

MeshCore KISS Modem Protocol

MeshCore KISS Modem Protocol

Standard KISS TNC firmware for MeshCore LoRa radios. Compatible with any KISS client (Direwolf, APRSdroid, YAAC, etc.) for sending and receiving raw packets. MeshCore-specific extensions (cryptography, radio configuration, telemetry) are available through the standard SetHardware (0x06) command.

Serial Configuration

115200 baud, 8N1, no flow control.

Frame Format

Standard KISS framing per the KA9Q/K3MC specification.

Byte	Name	Description
0xC0	FEND	Frame delimiter
0xDB	FESC	Escape character
0xDC	TFEND	Escaped FEND (FESC + TFEND = 0xC0)
0xDD	TFESC	Escaped FESC (FESC + TFESC = 0xDB)



Type Byte

The type byte is split into two nibbles:

Bits	Field	Description
7-4	Port	Port number (0 for single-port TNC)
3-0	Command	Command number

Maximum unescaped frame size: 512 bytes.

Standard KISS Commands

Host to TNC

Command	Value	Data	Description
Data	0x00	Raw packet	Queue packet for transmission
TXDELAY	0x01	Delay (1 byte)	Transmitter keyup delay in 10ms units (default: 50 = 500ms)
Persistence	0x02	P (1 byte)	CSMA persistence parameter 0-255 (default: 63)
SlotTime	0x03	Interval (1 byte)	CSMA slot interval in 10ms units (default: 10 = 100ms)
TXtail	0x04	Delay (1 byte)	Post-TX hold time in 10ms units (default: 0)
FullDuplex	0x05	Mode (1 byte)	0 = half duplex, nonzero = full duplex (default: 0)
SetHardware	0x06	Sub-command + data	MeshCore extensions (see below)
Return	0xFF	-	Exit KISS mode (no-op)

TNC to Host

Type	Value	Data	Description
Data	0x00	Raw packet	Received packet from radio

Data frames carry raw packet data only, with no metadata prepended. The Data command payload is limited to 255 bytes to match the MeshCore maximum transmission unit (MAX_TRANS_UNIT); frames larger than 255 bytes are silently dropped. The KISS specification recommends at least 1024 bytes for general-purpose TNCs; this modem is intended for MeshCore packets only, whose protocol MTU is 255 bytes.

CSMA Behavior

The TNC implements p-persistent CSMA for half-duplex operation:

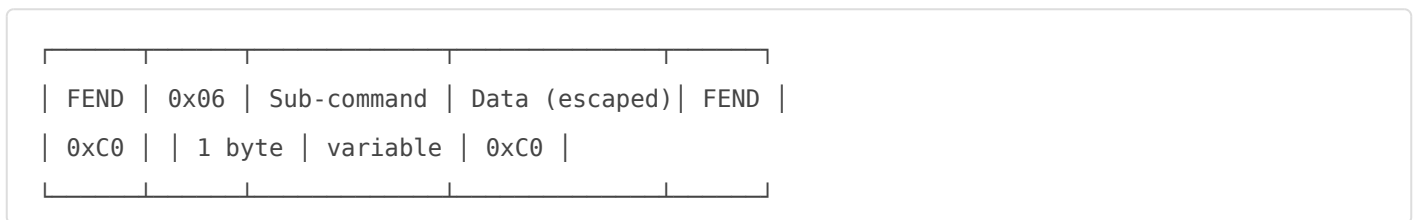
1. When a packet is queued, monitor carrier detect
2. When the channel clears, generate a random value 0-255
3. If the value is less than or equal to P (Persistence), wait TXDELAY then transmit
4. Otherwise, wait SlotTime and repeat from step 1

In full-duplex mode, CSMA is bypassed and packets transmit after TXDELAY.

SetHardware Extensions (0x06)

MeshCore-specific functionality uses the standard KISS SetHardware command. The first byte of SetHardware data is a sub-command. Standard KISS clients ignore these frames.

Frame Format



Request Sub-commands (Host to TNC)

Sub-command	Value	Data
GetIdentity	0x01	-
GetRandom	0x02	Length (1 byte, 1-64)
VerifySignature	0x03	PubKey (32) + Signature (64) + Data
SignData	0x04	Data to sign
EncryptData	0x05	Key (32) + Plaintext
DecryptData	0x06	Key (32) + MAC (2) + Ciphertext
KeyExchange	0x07	Remote PubKey (32)
Hash	0x08	Data to hash
SetRadio	0x09	Freq (4) + BW (4) + SF (1) + CR (1)
SetTxPower	0x0A	Power dBm (1)
GetRadio	0x0B	-
GetTxPower	0x0C	-

Sub-command	Value	Data
GetCurrentRssi	0x0D	-
IsChannelBusy	0x0E	-
GetAirtime	0x0F	Packet length (1)
GetNoiseFloor	0x10	-
GetVersion	0x11	-
GetStats	0x12	-
GetBattery	0x13	-
GetMCUTemp	0x14	-
GetSensors	0x15	Permissions (1)
GetDeviceName	0x16	-
Ping	0x17	-
Reboot	0x18	-
SetSignalReport	0x19	Enable (1): 0x00=disable, nonzero=enable
GetSignalReport	0x1A	-

Response Sub-commands (TNC to Host)

Response codes use the high-bit convention: `response = command | 0x80`. Generic and unsolicited responses use the `0xF0` + range.

Sub-command	Value	Data
Identity	0x81	PubKey (32)
Random	0x82	Random bytes (1-64)
Verify	0x83	Result (1): 0x00=invalid, 0x01=valid
Signature	0x84	Signature (64)
Encrypted	0x85	MAC (2) + Ciphertext
Decrypted	0x86	Plaintext
SharedSecret	0x87	Shared secret (32)
Hash	0x88	SHA-256 hash (32)
Radio	0x8B	Freq (4) + BW (4) + SF (1) + CR (1)
TxPower	0x8C	Power dBm (1)
CurrentRssi	0x8D	RSSI dBm (1, signed)
ChannelBusy	0x8E	Result (1): 0x00=clear, 0x01=busy

Sub-command	Value	Data
Airtime	0x8F	Milliseconds (4)
NoiseFloor	0x90	dBm (2, signed)
Version	0x91	Version (1) + Reserved (1)
Stats	0x92	RX (4) + TX (4) + Errors (4)
Battery	0x93	Millivolts (2)
MCUTemp	0x94	Temperature (2, signed)
Sensors	0x95	CayenneLPP payload
DeviceName	0x96	Name (variable, UTF-8)
Pong	0x97	-
SignalReport	0x9A	Status (1): 0x00=disabled, 0x01=enabled
OK	0xF0	-
Error	0xF1	Error code (1)
TxDone	0xF8	Result (1): 0x00=failed, 0x01=success
RxMeta	0xF9	SNR (1) + RSSI (1)

Error Codes

Code	Value	Description
InvalidLength	0x01	Request data too short
InvalidParam	0x02	Invalid parameter value
NoCallback	0x03	Feature not available
MacFailed	0x04	MAC verification failed
UnknownCmd	0x05	Unknown sub-command
EncryptFailed	0x06	Encryption failed
TxBusy	0x07	Transmitter busy; the radio could not accept the request because a transmission is already in progress

Unsolicited Events

The TNC sends these SetHardware frames without a preceding request:

TxDone (0xF8): Sent after a packet has been transmitted. Contains a single byte: 0x01 for success, 0x00 for failure.

RxMeta (0xF9): Sent immediately after each standard data frame (type 0x00) with metadata for the received packet. Contains SNR (1 byte, signed, value x4 for 0.25 dB precision) followed by RSSI (1 byte, signed, dBm). Enabled by default; can be toggled with SetSignalReport. Standard KISS clients ignore this frame.

Data Formats

Radio Parameters (SetRadio / Radio response)

All values little-endian.

Field	Size	Description
Frequency	4 bytes	Hz. The example value 869618000 (869.618 MHz) is in the EU 863-870 MHz band; US/Canada operators must use a 902-928 MHz value (e.g., 910525000) per FCC Part 15.247.
Bandwidth	4 bytes	Hz (e.g., 62500)
SF	1 byte	Spreading factor (5-12)
CR	1 byte	Coding rate (5-8)

Version (Version response)

Field	Size	Description
Version	1 byte	Firmware version
Reserved	1 byte	Always 0

Encrypted (Encrypted response)

Field	Size	Description
MAC	2 bytes	HMAC-SHA256 truncated to 2 bytes
Ciphertext	variable	AES-128 (ECB mode) block-encrypted data with zero padding

Airtime (Airtime response)

All values little-endian.

Field	Size	Description
-------	------	-------------

Airtime	4 bytes	uint32_t, estimated air time in milliseconds
---------	---------	--

Noise Floor (NoiseFloor response)

All values little-endian.

Field	Size	Description
Noise floor	2 bytes	int16_t, dBm (signed)

The modem recalibrates the noise floor every 2 seconds with an AGC reset every 30 seconds.

Stats (Stats response)

All values little-endian.

Field	Size	Description
RX	4 bytes	Packets received
TX	4 bytes	Packets transmitted
Errors	4 bytes	Receive errors

Battery (Battery response)

All values little-endian.

Field	Size	Description
Millivolts	2 bytes	uint16_t, battery voltage in mV

MCU Temperature (MCUTemp response)

All values little-endian.

Field	Size	Description
Temperature	2 bytes	int16_t, tenths of °C (e.g., 253 = 25.3°C)

Returns `NoCallback` error if the board does not support temperature readings.

Device Name (DeviceName response)

Field	Size	Description
Name	variable	UTF-8 string, no null terminator

Reboot

Sends an `OK` response, flushes serial, then reboots the device. The host should expect the connection to drop.

Sensor Permissions (GetSensors)

Bit	Value	Description
0	<code>0x01</code>	Base (battery)
1	<code>0x02</code>	Location (GPS)
2	<code>0x04</code>	Environment (temp, humidity, pressure)

Use `0x07` for all permissions.

Sensor Data (Sensors response)

Data returned in CayenneLPP format. See [CayenneLPP documentation](#) for parsing.

Cryptographic Algorithms

Operation	Algorithm
Identity / Signing / Verification	Ed25519
Key Exchange	X25519 (ECDH)
Encryption	AES-128 (ECB mode) block encryption with zero padding + HMAC-SHA256 (MAC truncated to 2 bytes)
Hashing	SHA-256

Notes

- Data payload limit (255 bytes) matches MeshCore MAX_TRANS_UNIT; no change needed for KISS "1024+ recommended" (that applies to general TNCs, not MeshCore)
 - Modem generates identity on first boot (stored in flash)
 - All multi-byte values are little-endian unless stated otherwise
 - SNR values in RxMeta are multiplied by 4 for 0.25 dB precision
 - TxDone is sent as a SetHardware event after each transmission
 - Standard KISS clients receive only type 0x00 data frames and can safely ignore all SetHardware (0x06) frames
 - See [packet format.md](#) for packet format
-

Revision #3

Created 2026-05-03 05:05:13 UTC by Mesh America Admin

Updated 2026-06-09 14:41:22 UTC by Mesh America Admin