**Version 1.0**

|  |
| --- |
|  |

http://www.jastecm.com/jbiz/common/images/logo.jpg

**History**

|  |  |  |  |
| --- | --- | --- | --- |
| ***Version*** | | ***Descriptions*** | ***Date*** |
| *0.1* | Initial Writing | | *2018.07.03* |
|  |  | |  |
|  |  | |  |
|  |  | |  |
|  |  | |  |
|  |  | |  |
|  |  | |  |
|  |  | |  |
|  |  | |  |
|  |  | |  |
|  |  | |  |
|  |  | |  |
|  |  | |  |
|  |  | |  |
|  |  | |  |
|  |  | |  |
|  |  | |  |
|  |  | |  |
|  |  | |  |

Contents

[1. Introduction 5](#_Toc518397450)

[1.1. Terminology 5](#_Toc518397451)

[1.2 Basic Flow 6](#_Toc518397452)

[1.2.1 Authentication 7](#_Toc518397453)

[1.2.3 Subscription 7](#_Toc518397454)

[1.2.4 Publish 7](#_Toc518397455)

[1.2.5 Items of Publish 7](#_Toc518397456)

[1.2.6 Items of subscription 7](#_Toc518397457)

[1.2.7 Firmware Update over the air 7](#_Toc518397458)

[1.2.8 Set actuator. (Sub ID $00) 8](#_Toc518397459)

[1.2.9 Get actuator support list (Sub ID $01) 8](#_Toc518397460)

[1.2.10 Open BCM. (Sub ID $02) 11](#_Toc518397461)

[1.2.11 Close BCM. (Sub ID $03) 12](#_Toc518397462)

[1.2.12 Get Supported Sensor Status. (Sub ID $04) 12](#_Toc518397463)

[1.2.13 Get Sensor Status. (Sub ID $05) 12](#_Toc518397464)

[1.2.14 BCM open 확인.(Sub ID $06) 15](#_Toc518397465)

[1.2.15 Set Auto Close Type. (Sub ID $07) 15](#_Toc518397466)

[1.2.16 Get Auto Close Type. (Sub ID $08) 16](#_Toc518397467)

[1.2.17 Set BCM open/close Interval. (Sub ID $09) 17](#_Toc518397468)

[1.2.18 Get BCM open/close Interval. (Sub ID $0A) 18](#_Toc518397469)

[1.2.19 Open BCM Interval. (Sub ID $0B) 18](#_Toc518397470)

[1.2.20 Get BCM 기록 조회. 19](#_Toc518397471)

[1.2.21 Get last open cmd. (Sub ID $0C) 20](#_Toc518397472)

[1.2.22 Set BCM Init Type. (Sub ID $0D) 20](#_Toc518397473)

[1.2.23 Get BCM Init Type. (Sub ID $0E) 21](#_Toc518397474)

[~~2~~ ~~USB-BT data transfer.~~ 22](#_Toc518397475)

[~~2.2~~ ~~Single frame data transfer.~~ 22](#_Toc518397476)

[~~2.3~~ ~~Multi frame data transfer.~~ 24](#_Toc518397477)

[~~2.4~~ ~~예제.~~ 25](#_Toc518397478)

[3 Read SRS DTC. 25](#_Toc518397479)

[4 Read ABS sensor. 27](#_Toc518397480)

[4.2 Get support item. 27](#_Toc518397481)

[4.3 Get sensor value. 27](#_Toc518397482)

[5 비밀 번호 설정 기능. 28](#_Toc518397483)

[5.2 기본 비밀 번호. 28](#_Toc518397484)

[5.2.3 기본 비밀 번호. (Sub ID $70) 28](#_Toc518397485)

[5.2.4 기본 비밀 번호 변경. (Sub ID $71) 29](#_Toc518397486)

[~~5.3~~ ~~중요 비밀 번호.~~ 30](#_Toc518397487)

[~~5.3.3~~ ~~중요 비밀 번호. (Sub ID $72)~~ 30](#_Toc518397488)

[~~5.3.4~~ ~~중요 비밀 번호 변경. (Sub ID $73)~~ 31](#_Toc518397489)

[~~5.3.5~~ ~~중요 비밀 번호 범위.~~ 31](#_Toc518397490)

[5.4 비밀번호 초기화. 32](#_Toc518397491)

[5.4.3 Key 요청. (Sub ID $74) 32](#_Toc518397492)

[5.4.4 Key 확인. (Sub ID $75) 32](#_Toc518397493)

[5.4.5 초기화 범위. 33](#_Toc518397494)

[6 질문 사항. 33](#_Toc518397495)

[6.2 Door close 가 발생하는 조건. 33](#_Toc518397496)

[6.3 Bcm type 질문. 34](#_Toc518397497)

[6.4 비밀번호 적용 범위 34](#_Toc518397498)

# Introduction

## **Terminology**

MQTT - *MQTT is a machine-to-machine (M2M)/"Internet of Things" connectivity protocol. It was designed as an extremely lightweight publish/subscribe messaging transport ,* <http://mqtt.org>

OBD- **On-board diagnostics** (**OBD**) is an [automotive](https://en.wikipedia.org/wiki/Automotive) term referring to a vehicle's self-diagnostic and reporting capability. OBD systems give the vehicle owner or repair technician access to the status of the various vehicle subsystems , [*https://en.wikipedia.org/wiki/On-board\_diagnostics*](https://en.wikipedia.org/wiki/On-board_diagnostics)

OBD-II - OBD-II standard specifies the type of diagnostic connector and its pinout, the electrical signalling protocols available, and the messaging format. It also provides a candidate list of vehicle parameters to monitor along with how to encode the data for each, [*https://en.wikipedia.org/wiki/On-board\_diagnostics*](https://en.wikipedia.org/wiki/On-board_diagnostics)

Protocol buffers - Protocol buffers are Google's language-neutral, platform-neutral, extensible mechanism for serializing structured data, <https://developers.google.com/protocol-buffers/>

LTE - **Long-Term Evolution** (**LTE**) is a [standard](https://en.wikipedia.org/wiki/Technical_standard) for high-speed [wireless](https://en.wikipedia.org/wiki/Wireless) communication for [mobile devices](https://en.wikipedia.org/wiki/Mobile_device) and data terminals, based on the [GSM](https://en.wikipedia.org/wiki/GSM)/[EDGE](https://en.wikipedia.org/wiki/Enhanced_Data_Rates_for_GSM_Evolution) and [UMTS](https://en.wikipedia.org/wiki/UMTS)/[HSPA](https://en.wikipedia.org/wiki/High_Speed_Packet_Access) technologies, <https://en.wikipedia.org/wiki/LTE_(telecommunication)>

3G- short for third generation, is the third generation of [wireless](https://en.wikipedia.org/wiki/Wireless) mobile telecommunications technology. It is the upgrade for [2G](https://en.wikipedia.org/wiki/2G) and [2.5G](https://en.wikipedia.org/wiki/2.5G) [GPRS](https://en.wikipedia.org/wiki/GPRS) networks, for faster internet speed. This is based on a set of standards used for mobile devices and mobile telecommunications use services and networks that comply with the International Mobile Telecommunications-2000 (IMT-2000) specifications by the [International Telecommunication Union](https://en.wikipedia.org/wiki/International_Telecommunication_Union), <https://en.wikipedia.org/wiki/3G>

**OBD-II PID** ([On-board diagnostics](https://en.wikipedia.org/wiki/On-board_diagnostics) **Parameter IDs**) are codes used to request data from a vehicle, used as a diagnostic tool, <https://en.wikipedia.org/wiki/OBD-II_PIDs>

VIN - A **vehicle identification number** (**VIN**) is a unique code, including a [serial number](https://en.wikipedia.org/wiki/Serial_number), used by the [automotive industry](https://en.wikipedia.org/wiki/Automotive_industry) to identify individual [motor vehicles](https://en.wikipedia.org/wiki/Motor_vehicle), [towed vehicles](https://en.wikipedia.org/wiki/Trailer_(vehicle)), [motorcycles](https://en.wikipedia.org/wiki/Motorcycle), [scooters](https://en.wikipedia.org/wiki/Scooter_(motorcycle)) and [mopeds](https://en.wikipedia.org/wiki/Moped), as defined in [ISO](https://en.wikipedia.org/wiki/International_Organization_for_Standardization) 3779:2009, <https://en.wikipedia.org/wiki/Vehicle_identification_number>

DTC - [diagnostic trouble code](https://en.wikipedia.org/wiki/Table_of_OBD-II_Codes) , <https://en.wikipedia.org/wiki/On-board_diagnostics#EOBD_fault_codes>

**TLS - Transport Layer Security** (**TLS**) – and its predecessor, **Secure Sockets Layer** (**SSL**), which is now deprecated by the [Internet Engineering Task Force](https://en.wikipedia.org/wiki/Internet_Engineering_Task_Force) [[1]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-1) (IETF) – are [cryptographic protocols](https://en.wikipedia.org/wiki/Cryptographic_protocol) that provide [communications security](https://en.wikipedia.org/wiki/Communications_security) over a [computer network](https://en.wikipedia.org/wiki/Computer_network)

## **Basic Flow**

Device

MQTT Broker

8.Response the Result

7.RPC Request

6.Publish Ack

5.Publish Message

4. Subscription Ack

3.Subscribe the Topic for RPC

2. Authetnication Ack

1.Request Authetication

## **Authentication**

* Test Server : 221.140.5.186 1883
* Supported Authentication Method For Device : ID/Password, TLS V1.2 For Mutual authentication
* For Change of MQTT Server, Discuss it with Customer support team
* For TLS 1.2, Discuss it with Customer support team

## **Subscription**

* TX : vonxc/rx/{device\_id} ex) /vonxc/rx/vonxc112345
* For Change of MQTT Server Topic it with Customer support team

## **Publish**

* TX : vonxc/rx/{device\_id} ex) /vonxc/rx/vonxc112345
* For Change of MQTT Server Topic it with Customer support team

## **Items of Publish**

* Dynamic Value - VIN , DTC etc
* Fixed Value – Engine Speed, Vehicle Speed , Throttle Position,
* Position Value – List of GPS Position ( latitude longitude degree )
* Response of device Control from server
* This is defined at other document “vonxc.proto”
* Without notification to customer, we can change this document

## **Items of subscription**

* Device Control Command from server
* Reboot, Beep, File Write , Firmware Update ,,
* This is defined at other document “vonxc.proto”

## **Firmware Update over the air**

* We provide Firmware update path
* When you send the command to device, you can update firmware by FTP Protocol
* FTP server is located in my company
* For Change of FTP Server it with Customer support team

# Sample MQTT Client

## **2.1 Basic Flow**

Sample MQTT Client

Device

MQTT Broker

## 

Verification

Send Remote Procedure Call

Request specific device

Items of Publishing

Items of Publishing

Response from device

## **2.2 Purpose**

* For Clear Understanding about MQTT Protocol
* For Clear Understanding about Protocol Buffer
* For Clear Understanding about vonxc MQTT Protocol
* We provide Full Source Code for Our MQTT Protocol Development By Python 2.7 Language
* For Other Language , Just We Provide Protocol Buffer Decoding/Encoding

## **Protocol Buffer**

## **Folders**

./vonxc-radio-message-format – Protocol Buffer Contract folder

./vonxc-radio-message-format/vonxc.proto – Protocol Buffer Contract

./vonxc-radio-message-format/vonxc.options– Protocol Buffer Contract option

./ vonxc – Python code for Protocol buffer

vonxclib.py = Python code for Protocol Buffer

mqttserver.py – Python code for MQTT client

./dist/mqttserver –win32 executables with mqttserver.py

## **Protocol Buffer Contract**

* vonxc.proto , vonxc.options files
* Without notification, we can change it
* RadioVehicleMessage is topmost Data Structure
* This RadioVehicleMessage is used between device and mqtt sample client