
LoRaWAN Network Server Demonstration: Configuration Command Definitions

1	History.....	2
2	Introduction.....	3
3	Conventions	3
3.1	Format of textual elements.....	3
4	Command console	4
4.1	Command line syntax.....	4
4.2	Command console control commands.....	5
5	Server command line syntax.....	6
5.1	Servers.....	6
5.2	Network server.....	7
6	Server commands (usable while server is running).....	8
6.1	Commands common to all servers.....	8
6.2	Network server commands.....	10
6.3	Application server commands.....	16
6.4	Customer server commands	17
6.5	Network controller commands	18
6.5.1	ADR service.....	Error! Bookmark not defined.
7	Description of service names.....	18
8	Region names	19
9	Example configuration.....	19
10	Glossary	20
11	References	24

1 History

Revision	Modification / Remarks / Motive	Author
1.0	Document created	DRo

2 Introduction

The document describes the general operation and interfaces of Version 2.0 of the LoRa™ servers (including the network controller).

3 Conventions

[] indicates an optional parameter

<text> indicates that a value must be entered

| indicates alternatives e.g. a|b

() indicates a group of items that must either all be included or all excluded.

3.1 Format of textual elements

<eui>	Hex text. Leading zeros need not be included. Separators of ':' and '-' may be used. For example, 9abc, 00009a:bc, 00:00-00:00-00:00-9a:bc
<key>	Hex text. Leading zeros need not be included. Separators of ':' and '-' may be used. For example, 9abc, 00009a:bc, 00:00-00:00-00:00-9a:bc
<app id>	<eui> <app text id>
<app text id>	<name> [<owner>]
<hex text>	a sequence of pairs of hex digits e.g. '11 1f f0', excluding the quotation marks.
<host address>	A domain name or an IP V4 address. E.g. 127.0.0.1 or iotdev.semnet.dom
<port address>	A host address followed by a colon followed by a non-zero port number. E.g. 127.0.0.1:4000

<service list>	a sequence of zero or more <service change>
<service change>	[- +]servicename (no space). A '-' removes the service; a '+' character or neither adds it.
<service name>	Service names are listed in Section 7.

An entered host address must be of the form 1.2.3.4 or semtech.com.

An entered port address must be of the form 1.2.3.4:4500 or semtech.com:4500.

When a hexadecimal (base sixteen) value is used it must be entered as a text string. The letters 'a'-'f' (in either case) represent values '10' to '16' respectively. The characters "0x" may be present immediately before the first (most significant) digit but are not required.

Signed positive values need not be prefixed by '+'.

Negative values must be prefixed by '-'.

4 Command console

The LoRa command console program allows a user to configure the servers. The program may be on the same host as the configured server program or, if the server's configured value 'allowremoteconfiguration' is on (true), on another host.

The console stores the addresses of 4 servers ('ns', 'as', 'cs' and 'nc'). The user may set these at console start time or subsequently.

One of the 4 servers is currently selected. All commands, apart from the server select commands, are sent to the selected server.

4.1 Command line syntax

`./loracmd† [-ns <NS address>] [-as <AS address including port>] [-cs <CS address including port>] [-nc <NC address including port>]`

Parameter order is unimportant.

The user can select a server by typing the code for the server ('ns' 'as' 'cs' or 'nc')

Typing the code for any server other than NS followed by an IP address (including port number) selects that server; it also allows its future reselection by typing just the code. After selection, the user may enter console commands for the selected server.

[†] The LoRa command console is called './loracmd' on the Linux build and

Examples:

```
./loracmd -as 127.0.0.1:4000 -cs 127.0.0.1:5000 -ns 127.0.0.1  
./loracmd -ns 127.0.0.1 -as 127.0.0.1:4000 -cs 127.0.0.1:5000  
./loracmd -ns 127.0.0.1 -as 127.0.0.1:4000
```

4.2 Command console control commands

ns [<host address> <port address>]	Selects the current NS and, if an address is entered, updates the console's record of its address. The NS address can either be that of a host (server.a.com) or a port address (server.a.com:1701). The port address, if used, must be the server's JSON/UDP port.
nc [<port address>]	Selects the current NC and, optionally, updates the console's record of its port's address
as [<port address>]	Selects the current AS and, optionally, updates the console's record of its address
cs [<port address>]	Selects the current CS and, optionally, updates the console's record of its address
include <filename>	Executes the commands contained in the indicated file. Blank lines are ignored. Comments begin with the '#' character and continue until the end of the line. <i>In Version 2.0.0, an included file cannot include the server select commands 'ns', 'nc', 'as', 'cs'.</i>

5 Server command line syntax

5.1 Servers

The following command line (start) parameters are accepted by all the LoRa servers, unless noted.

All parameters are optional, unless noted.

- console prints information to the system console. It should not be used if the program does not have access to a system console. If this parameter is not used, the server will not print to the console.
- noconsole prohibits the server from printing information to the system console. The server does not, by default, print to the console but this parameter may be entered in order to explicitly specify this.
- dbuser <name> sets the name by which the server logs into its relational database. If the 'dbuser' parameter is not used, the user will be prompted to enter the database username. The <name>/<password> combination must match one that is accepted by the server's database. *The parameter is required by all servers.*
- dbpass <password> sets the password used when the server logs into its relational database. If the 'dbpass' parameter is not used, the user will be prompted to enter the database password. The <name>/<password> combination must match one that is accepted by the server's database. *The parameter is optional but, if the parameter is missing, the server will prompt for the user to enter the value.*
- dbname <name> sets the name of the relational database. *If the parameter is not entered, the value listed in Table 1 is used.*

Server	Default database name
Network server	lora_network
Application server	lora_application
Customer server	lora_customer
Network controller	lora_networkcontroller

Table 1: Default database names used by LoRa servers

- dbhost <IP address> sets the host on which the server's relational database server resides. *If the parameter is not entered, 'localhost' is assumed.*
- log <file name> commands the server to print log information to the log file.

-port <number>	sets the port on which the server receives UDP and TCP messages to the indicated (decimal) value. <i>This is required by all servers (including the NC) except the network server. . If used on the network server, the parameter is equivalent to using '-port-gwmp <number> -port-json <number+1>'</i>
-major	sets the server to log only major errors
-minor	sets the server to log major and minor errors
-monitor	sets the server to log minor events, for example a mote joining the network, as well as major and minor errors.
-verbose	sets the server to log routine operation, for example printing each message received. The server also logs errors and minor events.

Example:

```
./loraAS -console -port 4000 -log logAS.txt
```

5.2 Network server

In addition to the start-up parameters listed in Section 5.1, the following parameters must be used.

-netid <id>	sets the server's LoRa network id to the indicated (hexadecimal) value.
-port-gwmp <number>	sets the UDP port on which the network server receives Gateway Message Protocol (GWMP) messages to the indicated (decimal) value.
-port-json <number>	sets the UDP and TCP ports on which the network service receives JSON messages to the indicated (decimal) value.

Example:

```
./loraNS -noconsole -netid 24 -log logNS.txt
```

6 Server commands (usable while server is running)

6.1 Commands common to all servers

app add <eui> <app name> [<app owner>]	Adds the application to the system. <i>This command must be executed before any 'app server' command is executed for the application.</i>
app delete <eui> <app name> [<app owner>]	Deletes the application from the system
app list	Lists the applications
app list full	Lists the applications and their servers
app server add <application eui> <remote server port address> active passive <service list>	Configures a server to transmit data associated with the application the server port. <service list> defines the data types to be sent to it. If 'active' is selected, this server attempts to establish a TCP connection to the remote server; if 'passive' is selected, it does not. <i>The 'app add ...' command must have been executed before this command is executed.</i>
app server set < application eui> <remote server port address> <service list>	Changes the services provided.
app server delete [<eui> <name> <owner>] <remote server port address>	Removes the server from the application
app server list [<eui> <name> <owner>]	Lists either all servers or the servers serving the specified application.
connection list	Lists the connections of the server to other LoRa servers.
connection test	Commands the selected server to send messages to the servers to which it has a TCP connection and list those from which it receives a response. <i>This command is not implemented in Version 2.0.0 of the server.</i>
deleteAllData yesREALLY	Causes ALL configuration data to be deleted. <i>This log</i> major minor monitor verbose console help Sets the amount of data printed to the server's log file. 'verbose' and 'monitor' are recommended for fault finding. 'major' and 'minor' are recommended for normal operation.

log file <filename>	Closes the existing log file and opens a new file. It is important that this is done periodically, to avoid the file becoming too big. The old file should either be deleted or backed up.
ping	Causes the server's version number and name (e.g. 'Network Server') to be displayed on the console.
server add <remote server port address> active passive <service list>	Configures a server to transmit data associated with no application the server port. <service list> defines the data types to be sent to it. If 'active' is selected, this server attempts to establish a TCP connection to the remote server; if 'passive' is selected, it does not.
server set <remote server port address> <service list>	Changes the services provided.
server delete <remote server port address>	Removes the server from the list
server list	Lists all servers associated with no application.
set	Prints the settable parameters and their respective values.
set <name> <value>	Sets configured value <name> to equal <value>. <name> is case insensitive.
set allowremoteconfiguration on off	If on, the server will accept a configuration command when the command's source is not the local host.

6.2 Network server commands

gateway add <eui> [region <r>] [lat <la>] [long <lo>] [alt <a>] [allowgps true|false]

Adds a gateway of EUI <eui>.

Each of the parameters 'region', 'lat', 'long', 'alt' and 'allowgps' is optional; however if either or both 'lat' or 'long' is missing, both 'lat' and 'long' will be ignored.

<r> must be one of the regions listed in Section 8.

<la> is the gateway's latitude in degrees and fractions of degree. It may be suffixed by 'N' or 'S' to indicate North or South. If neither suffix is present, <la> is taken to be signed with positive indicating North. '51.499' and '51.499N' are both valid values.

<lo> is the gateway's longitude in degrees and fractions of degree. It may be suffixed by 'E' or 'W' to indicate East or West. If neither suffix is present, the <lo> is taken to be signed with positive indicating East. '-0.1247', and '0.1247W' are both valid values.

<alt> is the gateway's altitude above sea level in metres.

If 'allowgps' is followed by 'true', the NS will update its record of the gateway's position by averaging position information received from the gateway.

If 'allowgps' is followed by 'false', the NS will only update its record of the gateway's position when the 'lat', 'long' and, optionally, 'alt' configuration parameters are entered.

If 'allowgps' is not present and both 'lat' and 'long' are present, allowgps will be set to false.

If 'allowgps' is not present and at least one of 'lat' and 'long' are absent, allowgps will be set to true.

gateway set <eui> [region <r>] [lat <la>] [long <lo>] [alt <a>] [allowgps true|false]

Sets the attributes of Gateway <eui>.

Each of the parameters 'region', 'lat', 'long', 'alt' and 'allowgps' is optional; however if either or both 'lat' or 'long' is missing, both 'lat' and 'long' will be ignored.

<r> must be one of the regions listed in Section 8.

<la> is the gateway's latitude in degrees and fractions of degree. It may be suffixed by 'N' or 'S' to indicate North or South. If neither suffix is present, <la> is taken to be signed with positive indicating North. '51.499' and '51.499N' are both valid values.

<lo> is the gateway's longitude in degrees and fractions of degree. It may be suffixed by 'E' or 'W' to indicate East or West. If neither suffix is present, the <lo> is taken to be signed with positive indicating East. '-0.1247', and '0.1247W' are both valid values.

<alt> is the gateway's altitude above sea level in metres.

If 'allowgps' is followed by 'true', the NS will update its record of the gateway's position by averaging position information received from the gateway.

If 'allowgps' is followed by 'false', the NS will only update its record of the gateway's position when the 'lat', 'long' and, optionally, 'alt' configuration parameters are entered.

If 'lat' and 'long' are present, and 'allowgps' is not, 'allowgps' will be set to false.

delete <eui>

Removes Gateway <eui> from the list of NS gateways.

gateway list

Lists the active gateways

mote add <eui> app <app eui>|(<app name> <app owner>) [key <authentication session key>]

Adds a personalized (provisioned) mote to the application server. If the authentication key is not included, the default value is used. *It is not necessary to use this command when 'over the air' activation is used.*

mote add <networkAddress> [<authentication key>]

Adds a personalized (provisioned) mote to the network server. The server must contain an application with EUI of 0. If the authentication key is not included, the default value is used. The mote EUI is set to equal its network address. *It is important that the network address chosen is not part of the range used by the 'over the air' protocol. This is not necessary for 'over the air' activation.*

(Deprecated)

mote reset <eui>

Informs the server that the mote has been reset.

mote delete <eui>

Removes the mote from the server

mote list

Lists the motes

```
mote set <eui> datarate <dataRate> [mask <channelMask>] [maskcontrol <maskControl>]  
[power <txPowerCode>] [uplinktxtries <tries>]
```

Sets the mote's data rate, channel mask, power and the number of times that the mote transmits an unacknowledged confirmed uplink frame. The 'name <value>' pairs may be entered in any order.

<dataRate> is a decimal integer between 0 and 15. This parameter is mandatory. 'datarate' may be abbreviated to 'dr'.

<channelMask> is a bit mask expressed as a hexadecimal number between 0 and 0xFFFF[‡]. The default value is 0x1F

<maskControl> is a number between 0 and 7. It sets the block of 16 channels to which <channelMask> refers. Only used when a mote has more than 16 channels. The default value is zero.

<txPower> is a decimal integer between 0 and 15. The default value is 1.

<uplinktxtries> is a decimal integer between 0 and 15. A value of 0 or omission of the <uplinktxtries> parameter signifies no change

The meaning of the parameter values are defined in Reference [1].

```
mote maccmd <eui> data <hex text>
```

Sends a MAC command to the indicated mote.

```
set autoCreateMotes on|off
```

If 'off' and a data frame is received from a mote that is not provisioned (personalized) and has not completed an 'over the air' activation sequence, the frame is discarded. If 'on' and the frame is authenticated (using the default authentication key), the server creates a provisioned mote and accepts the frame. The mote's application EUI will be zero.

[‡] A '0x' prefix to the value is optional.

set defaultGatewayRegion americas902 china779 europe433 europe863	Sets the region of any gateways that are either automatically created or are created by the configuration command 'gateway add'. The regions are described in Section 8.
set gatewayTxPower_dBm <integer power in dBm>	Sets the gateway transmit power in dBm. Only integer values can be used.
set netDelayUp_ms <network transmission delay in ms from gateway to network server>	Sets the maximum expected delay from any gateway to this server.
set netDelayDown_ms <network transmission delay in ms from network server to gateway >	Sets the maximum expected delay from this server to any gateway.
set netDelayBoth_ms <network transmission delay in ms from network server to gateway >	Equivalent to entering both 'set netdelayup_ms' and 'set netdelaydown_ms'.
set moteReceive window <0 1>	Sets the transmission window to be used by all gateways. Either '0', first window, or '1', second window.
set defaultMoteChannelMask <mask>	Sets the channel mask (ChMask) value that is transmitted to the mote in the LoRaWAN LinkADRReq MAC command [1].
set defaultMoteChannelMaskControl <number>	Sets the channel mask control (ChMaskCntl) value that is transmitted to the mote in the LoRaWAN LinkADRReq MAC command [1].
set maxTxsofUplinkFrame <number>	Sets the maximum number of times that the mote will transmit an uplink frame that has not been acknowledged. A value of either '1' or '0' represents one transmission.
set defaultAuthenticationKey <value>	Sets the default authentication key.
set timeToAssumeMoteLost_s <value>	Sets the number of seconds before the NS decides that the mote has lost communications and begins testing a wider range of inferred sequence numbers. If <value> is zero this function is disabled.
set timeToAssumeMoteReset_s <value>	Sets the number of seconds before the NS decides that the mote has been reset and begins testing for inferred sequence numbers near to zero. If <value> is zero this function is disabled.
set moteMissSeqNoSearchLimit <value>	Sets the range of values for the most significant 16 bits of the inferred sequence word that are tested when a frame that has failed authentication is received from a mote.

set moteMissSeqNoSearchRetries <value>	Sets the number of times that each possible value of the most significant 16 bits of the inferred sequence word is tested when a frame that has failed authentication is received from a mote.
set moteResetSeqNoSearchLimit <value>	Sets the range of values for the most significant 16 bits of the inferred sequence word that are tested when a frame a mote is assumed to have been reset.

6.3 Application server commands

mote add <eui> [<encryption session key>]	Adds a <u>personalized</u> (provisioned) mote to the application server. The server must contain an application with EUI of 0. If the encryption key is not included, the default value is used. <i>This is not necessary for 'over the air' activation. (Deprecated)</i>
mote add <eui> p app <app eui>(<app name> <app owner>) netaddr <LoRa network address> key [<encryption session key>]	Adds a <u>personalized</u> (provisioned) mote to the application server. If the encryption key is not included, the default value is used. <i>This is not necessary for 'over the air' activation.</i>
mote add <eui> ota app <app eui>(<app name> <app owner>) key <app key>	Adds an 'over the air' mote to the system.
mote add <eui> <app eui> <appkey>	Adds an 'over the air' mote to the system. (Deprecated)
mote add <eui> <app name> [<app owner>] <appkey>	Adds an 'over the air' mote to the system. The app name and owner must be registered. If the owner of the app is blank, the <app owner> parameter must be omitted. (Deprecated)
mote reset <eui>	Informs the server that the mote's transmit sequence number has been reset to zero.
mote delete <eui>	Removes the mote from the server
mote list [active]	Lists the personalized (provisioned) motes
mote list ota	Lists the OTA (join) motes
set autocreatemotes on off	If 'off' and a data frame from a mote that is not provisioned and has not completed an 'over the air' activation sequence, the frame is discarded. If 'on', the server creates a personalized (provisioned) mote and accepts the frame. The mote's application EUI will be zero.
set allowduplicatemotenonce on off	If 'off' a 'join' request from an 'over the air' mote that contains a previously used mote 'nonce' will be rejected.
set defaultencryptionkey <value>	Sets the default authentication key. Value is a hex string, with optional ':' separators. E.g. 'set defaultencryptionkey 12:34'

6.4 Customer server commands

mote add <eui> app <app eui> (<app name> <app owner>)	Adds a mote to the customer server.
mote add <eui> [<app eui>]	Adds a mote to the server. If the application EUI is not included, it is assumed to be zero. (Deprecated)
mote add <eui> <app name> [<app owner>]	Adds a mote to the server. The app name and owner must be registered. If the owner of the app is blank, the <app owner> parameter must be omitted. (Deprecated)
mote delete <eui>	Removes the mote from the server
mote list	Lists the motes
mote send <eui> port <LoRa port number> data <hex text>	Sends a command to the indicated mote
sqloutput on off	If 'on', commands the server to write received data into its relational database.
fileoutput [on] <filename>	Commands the server to write received application data, mote transmit metadata and gateway receive metadata into the ASCII files <filename>.appData.txt, <filename>.motetx.txt <filename>.gatewayrx.txt respectively.
fileoutput on off	If 'on', commands the server to write received data into the previously configured ASCII files. If 'off', commands the server to stop writing data to the current files.

6.5 Network controller commands

NOTE: Configuration set by mote service commands is always set to its default value when the NC restarts.

set defaultMoteChannelMask <mask>	Sets the default channel mask (ChMask) value that is transmitted to the mote in the LoRaWAN LinkADRReq MAC command [1].
set defaultMoteChannelMaskControl <number>	Sets the channel mask control (ChMaskCntl) value that is transmitted to the mote in the LoRaWAN LinkADRReq MAC command [1].
set "maxTxOfUplinkFrame <number>	Sets the maximum number of times that the mote will transmit an uplink frame that has not been acknowledged. A value of either '1' or '0' represents one transmission.
alg app <app name> [<app owner>] alg <alg name> enable true false	Enables or disables a NC algorithm's operation on the motes of an application.
alg app <app EUI> alg <alg name> enable true false	Enables or disables a NC algorithm's operation on the motes of an application.
alg app <app name> [<app owner>] list	Displays a lists showing, for each installed algorithm, whether it is enabled for the application.
alg app <app EUI> list	Displays a lists showing, for each installed algorithm, whether it is enabled for the application.
alg list	Displays a list of all configured applications, showing, for each installed algorithm, whether it is enabled for that application.
mote add <eui> app <app eui> (<app name> <app owner>)	Adds a mote to the network controller.
mote add <eui> [<app eui>]	Adds a mote to the network controller. If the application EUI is not included, the application EUI is assumed to be zero. (Deprecated)
mote add <eui> <app name> [<app owner>]	Adds a mote to the network controller. The app name and owner must be registered. If the owner of the app is blank, the <app owner> parameter must be omitted. (Deprecated)
mote delete <eui>	Removes the mote from the network controller
mote list	Lists the motes

7 Description of service names

user	The remote server receives upstream user (application) data
motetx	The remote server receives data describing the LoRa PHY layer transmission parameters used by the mote.
gwrx	The remote server receives data describing characteristics of the signal received by the gateway.
joinserver	The remote server performs 'application' functions of the 'over the air' activation protocol.
joinmonitor	The remote server is informed when a mote has completed the 'over the air' activation protocol and has been admitted to the LoRa network
downstream	The remote server is closer to the mote than the local server.
maccmd	The remote server receives upstream header extensions received from the mote
gwst	The remote server receives gateway status information.

8 Region names

americas902	902-928MHz ISM band specified by Part 15 of Title 47 the United States Federal Communications Commission (FCC) document 'Code of Federal Regulations' [2].
china779	779-787MHz ISM band licenced by the People's Republic of China.
europa433	433MHz LPD433 ITU Region 1 ISM band, specified by ETSI EN 300 220-1 [3]
europa863	866-870MHz SRD860 specified by ECC Recommendation 70-03 [4]

9 Example configuration

Reference [5] includes an example configuration.

10 Glossary

'/':	The construct 'a/b' is used when Protocol 'a' is transported by Protocol 'b'.
ADR:	Adaptive Data Rate. ADR observes the quality of the signal received by the mote and changes the mote's spreading factor and transmit power in order to optimise the time and energy required for the mote to transmit a frame.
Application:	An application is identified by an 'application EUI'. Each mote is a member of a single application. The configuration rules that set the remote server to which information is forwarded (for example the AS to which an NS forwards are received frame) are set for each application.
AS:	The LoRa application server
ASCII:	American Standard Code for Information Interchange. A widely used standard for representing Latin text, Arabic numerals and punctuation as binary values.
CEPT:	European Conference of Postal and Telecommunications Administrations. <i>The acronym is derived from the French version of its name, 'Conférence européenne des administrations des postes et des télécommunications'.</i>
Command Console:	The LoRa command console is a program that allows a user to configure LoRa servers.
CS:	The LoRa customer server
dB:	decibel; a logarithmic ratio of power. Defined by Bell Laboratories
dBm	A logarithmic measure of power, decibel relative to 1mW
Downstream:	Toward the mote
End-device:	Synonymous with 'mote'
EUI:	Extended Unique Identifier. In this document 'EUI' refers to a value from the 'EUI-64' number space managed by the IEEE.
ECC:	Electronic Communications Committee of CEPT
Exception:	A programming construct in C++ where a thread encounters a 'throw' statement. The thread ceases to execute its existing function and any calling functions until a 'catch' statement is encountered.
Gateway:	A LoRa gateway is transmits LoRa frames to, and receives LoRa frames from, LoRa motes

IEEE:	Institution of Electrical and Electronic Engineers (www.ieee.org).
IETF:	Internet Engineering Task Force (www.ietf.org).
IP:	Internet Protocol
IP port address	An IP address or host name and either a UDP or a TCP port number. This document represents a port address in the form <IP address>:<port number> or <host name>:<port number>. E.g. 1.2.3.4:4500 or a.com:4500.
ISM:	Industrial, Scientific and Medical radio band(s). The bands are allocated by treaty to industrial, scientific and medical purposes other than communication. In addition, the use of the bands for low power communication is permitted.
ITU:	International Telecommunications Union
Join:	A colloquial name for 'Over The Air' activation.
Join request frame:	A LoRa frame sent as the initial part of the OTA activation protocol. The frame contains the mote's EUI, its application's EUI and its device-nonce (a 16 bit random number).
Join accept frame	A LoRa frame sent as the concluding part of the OTA activation protocol. The frame contains the mote's LoRa network address, its network Id and its application nonce (a 24 bit random number).
JSON:	JavaScript Object Notation. JSON is a textual based method of representing name, value pairs. The value of an object may itself be a JSON object. Within LoRa, JSON objects contain only ASCII characters.
JSON object	A JSON name, value pair.
Key:	In cryptography, a key is a piece of information (a parameter) that determines the functional output of a cryptographic algorithm or cipher. Without a key, the algorithm would produce no useful result.
LDP:	Low power device
LoRa:	Long Range. Defined by the LoRa Alliance
LoRa Alliance:	The industry body that defines the LoRaWAN protocol, (http://lora-alliance.org/)
LoRa port:	Any user data transmitted to or received from the mote is associated with a 'port' number. User data to or from LoRa Port 0 is MAC command or MAC status data. The remaining 255 LoRa port values are available to the mote user.

LoRaWAN:	The protocol by which a LoRa mote communicates with a LoRa gateway. LoRaWAN is defined in Reference [1].
MAC:	Media Access Control
MAC command:	A command transmitted to the mote. A MAC command is transmitted to the mote either in the LoRa frame 'header option' area or as user data to LoRa Port 0. Multiple commands may be transmitted in a single frame.
MAC status:	Status information received from the mote. A MAC status message is transmitted by the mote either in the LoRa frame 'header option' area or as user data from LoRa Port 0. Multiple status messages may be transmitted in a single frame.
Metadata:	LoRa Metadata refers to information about the transmission or reception of a LoRa frame.
Mote:	A LoRa end device. A LoRa mote communicates with a LoRa Gateway using the LoRa MAC or LoRa WAN protocol.
NC:	The LoRa network controller
Network id:	The 'network id' of a mote is its 'network address' shifted right by 25 bits, leaving 7 bit value.
Network address:	The LoRa network address is a 32 bit value contained in the LoRa frame that identifies its source or destination mote. The network address need be unique only within the transmission range of a mote or gateway and is distinct from the mote EUI.
NS:	The LoRa network server
OTA:	Over the air
Over the air:	One of two methods of adding a LoRa mote to a LoRa network. In the OTA method, the mote is configured with a mote EUI, an application EUI and a 128 bit cypher key ('appKey'). Handshaking between the mote and the LoRa servers causes a 32 bit LoRa network address and two 128 bit session keys to be generated. One session key (the 'authentication' key) is known to the mote and the NS. The other (the 'encryption' key) is known to the mote and the AS.
Personalization:	One of two methods of adding a LoRa mote to a LoRa network. The mote is configured with its network address and its authentication and encryption keys. The mote's EUI is always equal to its network address and the application EUI is always zero.
Provisioning:	A synonym for 'personalization'
Rx:	Receive

Spreading factor:	A parameter of a LoRa transmission. Two to the power of 'spreading factor' 'on the air' bits are transmitted to represent each frame bit.
SRD:	Short Range Device
TCP:	Transmission Control Protocol. A connection based protocol for transporting a sequence of bytes. While the connection exists, the content is guaranteed to be delivered in order and without loss or corruption.
Tx:	Transmit
UDP:	User Datagram protocol: a simple protocol for transporting data packets. Delivery is not guaranteed. In addition the order of receipt is not necessarily the same as the order of transmission.
upstream:	Away from the mote
UTC	Co-ordinated Universal Time; also known as Greenwich Mean Time and Zulu

11 References

Each trademark is the property of its owner.

- [1] LoRa Alliance, “LoRaWAN Specification,” LoRa Alliance, 2015.
- [2] Federal Communications Commission, Title 47 of the Code of Federal Regulations, 2014.
- [3] European Telecommunications Standards Institute, Electromagnetic compatibility and Radio spectrum Matters (ERM); ETSI EN 300 220-1 V2.4.1., 2012-01.
- [4] Electronic Communications Committee of CEPT, “ERC Recommendation 70-03,” April 2015.
- [5] Semtech Ltd, “LoRaWAN Network Server Demonstration: Installation Guide,” 2015.

© Semtech 2015

All rights reserved. Reproduction in whole or in part is prohibited without the prior written consent of the copyright owner. The information presented in this document does not form part of any quotation or contract, is believed to be accurate and reliable and may be changed without notice. No liability will be accepted by the publisher for any consequence of its use. Publication thereof does not convey nor imply any license under patent or other industrial or intellectual property rights. Semtech assumes no responsibility or liability whatsoever for any failure or unexpected operation resulting from misuse, neglect improper installation, repair or improper handling or unusual physical or electrical stress including, but not limited to, exposure to parameters beyond the specified maximum ratings or operation outside the specified range.

SEMTECH PRODUCTS ARE NOT DESIGNED, INTENDED, AUTHORIZED OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT APPLICATIONS, DEVICES OR SYSTEMS OR OTHER CRITICAL APPLICATIONS. INCLUSION OF SEMTECH PRODUCTS IN SUCH APPLICATIONS IS UNDERSTOOD TO BE UNDERTAKEN SOLELY AT THE CUSTOMER'S OWN RISK. Should a customer purchase or use Semtech products for any such unauthorized application, the customer shall indemnify and hold Semtech and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs damages and attorney fees which could arise.

Contact Information

Semtech Corporation
Wireless Sensing and Timing Products Division
200 Flynn Road, Camarillo, CA 93012
Phone: (805) 498-2111 Fax: (805) 498-3804
E-mail: support_rf_na@semtech.com
Internet: <http://www.semtech.com>