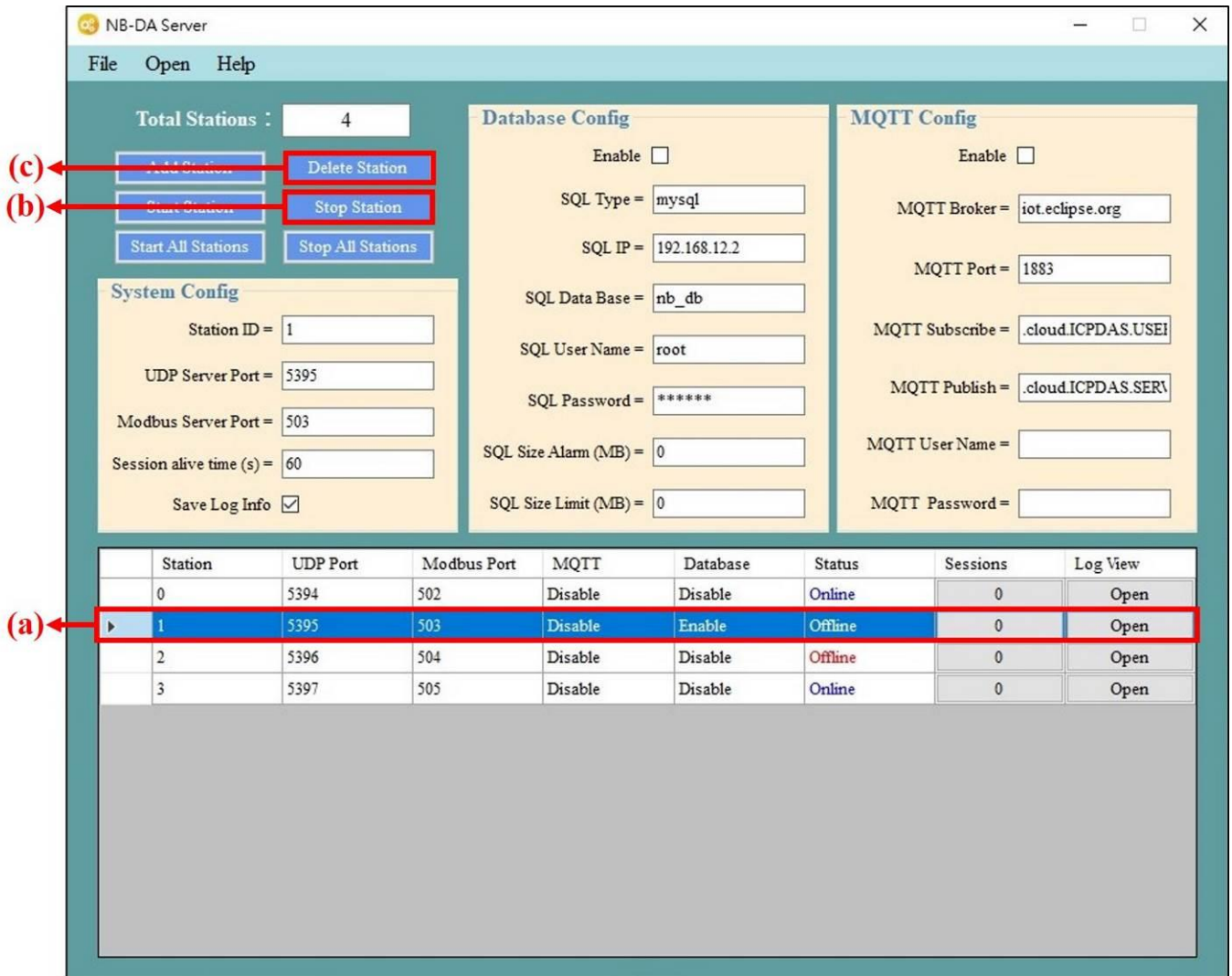
(d) Click “Start Station” button and the “Status” will change to “Online”.

3.2.2 Update Station



- (a) Choose the station that needs to be updated.
- (b) Click “Stop Station” button and the “Status” will change to “Offline”.
- (c) Change the station’s parameters.
- (d) Click “Add Station” button and the station’s parameters will be updated.

3.2.3 Delete Station



(a) Choose the station that needs to be deleted.

(b) Click “Stop Station” button and the “Status” will change to “Offline”.

(c) Click “Delete Station” button and the config of this station will be deleted.

3.3 Get Data from NB-DA Server

NB-DA Server provides four kinds of function—Database, MQTT Client, Modbus TCP Server, and RAM Disk (option) to let user get the data. The DEVINFO/GPS/DO/DI/AO/AI data, Session Live Info, and Data Update Info get by this four functions can be checked in the below table:

Function \ Data Type	DEVINFO	GPS	DO/DI/AO/AI	Session Live Info	Data Update Info
	Database	✗	✓	✓	✓
MQTT Client	✓	✓	✓	✓	✗
Modbus TCP Server	✗	✗	✓	✓	✓
RAM Disk (option)	✗	✓	✓	✓	✗

“Session Live Info” is used to check the session transmits data during the “Session alive time” or not.

“Data Update Info” is used to check the data of any session is new or old.

3.3.1 Database

NB-DA Server will auto create three data tables—“station_data_comm”, “station_data_modbus”, and “station_last_record” to store GPS data, I/O data, and last data (include GPS and I/O).

Data Table	station_data_comm	station_data_modbus	station_last_record	
Data Type	Session ID 0~1999	Session ID 0~1999	Session ID 0~1999	Session ID 2000
	GPS	DO/DI/AO/AI	GPS/DO/DI/AO/AI	Session Live Info (AI)
Info	\$GPRMC data of NMEA 0183 protocol	Each kinds of I/O has 32 values (empty I/O will be zero)	Last data with GPS and all I/O values (empty I/O will be zero)	125 AI values, every AI (16 bits) stands for 16 sessions’ live info (16 bits from right to left)

3.3.2 MQTT Client

User can use any MQTT Broker with this function. If user wants to get the data or send control message, the topic needs to be subscribed or published is like “[Topic of server publish]/[Station ID]/[Session ID]/[Type]”.

➤ The MQTT control message for DO/AO like below: **(user needs to publish data)**

Type	Topic	Data	Data example
DO	[Topic of server subscribe]/[Station ID]/[Session ID]/DO	1 byte for every DO, data length = 32, hex format. (set 00 for empty DO)	00010001000100010001000100010001 0001000100010001000100010000000000
AO	[Topic of server subscribe]/[Station ID]/[Session ID]/AO	2 bytes for every AO, data length = 32, hex format. (set 0000 for empty AO)	00000001000200030004000500060007 00080009001000110012001300140015 00160017001800190020002100220023 00240025002600270000000000000000

➤ The DEVINFO/GPS/DI/AI data like below: **(user needs to subscribe data)**

Type	Topic	Data	Data example
DEVINFO	[Topic of server publish]/[Station ID]/[Session ID]/DEVINFO	RSRP, ECL, SNR, Battery level	-80,0,16,0
GPS	[Topic of server publish]/[Station ID]/[Session ID]/GPS	\$GPRMC data of NMEA 0183 protocol	\$GPRMC:083559.00:A:4717:11437:N:0 0833:91522:E:0.004:77.52:091202:::A* 57
DI	[Topic of server publish]/[Station ID]/[Session ID]/DI	1 byte for every DI, data length = 32, hex format	00010001000100010001000100010001 00010001000100010001000000000000
AI	[Topic of server publish]/[Station ID]/[Session ID]/AI	2 bytes for every AI, data length = 32, hex format	00000001000200030004000500060007 00080009001000110012001300140015 00160017001800190020002100220023 00240025002600270000000000000000

Notice: In the AI data, data of session 2000 is used to store “Session Live Info”. “Session Live Info” has 125 AI values, every AI (16 bits) stands for 16 sessions’ live info (16 bits from right to left).

3.3.3 Modbus TCP Server

In the PC which is running NB-DA Server, user can use any Modbus TCP software to connect to the local IP (127.0.0.1) with the port of Modbus TCP Server (set on NB-DA Server). After connecting to Modbus TCP Server, user can send Modbus TCP command to get the values or set the values in the Modbus TCP Server.

● **Modbus Address :**

➤ **DO (Coil) :**

Address	Info
00000 ~ 00031	32 DOs for Session 0
00032 ~ 00063	32 DOs for Session 1
63936 ~ 63967	32 DOs for Session 1998
63968 ~ 63999	32 DOs for Session 1999
64000 ~ 65535	Reserved

➤ **DI (Discrete Inputs) :**

Address	Info
00000 ~ 00031	32 DIs for Session 0
00032 ~ 00063	32 DIs for Session 1
63936 ~ 63967	32 DIs for Session 1998
63968 ~ 63999	32 DIs for Session 1999
64000 ~ 65535	Reserved

➤ **AO (Holding Registers) :**

Address	Info
00000 ~ 00031	32 AOs for Session 0
00032 ~ 00063	32 AOs for Session 1
63936 ~ 63967	32 AOs for Session 1998
63968 ~ 63999	32 AOs for Session 1999
64000 ~ 65535	Reserved

➤ **AI (Input Registers) :**

Address	Info
00000 ~ 00031	32 AIs for Session 0
00032 ~ 00063	32 AIs for Session 1
63936 ~ 63967	32 AIs for Session 1998
63968 ~ 63999	32 AIs for Session 1999
64000 ~ 64124	125 AIs for Session 2000, use 2000 bits to save live status of session 0~1999
64125 ~ 64249	125 AIs, use 2000 bits to save receiving status of session 0~1999 (reverse bit when receiving new message)
64250 ~ 65535	Reserved

3.3.4 RAM Disk (option)

This function is an option for user, because it only work if user install “ImDisk Virtual Disk Driver”. If the user already install “ImDisk Virtual Disk Driver” in PC, the NB-DA Server will auto create 100 MB for “Z:\” when it is startup. The I/O data and GPS data will be store in .csv file and classify by station id and session id. The .csv file will be updated every time when the station received new data.

Example:

The DO data of Session 10 of Station 3 will be store in “Z:\NB-DA Server\3\10\DO.csv” and the .csv file will have 32 DO values (empty I/O will be zero).

Notice: If the PC reboot, the RAM Disk (“Z:\”) and all the data inside it will be cleared.

RAM Disk also provides all Sessions’ Live Info of every Station. The Info is stored in “Z:\NB-DA Server\[Station ID]\LIVE_LIST.csv” .

3.4 Control Remote Device I/O

NB-DA Server support three kinds of function—Modbus TCP Command, publish MQTT Message or store DO/AO control message in RAM Disk to let user control remote DO/AO.

3.4.1 Modbus TCP Command

In the PC which is running NB-DA Server, user can use any Modbus TCP software to connect to the local IP (127.0.0.1) with the port of Modbus TCP Server (set on NB-DA Server). After connecting to Modbus TCP Server, user can send Modbus TCP command to change the DO/AO values in the Modbus TCP Server. (the DO/AO address of every sessions can be checked in Section 3.3.3) After change the value of DO/AO, next time when the session ask for DO/AO, the values will be updated by the device of this session.

3.4.2 Publish MQTT Message

If the station of NB-DA Server enables MQTT function, user can connect to the same MQTT Broker and publish the DO/AO message of specific session with the Topic that subscribe by station. When the station receives the DO/AO message, it will also update the DO/AO values in the Modbus TCP Server, and next time when the session ask for DO/AO, the values will be update by the device of this session. The MQTT control message for DO/AO like below:

Type	Topic	Data	Data example
DO	[Topic of server subscribe]/[Station ID]/[Session ID]/DO	1 byte for every DO, data length = 32, hex format. (set 00 for empty DO)	00010001000100010001000100010001 0001000100010001000100010000000000
AO	[Topic of server subscribe]/[Station ID]/[Session ID]/AO	2 bytes for every AO, data length = 32, hex format. (set 0000 for empty AO)	00000001000200030004000500060007 00080009001000110012001300140015 00160017001800190020002100220023 00240025002600270000000000000000

3.4.3 Store DO/AO control message in RAM Disk

User can control remote I/O by writing message line-by-line to the "DO_CONTROL.csv" or "AO_CONTROL.csv" in "Z:\NB-DA Server\[Station ID]". Control message for DO/AO like below:

Type	Message	Message example
DO	[Session ID],[DO Number],[DO Data]	11,3,1,0,1
AO	[Session ID],[AO Number],[AO Data]	20,3,555,0,65535

User can know success or fail by checking the session's DO/AO file in the RAM Disk.

Appendix A. Revision History

This chapter provides revision history information to this document.

The table below shows the revision history.

Version	Date	Author	Description of changes
1.0.0	2019-01-01	Shepard Lee	The First Release Revision
1.0.1	2019-05-17	Shepard Lee	Add remote I/O control through RAM Disk Add Session Live List to RAM Disk
1.0.2	2019-06-05	Shepard Lee	Correct some error