INTERNATIONAL STANDARD

ISO/IEC 15149

First edition
2011-12-01

Information technology — Telecommunications and information exchange between systems — Magnetic field area network (MFAN)

Technologies de l'information — Téléinformatique — Réseau de zone de champ magnétique (MFAN)

Reference number
ISO/IEC 15149:2011(E)

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

Contents

Page

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

ISO/IEC 15149 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 6, *Telecommunications and information exchange between systems*.

Information technology — Telecommunications and information exchange between systems — Magnetic field area network (MFAN)

1 Scope

This International Standard specifies the physical layer and media access control layer protocols of a wireless network over a magnetic field in a low frequency band (~300 kHz), for wireless communication in harsh environments (i.e. around metal, underwater, underground, etc.).

The physical layer protocol is designed for the following scope:

— low carrier frequency for large magnetic field area and reliable communication in harsh environments;

— simple and robust modulation for a low implementation cost and error performance;

— variable coding and bandwidth for a link adaptation.

The media access control layer protocol is designed for the following scope:

— simple and efficient network topology for low power consumption;

— variable superframe structure for compact and efficient data transmission;

— dynamic address assignment for small packet size and efficient address management.

This International Standard supports several kbps data transmission in a wireless network within a distance of several metres. It can be applied to various services such as the following areas:

— in the environmental industry, to manage pollution levels in soil and water using wireless underground or underwater sensors;

— in the construction industry, to monitor the integrity of buildings and bridges using wireless, inner-corrosion sensors;

— in the consumer-electronics industry, to detect food spoilage in wet, airtight storage areas and to transfer the sensing data from the inside to the outside;

— in the agricultural industry, to manage the moisture level as well as mineral status in soil using buried wireless sensors;

— in the transportation industry, to manage road conditions and traffic information using wireless underground sensors.

2 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

2.1
magnetic field area network
MFAN
wireless network that provides reliable communication in harsh environments using magnetic field

2.2
magnetic field area network coordinator
MFAN-C
device that manages the connection and release of nodes within the communication area and the sending and receiving time of data in an MFAN

2.3
magnetic field area network node
MFAN-N
device except the coordinator that forms a network in an MFAN

3 Symbols and abbreviated terms

ARq	Association Request
ARs	Association Response
ARA	Association Response Acknowledgement
ASC	Association Status Check
ASK	Amplitude Shift Keying
ASRq	Association Status Request
ASRs	Association Status Response
ASRA	Association Status Response Acknowledgement
BPSK	Binary Phase Shift Keying
CRC	Cyclic Redundancy Check
DA	Data Acknowledgement
DaRq	Disassociation Request
DaRs	Disassociation Response
DaRA	Disassociation Response Acknowledgement
DRq	Data Request
DRs	Data Response
DRA	Data Response Acknowledgement
FCS	Frame Check Sequence
GSRq	Group ID Set-up Request
GSRs	Group ID Set-up Response
GSRA	Group ID Set-up Response Acknowledgement
HCS	Header Check Sequence
LSB	Least Significant Bit
MAC	Media Access Control
MFAN	Magnetic Field Area Network
MFAN-C	Magnetic Field Area Network Coordinator
MFAN-N	Magnetic Field Area Network Node
NRZ-L	Non-Return-to-Zero Level
PHY	PHYsical layer protocol
RA	Response Acknowledgement
RR	Response Request
SIFS	Short InterFrame Space
TDMA	Time Division Multiple Access
UID	Unique IDentifier

4 Overview

MFAN is a wireless communication network that can transmit and receive data over a magnetic field in a low frequency band. Wireless communication over a magnetic field enables reliable communication and extends the communication system coverage around metal, soil, and water. It is designed using those characteristics of the magnetic field communication. It uses a low carrier frequency for reliable communication and large magnetic field area in harsh environments, a simple and robust modulation like BPSK for a low implementation cost and error probability, and a dynamic coding technique like Manchester or NRZ-L coding for noise robustness. In essence, it provides several kbps data transmission within a distance of several meters.

Also, it uses a simple and efficient network topology like a star topology for low power consumption. The dynamic address assignment is used for the small packet size and efficient address management, and the adaptive link quality control is considered with variable data transmission speed and coding. The devices in MFAN are specified into two elements according to the role: MFAN-C for a coordinator, and MFAN-N for a node. In a network, there can be one coordinator. When a node joins the network, the coordinator assigns the time-slots for each device upon the node's request and the coordinator's decision. So, TDMA is considered for the data transmission.

As shown in Figure 1, MFAN-Ns are buried under the ground, and MFAN-C is placed above the ground. If MFAN-N receives the sensing data from the sensors, MFAN-N sends its received data to MFAN-C over a magnetic field. MFAN-C sends the received data from MFAN-N to the monitoring center by either wireless or wired communication for long distance.

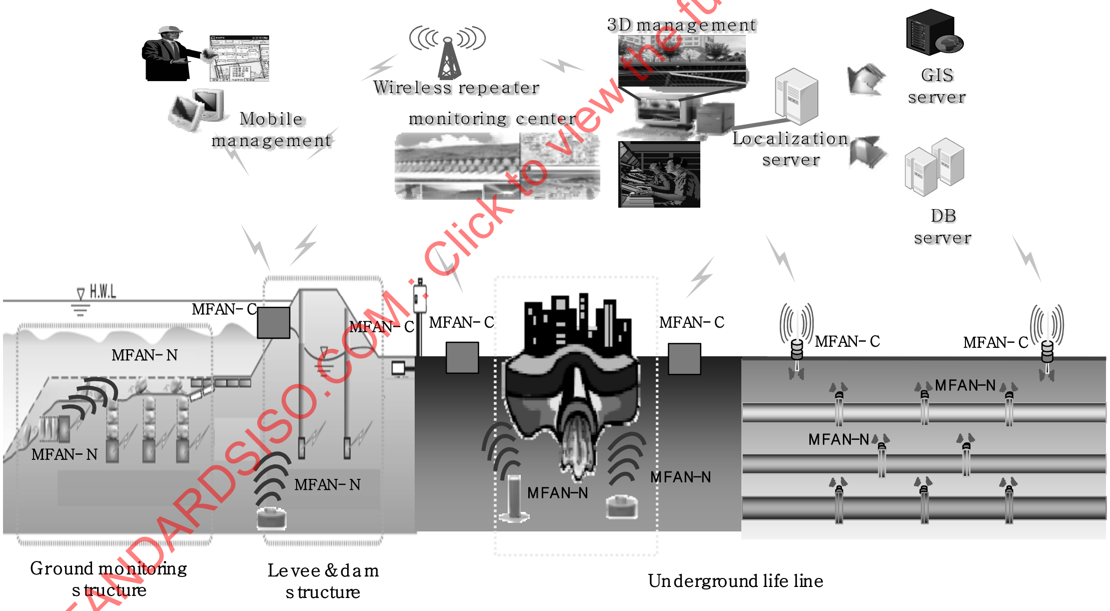


Figure 1 – Underground monitoring system

Wireless communication in harsh environments has been significantly required in various industries. It is difficult for a sensor node to transmit its data by a radio frequency around metal, soil, and water with the existing standards of wireless communication. MFAN is an alternative standard that enables several sensor nodes inside metal, soil, and water to transfer their data to a coordinator outside using the characteristics of magnetic field. Therefore it can be applied to various services in harsh environments.

For example, in ground status monitoring as shown in Figure 2, the sensor nodes can be buried under the ground to sense ground cave-in, ground sinking, land sliding, and so on. Another example of MFAN is the underground infrastructure management in Figure 3. In this example, the sensor nodes are attached to the pipes, and can detect gas or water leaks and notify its location. For the building and bridge management in Figure 4, the sensor nodes can be placed on beams and columns to detect the integrity of the structure. In pollution monitoring as shown in Figure 5, the sensor nodes can detect the quality of the soil and water. It can detect poisonous chemical, pH(Hydrogen ion exponent), and temperature by sensor nodes that are placed below ground and water.

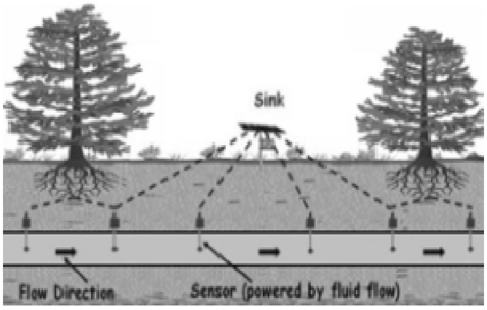


Figure 2 – Ground status monitoring

Figure 3 – Underground infrastructure management

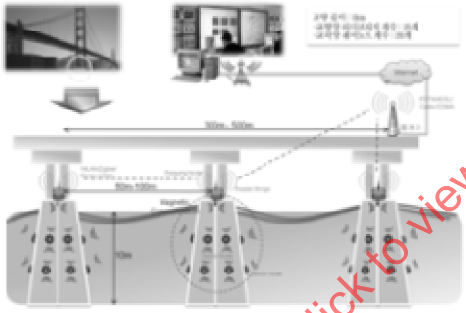


Figure 4 – Building & bridge management

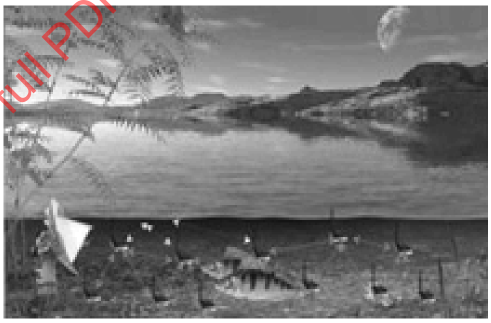


Figure 5 – Pollution monitoring

5 Network elements

5.1 General

The main elements of MFAN are divided into time and physical element. The time element refers to the superframe consisting of a request period, a response period, and a spontaneous period, and the physical element refers to the network consisting of MFAN-C and MFAN-Ns. The most basic one in the physical element is the node. Node is classified into two types: MFAN-C to manage the network and MFAN-N to communicate with MFAN-C.

Figures 6 and 7 show the structures of superframe and network which are the time and physical elements, respectively. The node that needs to be decided first in MFAN is MFAN-C, and the superframe begins with MFAN-C transmitting a request packet in the request period. MFAN-C is charged of managing the association, disassociation, release, and scheduling of MFAN-Ns. One MFAN can use one channel where only one node is utilized as MFAN-C and the rest of them become MFAN-N. The rest of the nodes in MFAN excluding MFAN-C become MFAN-N. Note that any nodes can become either MFAN-C or MFAN-N depending upon its role. Basically, a peer-to-peer connection between MFAN-C and MFAN-N is considered.

5.2 Time element

The time element used in MFAN is the time slot of the TDMA method. MFAN-C manages the MFAN-N group that transmits data in the response period, and the time slots are self-arranged by the selected MFAN-Ns. The superframe of MFAN, as shown in Figure 6, consists of a request period, a response period, and a spontaneous period, and the lengths of the request and response period are variable. The superframe begins with MFAN-C transmitting a RR packet to MFAN-Ns in the request period.

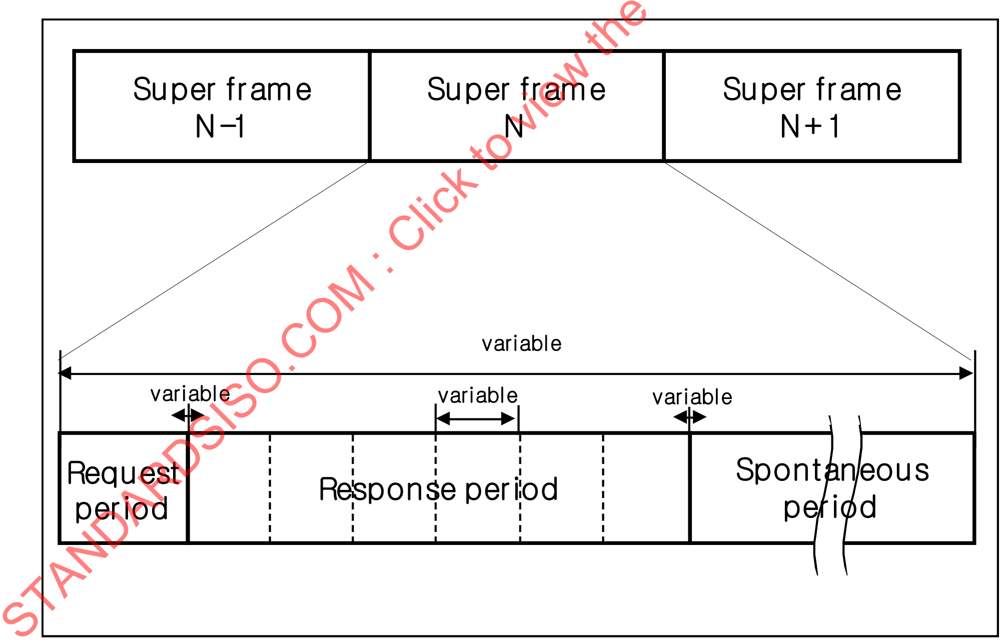


Figure 6 – MFAN superframe structure

The RR packet has information which MFAN-Ns can send response packets during response periods, and the selected MFAN-Ns can transmit the response packet in the response period according to the RR packet information.

5.2.1 Request period

In the request period, MFAN-C transmits the RR packet with the information about the usage of MFAN-Ns in order for MFAN-N to send the response packet during response periods.

5.2.2 Response period

In the response period, MFAN-N can transmit response packet according to the received RR packet of MFAN-C, and the response period can be divided into several time slots according to the number of the selected MFAN-Ns in MFAN. Each time slot length is variable according to the length of the response frame and the acknowledgement. If the MFAN-C schedules a response period, the slot number is decided by the order of the divided time slot. Otherwise the slot number is zero. MFAN-C assigns time slots to either MFAN-N or a particular group for the use of the response period, and the nodes in the assigned group independently transmit the data frame in the response period.

5.2.3 Spontaneous period

The spontaneous period begins when there is no node transmitting the response packet for a certain period of time. In this period, nodes can transmit data even without MFAN-C's request. This period is maintained until MFAN-C transmits a request packet.

5.2.4 Network activation

The superframe of MFAN is divided into the request period, the response period, and the spontaneous period. MFAN-C and MFAN-Ns in MFAN operate in each period as follows:

5.2.4.1 Request packet transmission within the request period

In the request period, MFAN-C sends the RR packet to MFAN-Ns. Based on this, the MFAN-N that have received the RR packet decide whether to transmit response packets in the response period. MFAN-C can determine the MFAN-N group to transmit in the response period.

5.2.4.2 Response packet transmission within the response period

The MFAN-Ns selected by MFAN-C can transmit the response packet in the response period. When MFAN-N transmits the response packet in the response period, MFAN-C that has received the response packet transmits the RA packet. MFAN-N that has not received the RA packet transmits response packets every time-slot until it receives a RA packet from MFAN-C.

5.2.4.3 Data packet transmission in the spontaneous period

A spontaneous period begins if MFAN-N does not transmit any response packets for a certain period of time, and this period is maintained until MFAN-C transmits a RR packet. In the spontaneous period, MFAN-N can transmit data without the request of MFAN-C.

5.3 Physical element

The physical element configuring MFAN is divided into MFAN-C and MFAN-N in which all MFAN-Ns are connected into MFAN-C (i.e. a central connectivity device). The basic element, node, is distinguished into MFAN-C and MFAN-N according to its role. MFAN-C manages the whole MFAN and there must exist only one MFAN-C per one network. MFAN-C manages MFAN-N by sending the RR packet. MFAN-N must transmit response packets according to MFAN-C's management. MFAN can be configured as shown in Figure 7.

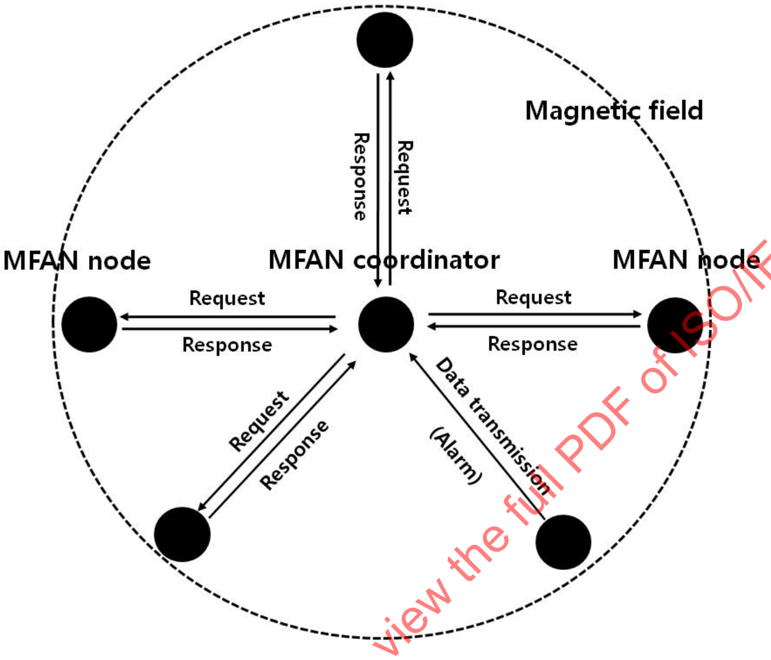


Figure 7 – MFAN

5.3.1 MFAN-C

MFAN-C is a node that manages MFAN; only one MFAN-C exists per one network, and it manages and controls MFAN-N by the RR packet.

5.3.2 MFAN-N

MFAN-N is a node that resides within an MFAN (excluding MFAN-C), and a maximum of 65,519 MFAN-Ns can exist per network. It transmits response packets according to the RR packet transmitted by MFAN-C.

5.4 Address element

In order to identify MFAN-Ns, MFAN uses address systems such as MFAN ID, UID, group ID and node ID.

5.4.1 MFAN ID

MFAN has its own ID that identifies each network from the others; the value should not be duplicated in other MFANs, and the value is maintained as long as MFAN exists. Its value is defined by user to distinguish networks.

5.4.2 UID

UID is a unique identifier consisting of 64 bits; it consists of group ID, IC manufacturer's code, and IC manufacturer's serial number. MFAN-N is identified by UID.

Unit: Byte

1	1	6
Group ID	IC manufacturer's code	IC manufacturer's serial number

Figure 8 – UID structure

5.4.3 Group ID

MFAN-N can be grouped by applications. Group ID is the identifier for the grouped MFAN-Ns within the network. MFAN-C can request a response to a specific MFAN-N group in order to mitigate the packet collision. Some group IDs are reserved in Table 1. Its value is defined by user to distinguish groups.

Table 1 – Reserved group ID

Group ID	Content	Remarks
0xFF	All groups	When selecting all groups
0xF0 – 0xFE	Reserved	-

5.4.4 Node ID

Node ID is an identifier used instead of UID to identify nodes, and it has a 16 bit address assigned by MFAN-C. Some node IDs are reserved in Table 2.

Table 2 – Reserved node ID

Node ID	Content	Remarks
0xFFFF	All nodes	When broadcasting or transmitting all nodes
0xFFFE	Unjoined node	Default ID for MFAN-N
0xFFF0 – 0xFFFD	Reserved	-

6 Network status

6.1 General

In an MFAN, MFAN-N may enter the active states of network configuration, network association, response transmission, data transmission, network disassociation, and network release.

6.2 Network configuration

MFAN-C configures a network by transmitting a request packet to MFAN-N in the request period. MFAN ID is included in the request packet so that MFAN-N can identify the connecting network. The minimum period of network means when only MFAN-C exists, and it consists of only the request period and the spontaneous period.

6.3 Network association

When MFAN-C sends the ARq packet in the request period, MFAN-N probes the received packet and then if it is the ARq packet for the desired MFAN, MFAN-N sends the ARs packet to the MFAN-C in the response period. MFAN-C, having received the ARs packet, transmits the ARA packet to MFAN-N. The network association of MFAN-N is completed upon receiving the ARA packet from MFAN-C.

6.4 Network disassociation

MFAN-N, associated with MFAN, can be disassociated either by MFAN-C’s request or by itself. MFAN-C can send the DaRq packet to MFAN-N according to the current network status for a forced disassociation. In the case of spontaneous disassociation due to shutting down and going out of the network coverage, MFAN-C can know the association status of MFAN-N by the response of ASRq from MFAN-C.

6.5 Data transmission

When MFAN-C sends the DRq packet in the request period to MFAN-N, MFAN-N sends DRs packet to MFAN-C according to the requested data type. Upon receiving the DRs packet, MFAN-C sends the DRA packet to MFAN-N, and MFAN-N, having received the DRA packet, completes the data transmission.

6.6 Network release

MFAN release can be divided into normal release through the request of MFAN-Ns and abnormal release due to a sudden situation. Normal release refers to MFAN-C releasing the network by its own decision and by sending the DaRq packet to all MFAN-Ns. Abnormal network release refers to MFAN-C shutting down or going out of the network coverage.

6.7 MFAN device state

MFAN device state includes the MFAN-C state and the MFAN-N state. In detail, MFAN-C states are divided into the standby state, the packet analysis state, and the packet generation state whereas MFAN-N states are composed of the hibernation state, the activation state, the standby state, the packet analysis state, and the packet generation state.

6.7.1 MFAN-C state

The state of MFAN-C goes to the standby state when the power turns on. In the standby state, when the application system commands sending the RR packet or the superframe begins, the state of MFAN-C goes to the packet generation state and MFAN-C sends the RR packet to MFAN-Ns. And then the state of MFAN-C goes back to the standby state. If MFAN-C receives the packet (either response or data packet) from MFAN-Ns while doing the carrier detection in the standby state, the state of MFAN-C goes to the packet analysis state. If the destination ID of the received packet and the node ID of MFAN-C are the same, the state of MFAN-C goes to the packet generation state, and then MFAN-C generates the RA or DA packet and sends it to MFAN-N in the packet generation state. After that, the state of MFAN-C goes back to the standby state. On the other hand, if there are errors in the data packet, the state of MFAN-C goes back directly to the standby state. In the packet analysis state, when there are errors in the received response packet or destination ID of the received response packet and node ID of MFAN-C do not correspond, MFAN-C regenerates the RR packet in the packet generation state and retransmits it to MFAN-Ns, and then the state goes to the standby state. If these failures occur consecutively, the procedure of the packet analysis state is repeated as many times as needed (maximum N times). In (N+1)th procedure, the state of MFAN-C goes from the packet analysis state to the standby state. MFAN-C state diagram is as Figure 9.

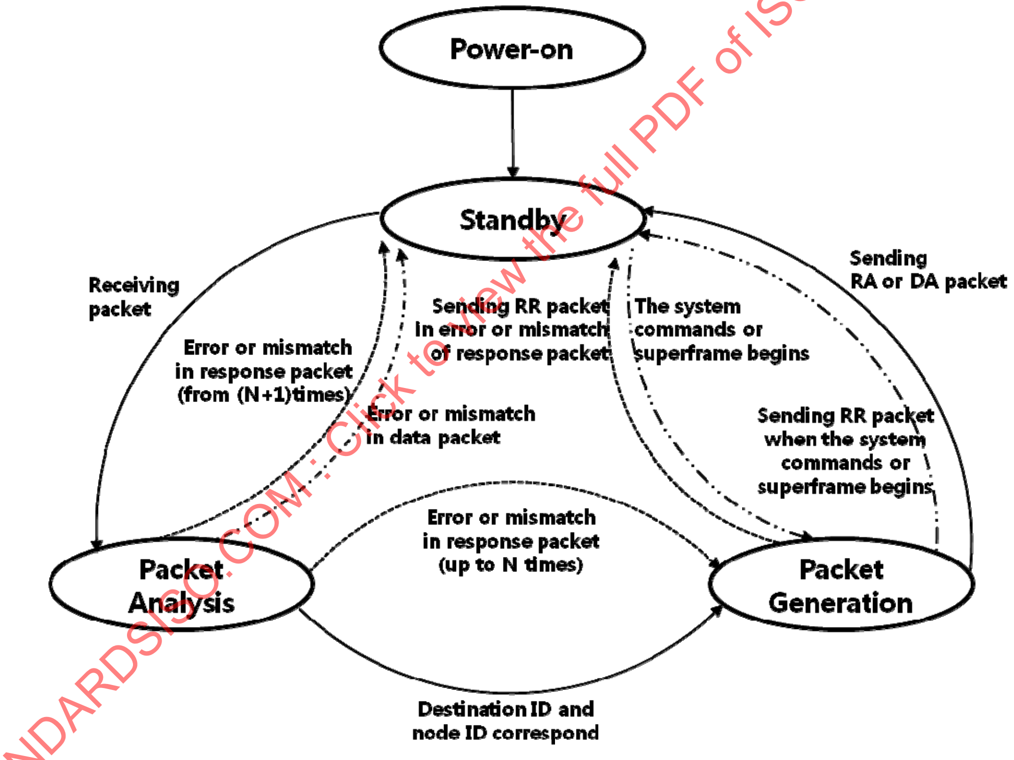

Power-on
Standby
Packet Analysis
Packet Generation
Receiving packet
Error or mismatch in response packet (from (N+1)times)
Sending RR packet In error or mismatch of response packet
The system commands or superframe begins
Sending RA or DA packet
Error or mismatch in data packet
Sending RR packet when the system commands or superframe begins
Error or mismatch in response packet (up to N times)
Destination ID and node ID correspond

Figure 9 – MFAN-C state diagram

6.7.2 MFAN-N state

The state of MFAN-N goes into the hibernation state when the power turns on. In the hibernation state, when the wake-up sequence is detected, the state goes into the activation state. The wake-up sequence is defined in 7.1. When MFAN-N receives the RR packet, the state of MFAN-N goes into the packet analysis state and MFAN-N analyzes the received RR packet. If the destination ID of the RR packet and MFAN-N ID (group ID and node ID) correspond, the state of MFAN-N goes into the packet generation state and MFAN-N sends the response packet to MFAN-C, and then the state of MFAN-N moves into the standby state. If not, the state goes back to the hibernation state.

While doing the carrier detection in the standby state, the state of MFAN-N goes to the hibernation state when MFAN-N receives the RA packet of its own node or to the packet generation state when MFNA-N receives the RA packet of other nodes. And the state of MFAN-N goes to the hibernation state when the slot-number is not allocated and the time-out period is over in the standby state or to the packet generation state when the slot-number is allocated and the time-out period is over (up to N times consecutively). However, the state goes to the hibernation state when the slot-number is allocated and the time-out period at N+1th is over. If the slot-number is allocated and MFAN-N does not receive RA packet during the time-out period, the state of MFAN-N goes from the standby state to the packet generation state. And then MFAN-N regenerates and retransmits the response packet to MFAN-C and the state of MFAN-N goes from the packet generation state to the standby state. The retransmission of the response packet is repeated as many times as needed (maximum N times). In the (N+1)th time-out period, the state goes from the standby state to the hibernation state. If MFAN-N receives the RR packet in the standby state while doing the carrier detection, the state is moved to the packet analysis state.

When the system interrupt occurs in the hibernation state, the state of MFAN-N is changed to the activation state. If MFAN-N receives data from the system, the state goes to the packet generation state. And then MFNA-N generates and sends data packet to the MFAN-C, and the state of MFAN-N goes to the standby state. If MFAN-N receives the DA packet, the state goes back to the hibernation state. If not, the state goes to the packet generation state and MFAN-N retransmits the data and then the state goes back to the standby state up to N times. MFAN-N state diagram is as Figure 10.

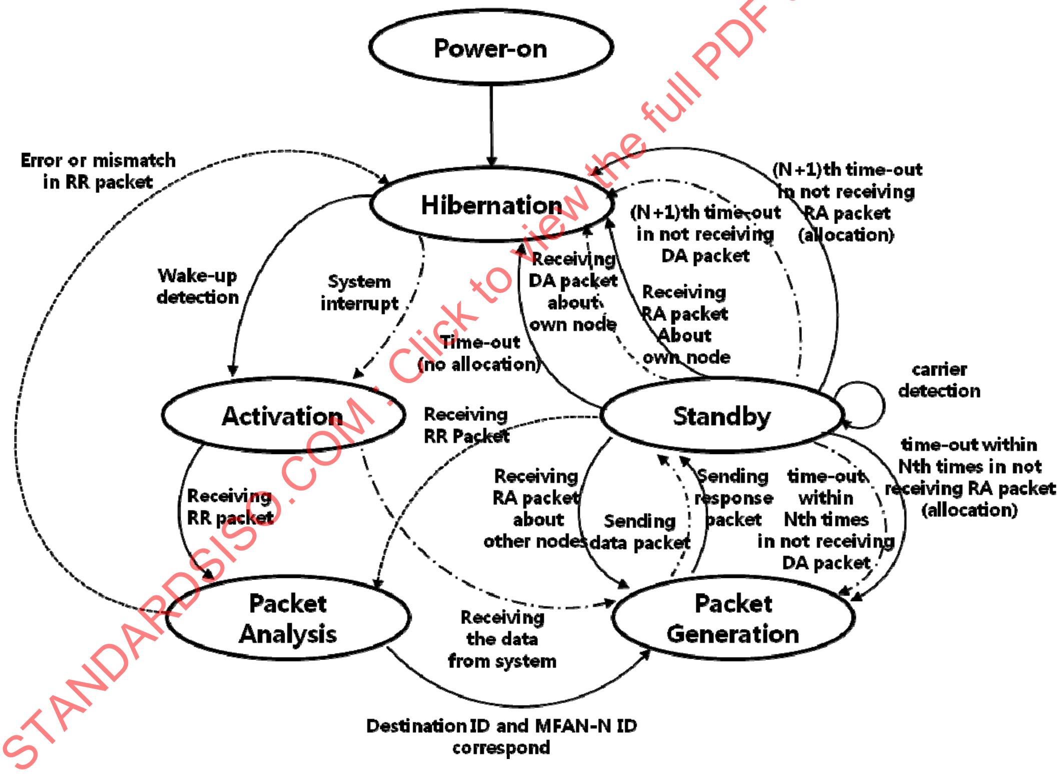

Power-on
Error or mismatch in RR packet
Hibernation
(N+1)th time-out in not receiving RA packet (allocation)
(N+1)th time-out in not receiving DA packet
Wake-up detection
System interrupt
Receiving DA packet about own node
Receiving RA packet About own node
Time-out (no allocation)
carrier detection
Activation
Receiving RR Packet
Standby
time-out within Nth times in not receiving RA packet (allocation)
Receiving RR packet
Receiving RA packet about other nodes
Sending data packet
Sending response packet
time-out within Nth times in not receiving DA packet
Packet Analysis
Receiving the data from system
Packet Generation
Destination ID and MFAN-N ID correspond

Figure 10 – MFAN-N state diagram

7 PHY layer

7.1 PHY layer frame format

7.1.1 General

This section describes the physical layer frame format. As shown in Figure 11, the PHY layer frame consists of three components: the preamble, the header, and the payload. When transmitting the packet, the preamble is sent first, followed by the header and finally by the payload. An LSB is the first bit transmitted.

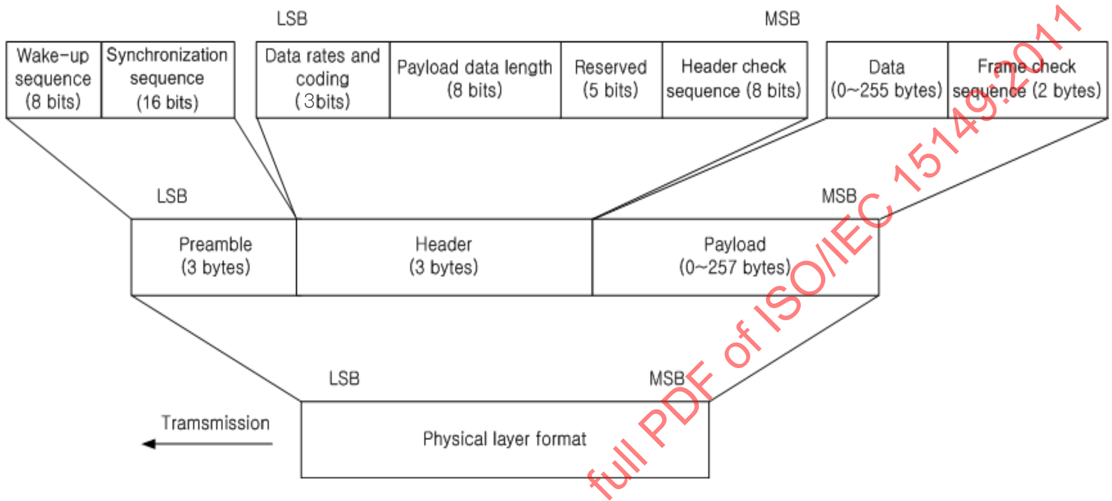

LSB
MSB
Wake-up sequence (8 bits)
Synchronization sequence (16 bits)
Data rates and coding (3bits)
Payload data length (8 bits)
Reserved (5 bits)
Header check sequence (8 bits)
Data (0~255 bytes)
Frame check sequence (2 bytes)
LSB
MSB
Preamble (3 bytes)
Header (3 bytes)
Payload (0~257 bytes)
LSB
MSB
Tramsmission
Physical layer format

Figure 11 – PHY layer frame format

7.1.2 Preamble

As shown in Figure 12, the preamble consists of two portions: a 8-bit wake-up sequence of [0000 0000] and a 16-bit synchronization sequence consisting of a 12-bit sequence of [000000000000] followed by a 4-bit sequence of [1010]. The wake-up sequence is only included in the preamble of RR packet in the request period. The synchronization sequence can be used for the packet acquisition, the symbol timing and the carrier frequency estimation.

The preamble is coded using the TYPE 0 defined in 7.1.3.1. The wake-up sequence is modulated by ASK, but the synchronization sequence by BPSK.

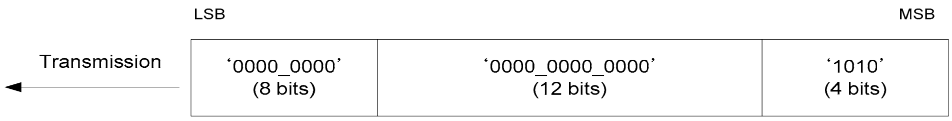
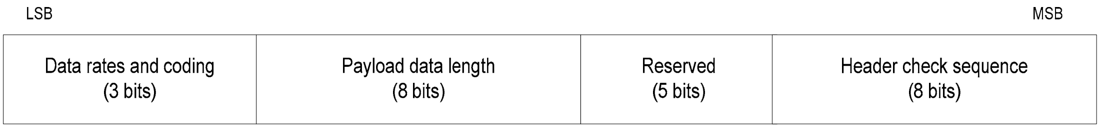


Figure 12 – Preamble format

7.1.3 Header

The header is added after the preamble to convey information about a payload. As shown in Figure 13, the header is composed of 24 bits. Bits 0-2 are the data rate and coding field. Bits 3-10 are the payload data length field. Bits 16-23 are a CRC-8 HCS. The details are defined in Table 3.

The header is coded using the TYPE 0 defined in 7.1.3.1.



LSB			MSB
Data rates and coding (3 bits)	Payload data length (8 bits)	Reserved (5 bits)	Header check sequence (8 bits)

Figure 13 – Header format

Table 3 – Header definition of physical layer

Bit	Content	Description
b2-b0	Data rate and coding	Specifies the data rate and coding at which the payload is received (see Table 4)
b10-b3	Payload data length	Specifies the number of octets in the payload (which does not include the FCS)
b15-b11	Reserved	Reserved and set to zero
b23-b16	HCS	Provides a CRC-8 HCS (see 7.1.3.3)

7.1.3.1 Data rate and coding

Depending on the data rate and coding used, bits 0-2 are set according to the values in Table 4. The details of TYPE 0~7 are described in 7.2.2.

Table 4 – Definition of the data rate and coding

Type	Value (b2 b1 b0)	Data rate	Coding method
TYPE 0	000	1 kbps	Manchester
TYPE 1	001	2 kbps	Manchester
TYPE 2	010	4 kbps	Manchester
TYPE 3	011	2 kbps	NRZ-L + Scrambling
TYPE 4	100	4 kbps	NRZ-L + Scrambling
TYPE 5	101	8 kbps	NRZ-L + Scrambling
TYPE 6	110	Reserved	-
TYPE 7	111	Reserved	-

7.1.3.2 Payload data length

The payload data length is an unsigned 8-bit integer that indicates the number of octets in the payload, which does not include the FCS. It ranges from 0x00 to a maximum of 0xFF bytes.

7.1.3.3 Header check sequence (HCS)

The header is checked for errors using a CRC-8 HCS. The HCS covers a data rate and coding, a payload data length and a reserved 5-bit. The primitive polynomial is given as,

$$g(D) = (1+D)(1+D^2+ D^3+ D^4+ D^7) = 1+ D+ D^2+ D^5+ D^7+ D^8$$

A schematic of the processing order is shown in Figure 14. The registers are initialized to all zeros.

Data is accumulated while the switch S in Figure 14 is being placed at ‘1’; when the last bit has been accumulated, the switch S goes to ‘2’ and HCS is transmitted from the register beginning with in D^7.

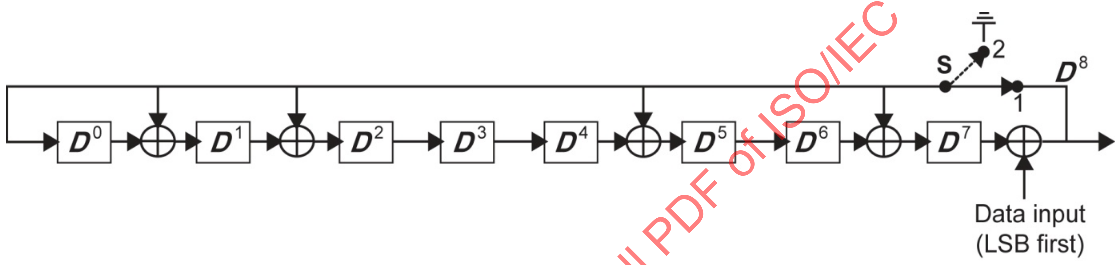


Figure 14 – Encoder of header check sequence

7.1.4 Payload

As shown in Figure 15, the payload consists of a variable length data and the FCS. If the payload data length field in the header has zero, FCS is not sent.

Data (0~255 byte)	Frame check sequence (2 byte)

Figure 15 – Format of payload

7.1.5 Frame check sequence (FCS)

The payload is checked for errors using a CRC-16 FCS defined in Table 5. The FCS covers a variable length data. The primitive polynomial is $X^{16}+X^{12}+X^5+1$. The registers are initialized to all ones. The frame check sequence is obtained by inverting the calculated CRC-16 bits.

Table 5 – Cyclic redundancy check for frame check sequence

CRC type	Length	Polynomial	Preset	Residue
ISO/IEC 13239	16 Bit	$X^{16}+X^{12}+X^5+1$	0xFFFF	0x1D0F

7.2 Coding and modulation

7.2.1 Coding

7.2.1.1 Manchester coding

Manchester code has a transition in the middle of every bit interval whether a one or a zero is being sent. A zero is represented by a half-bit-wide pulse positioned during the first half to the bit interval. A one is represented by a half-bit-wide pulse positioned during the second half to the bit interval.

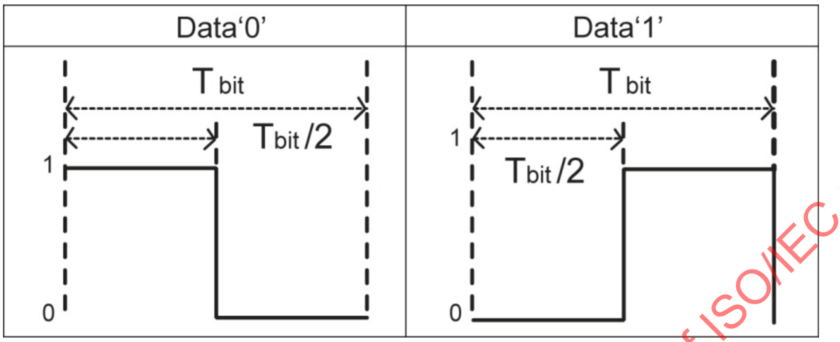


Figure 16 – Definition of Manchester coding

7.2.1.2 NRZ-L coding

With NRZ-L(Level) a zero is represented by zero level and a one is represented by one level.

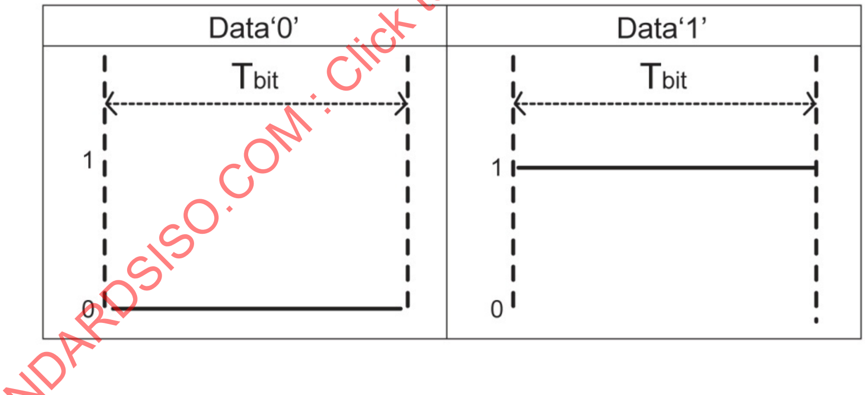


Figure 17 – Definition of NRZ-L coding

7.2.1.3 Scrambling

A scrambler is used to whiten only a payload data encoded by NRZ-L. The primitive polynomial g(D) is given as,

$$g(D) = 1+ D^{14}+ D^{15}$$

where D is a bit delay element. Figure 18 shows the scrambler's block diagram. d_k is generated as follows,

$$d_k = d_{k-14} \oplus d_{k-15}$$

where '$\oplus$' denotes modulo-2 addition. The scrambler is initialized to a seed value, 0xFFFF. The scrambled data bit, b_k, is obtained as:

$$b_k = s_k \oplus d_k$$

where s_k represents the non-scrambled data bit.

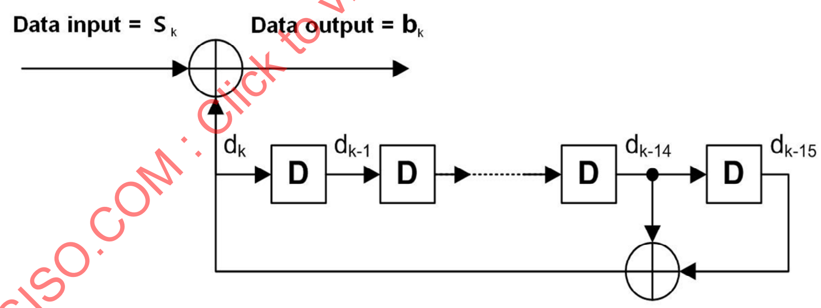


Figure 18 – Scrambler block diagram

7.2.2 The data rate and coding type

The physical layer supports 8 Types as shown in Table 4.

Preamble and header are encoded by TYPE 0 using Manchester coding at a data rate of 1 kbps, however payload is encoded using the appropriate data rate of 1, 2, 4, or 8 kbps and coding. The data rate and coding type of the payload are specified in the data rate and coding field in the header.

7.2.3 Modulation

The communications between MFAN-C and MFAN-N uses either ASK modulation or BPSK modulation.

7.2.3.1 ASK modulation

As shown in Figure 19, the encoded serial input data is converted into a number representing one of the two ASK constellation points. (ω_C is the carrier frequency of MFAN.)

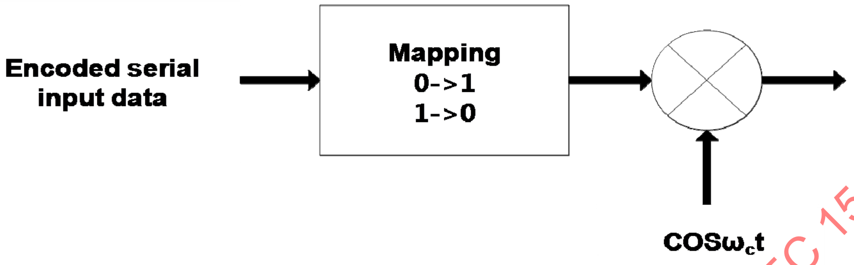


Figure 19 – ASK modulation diagram

7.2.3.2 BPSK modulation

The transmission between MFAN-C and MFAN-N uses the BPSK modulation. As shown in Figure 20, the encoded serial input data is converted into a number representing one of the two BPSK constellation points. (ω_C is the carrier frequency of MFAN.)

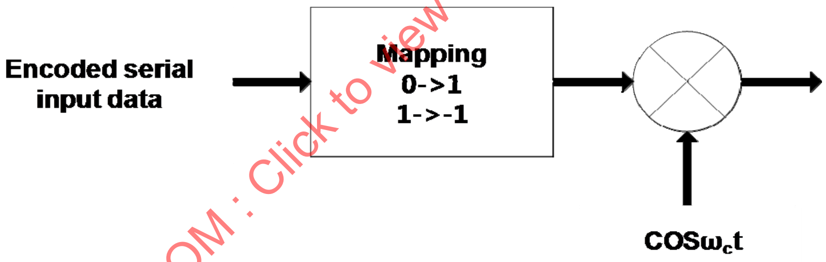


Figure 20 – BPSK modulation diagram

7.2.4 The coding and modulation process

7.2.4.1 The coding and modulating process of the preamble

The preamble sequence (see 7.1.2) is encoded using the TYPE 0 (see 7.2.2). The wake-up sequence, if any, is modulated by ASK and the synchronization sequence by BPSK.

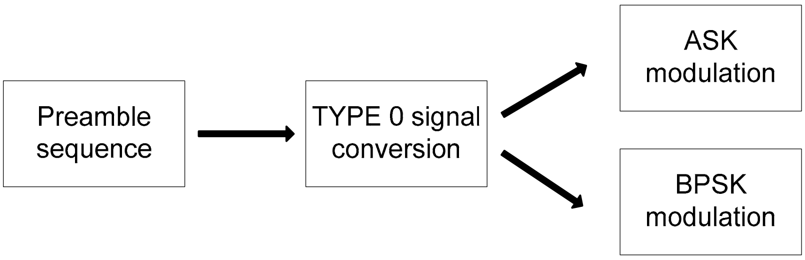


Figure 21 – The coding and modulation process of the preamble

7.2.4.2 The coding and modulation process of the header

As shown in Figure 22, header is formatted by appending a data rate and coding, a payload data length, a 5-bit zero (see 7.1.3) and HCS value. The resulting combination is encoded using the TYPE 0 (see 7.2.2) and then modulated by BPSK.

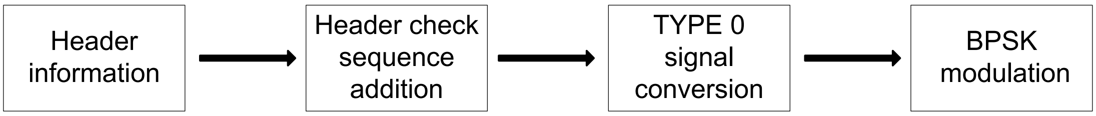


Figure 22 – The coding and modulation process of header

7.2.4.3 The coding and modulation process of the payload

As shown in Figure 23, payload is formatted by appending data and FCS value, which is calculated over the data. The resulting combination is encoded using the TYPE I (I = 0~7) (see 7.2.2) and then modulated by BPSK.

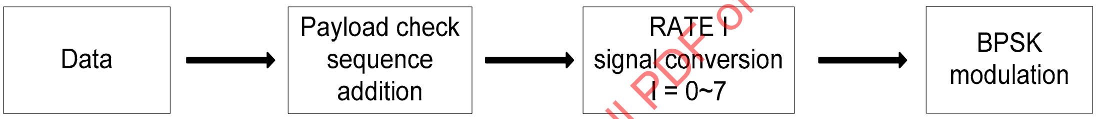

Data
Payload check sequence addition
RATE I signal conversion I = 0~7
BPSK modulation

Figure 23 – The coding and modulation process of payload

8 MAC layer frame format

8.1 General

The MAC frame of MFAN consists of the frame header and the frame body. The frame header has information for data among MFAN-Ns, and the frame body has the data for transmissions between MFAN devices.

8.2 Frame format

All frame of MAC consists of the frame header and the frame body as shown in Figure 24.

1) Frame header: Consists of the MFAN ID, frame control, source node ID, destination node ID, and sequence number. The frame header can be used for the data transmission.
2) Frame body: Consists of the payload that contains the data for transmissions between MFAN devices and the FCS used to check errors within the payload.

Unit: Byte

1	2	2	2	1	Variable	2
MFAN ID	Frame control	Source node ID	Destination node ID	Sequence number	Payload	Frame check sequence
Frame header					Frame body	

Figure 24 – MAC layer frame format

8.2.1 Frame header

Frame header has information for the transmission/reception of frames and flow control.

8.2.1.1 MFAN ID

As shown in Figure 24, MFAN ID field consists of 1 byte and is used to identify networks.

8.2.1.2 Frame control

Frame control fields consist of the frame type, the acknowledgement policy, the first fragment, the last fragment, and the protocol version; its format is shown in Figure 25.

Unit: Bit

0-2	3-4	5	6	7-8	9-15
Frame type	Acknowledgement policy	First fragment	Last fragment	Protocol version	Reserved

Figure 25 – Format of frame control field

Each field is explained as follows:

1) Frame type field consists of 3 bits; refer to 8.3 for the details on frame types.

2) Acknowledgement policy field consists of 2 bits; in the case where the received frame is an acknowledgement frame, it indicates the policy of the received acknowledgement frame, otherwise it indicates the policy of the acknowledgement frame for a destination node. The following shows the acknowledgement policy:

   a) No acknowledgement: Destination node does not acknowledge the transmitted frame, and the source node considers the transmission successful regardless of the transmission result. Such method can be used in the frame that is transmitted for 1:1 or 1:N, transmission which do not required the acknowledgement.

   b) Single acknowledgement: The destination node that received the frame sends an acknowledgement frame as a response to the source node after an SIFS. This acknowledgement policy can only be used for 1:1 transmission.

   c) Multiple acknowledgement: The destination node that received the frames sends an acknowledgement frame as a response to the multiple source nodes after an SIFS. This acknowledgement policy can be used for 1:N transmission.

   d) Data acknowledgement: The destination node that received the data frame sends data acknowledgement frame as a response to the source node after an SIFS. This acknowledgement policy can only be used for 1:1 data transmission.

3) First fragment field is 1 bit; ‘1’ indicates that frame is the start of the request, response or data packet from a higher layer, while ‘0’ means that it is not the start.

4) End fragment field is 1 bit; ‘1’ indicates that frame is the end of the request, response or data packet from a higher layer, while ‘0’ means that it is not the end.

5) Protocol version field consists of 2 bits, and the size and location are fixed regardless of the protocol version of the system. The present value is 0, and increases by 1 each time a new version is released. When a node receives a packet with a version higher than its own, it discards it without notifying the source node.

6) Reserved: a field reserved for the future use.

8.2.1.3 Sequence number

The sequence number field has 8 bits of length, and indicates the frame sequence number. In data frames, a sequence number between 0 and 255 is assigned by means of an incremental counter for each packet, and once it reaches 255 it wraps back to 0.

8.2.2 Frame body

Frame body has a variable length and consists of the payload and the FCS. Each payload has a different format according to the frame type in the frame control field, and the FCS is used to check for error in the frame.

8.2.2.1 Payload

Payload has the transmission data between MFAN-C and each MFAN-N, and the length has a variable value between 0 and 247.

8.2.2.2 Frame check sequence

FCS is 16 bits in length, and is used to verify whether the frame body was received without error. It is generated by using the following 16th standard generator polynomial:

$$G(x) = x^{16} + x^{12} + x^{5} + 1$$

8.3 Frame type

The frame type is defined as 4 kinds of types, the request frame, the response frame, the data frame, and the n acknowledgement frame.

Table 6 – Frame type value

Frame type	**Value (Binary)**	**Content**	**Period**
Request frame	000	Request for the response of association (ARq), disassociation (DaRq), association status (ASRq), data transmission (DRq), and so on.	Request
Response frame	001	Response for the request of association (ARs), disassociation (DaRs), association status (ASRs), data transmission (DRs), and so on.	Response
Data frame	010	Data transmission without the request of MFAN-C	Spontaneous
Acknowledgement frame	011	Acknowledgement of the response (RA*) and data transmission (DA) for MFAN-N	Response, Spontaneous

* RA includes ARA, DaRA, ASRA, and so on.

8.3.1 Request frame

The request frame is used when MFAN-C sends a RR packet in the request period to a certain MFAN-N in MFAN, or broadcasts information to all MFAN-Ns. The request frame format is shown in Figure 26. Note that the acknowledgement policy, when broadcasting the RR packet, is no acknowledgement. And the RR packet includes ARq, DaRq, ARsRq, DRq, and so on.

Unit: Byte

<table>
<tr><td><b>8</b></td><td><b>1</b></td><td><b>1</b></td><td><b>1</b></td><td><b>L1</b></td><td><b>L2</b></td><td><b>…</b></td><td><b>Ln</b></td><td><b>2</b></td></tr>
<tr><td rowspan="3">Frame header</td><td>Group ID</td><td>Request code</td><td>Length (=$\sum L_n$)</td><td>Request block-1</td><td>Request block-2</td><td>…</td><td>Request block-n</td><td rowspan="2">Frame check sequence</td></tr>
<tr><td colspan="7">Request frame payload</td></tr>
<tr><td colspan="8">Frame body</td></tr>
</table>

Figure 26 – Request frame format

8.3.2 Response frame

The response frame is used when MFAN-Ns send the response packet for the request of MFAN-C in the response period. The MFAN-N sends the response packet within a certain number of times in the response period until an acknowledgement packet is received.

Unit: Byte

8	1	1	1	L1	L2	…	Ln	2
Frame header	Group ID	Response code	Length $(=\sum L_n)$	Response block-1	Response block-2	…	Response block-n	Frame check sequence
	Response frame payload							
	Frame body							

Figure 27 – Response frame format

8.3.3 Data frame

The data frame is used when MFAN-N transmits data to MFAN-C in the spontaneous period without MFAN-C' s request.

Unit: Byte

8	8	L	2
Frame header	UID	Data	Frame check sequence
	Data frame payload		
	Frame body		

Figure 28 – Data frame format

8.3.4 Acknowledgement frame

The acknowledgement frame includes the RA frame and the DA frame. In the case that MFAN-C transmits a RR packet, MFAN-N receiving the RR packet transmits the response packet and MFAN-C receiving the response packet transmits the RA packet. The payload of the acknowledgement frame includes the response confirmation data about the received response packet. MFAN-C, having received the response packet, answers MFAN-N by sending the RA packet after the SIFS in the response period. DA frame is the acknowledgement frame about the received data packet. MFAN-C answers MFAN-N that has transmitted the data packet by sending the DA packet after an SIFS in the spontaneous period.

Unit: Byte

8	1	1	1	L1	L2	…	Ln	2
Frame header	Group ID	Response confirmation code	Length $(=\sum L_n)$	Response confirmation block-1	Response confirmation block-2	…	Response confirmarion block-n	Frame check sequence
	Acknowledgement frame payload							
	Frame body							

Figure 29 – Frame format of response acknowledgement

As shown in Figure 30, the DA frame consists of the frame header and the frame body. When the destination node ID is 0xFFFE which is un-joined node ID, the UID field has been included.

Unit: Byte

1	2	2	2	1	8(option)	2
MFAN ID	Frame control	Source node ID	Destination node ID	Sequence number	UID	Frame check sequence
Frame header					Frame body	

Figure 30 – Frame format of data acknowledgement

8.4 Payload format

The payload format is composed of the request frame, response frame, data frame, and acknowledgement frame.

8.4.1 Request frame

As shown in Figure 31, the payload for the request frame consists of the group ID, the request code, the length, and more than one request block. When the group ID is 0xFF, it indicates that MFAN-C requests a response to all MFAN-N groups.

Unit: Byte

1	1	1	L1	L2	…..	Ln
Group ID	Request code	Length $(=\sum L_n)$	Request block-1	Request block-2	….	Request block-n

Figure 31 – Payload format of request frame

8.4.1.1 Group ID

The group ID field consists of 1 byte and is used to send RR packets to certain groups. For the details of the group ID, refer to 5.4.3.

8.4.1.2 Request code

The request code in the payload of a request frame is shown in Table 7.

Table 7 – Payload request code of request frame

Category	Request code	Content	Remarks
Network	0x01	Association request	Association response request to unjoined node
	0x02	Disassociation request	Disassociation response request to joined node
	0x03	Association status request	Association status response request to joined node
	0x04 – 0x0F	Reserved	-
Data	0x11	Data request	Data transmission request to joined node
	0x12 – 0x1F	Reserved	-
Configuration	0x21	Group ID set-up	Group ID change request to joined node
	0x22 – 0x2F	Reserved	-
Reserved	0x31 – 0xFF	Reserved	-

8.4.1.3 Length

The length field consists of 1 byte; it indicates the total length of the request block, and the length field value is variable according to the length and number of the request block.

8.4.1.4 Request block

The data format of the request block is composed differently according to the request code, and more than one request block can be included the payload of the request frame.

The details for the data format of each request block is as follows:

1) Association request

The block format of the ARq is shown in Figure 32 and consists of 8 bytes UID mask. This UID mask can be used to implement a binary search algorithm.

Unit: Byte

8
UID mask

Figure 32 – Block format of association request

2) Disassociation request

The block format of the DaRq is shown in Figure 33. The first 2 bytes are the node ID of the MFAN-N for the DaRq, and the next 1 byte is the slot number to be used in the response period. If the node ID is 0xFFFF, the DaRq is sent to all the MFAN-Ns under the group ID.

Unit: Byte

2	**1**
Node ID	Slot number

Figure 33– Block format of disassociation request

3) Association status request

The block format of the ASRq is shown in Figure 34. The first 2 bytes are the node ID of the MFAN-N for the ASRq. If the node ID is 0xFFFF, the ASRq is requested to all MFAN-Ns under the group ID.

Unit: Byte

2	**1**
Node ID	Slot number

Figure 34 – Block format of association status request

4) Data request

The block format of the DRq is shown in Figure 35. The first 2 bytes are the node ID, the next 1 byte is the slot number, and the last L bytes are the received data type. The data type is determined according to the application product.

Unit: Byte

2	**1**	**L**
Node ID	Slot number	Data

Figure 35 – Block format of data request

5) Group ID set-up request

The block format of the GSRq is shown in Figure 36. The first 2 bytes are the node ID, the next 1 byte is the slot number, and the last byte is the group ID to be set up.

Unit: Byte

2	**1**	**1**
Node ID	Slot number	Group ID

Figure 36 – Block format of group ID set-up request

8.4.2 Response frame

The payload format of the response frame has the response information about the request of MFAN-C. The response frame payload is shown in Figure 37. The first byte is the group ID, the second byte is the response code, the third byte is the response date length (L), and the next L bytes are the response data.

Unit: Byte

1	1	1	L1	L2	…	Ln
Group ID	Response code	Length(=L)	Response block-1	Response block-2	…	Response block-n

Figure 37 – Payload format of response frame

8.4.2.1 Group ID

The group address field consists of 1 byte and is used to send RR packets to certain groups. For the details of the group ID, refer to 5.4.3.

8.4.2.2 Response code

Response code types are shown in Table 8.

Table 8 – Response code of response frame payload

Category	Response code	Content	Remarks
Network	0x01	Association response	UID transmission of MFAN-N
	0x02	Disassociation response	UID transmission of MFAN-N
	0x03	Association status response	UID transmission of MFAN-N
	0x04 – 0x0F	Reserved	-
Data	0x11	Data response	Requested data transmission
	0x12 – 0x1F	Reserved	-
Set-up	0x21	Group ID set-up response	UID and group ID transmission after group ID changes
	0x22 – 0x2F	Reserved	-
Reserved	0x31 – 0xFF	Reserved	-

8.4.2.3 Length

The length field consists of 1 byte and indicates the length of the response data; it is variable according to the response data.

8.4.2.4 Response data

Response data are divided into ARs, DaRs, ASRs, DRs, and GSRs. The response data format is as follows:

1) Association response

The block format of the ARs is shown in Figure 38. The ARs data consists of 8 bytes UID.

Unit: Byte

8
UID

Figure 38 – Block format of association response

2) Disassociation response

The block format of the DaRs is shown in Figure 39. The DaRs data consists of 8 bytes UID.

Unit: Byte

8
UID

Figure 39 – Block format of disassociation response

3) Association status response

The block format of the ASRs is shown in Figure 40. The ASRs data consist of 8 bytes UID and 1 byte of the status value.

Unit: Byte

8	1
UID	Status value

Figure 40 – Block format of association status response

The status value is as shown in Table 9.

Table 9 – Association status check value

Value	Content
0x00	Disassociation status
0x01	Association status
0x02 – 0xFF	Reserved

4) Data response

The block format of the DRs is shown in Figure 41. The data of DRs consists of L bytes of requested data.

Unit: Byte

L
Requested data

Figure 41 – Block format of data response

5) Group ID set-up response

The block format of the GSRs is shown in Figure 42. The GSRs data consist of 8 bytes for UID with the changed group ID and 1 byte for the changed group ID.

Unit: Byte

8	**1**
UID	Assigned group ID

Figure 42 – Block format of group ID set-up response

8.4.3 Data frame

The data frame payload includes the data to be transmitted. The data frame payload consists of 8 bytes of UID and L bytes of data as shown in Figure 43.

Unit: Byte

8	**L**
UID	Data

Figure 43 – Payload format of data frame

8.4.4 Acknowledgement frame

The RA frame payload has the data about the received response packet. The RA payload format is shown in Figure 44. The first byte is the group ID, the second byte is the response confirmation code, the third byte is the length (L), and the next L bytes are the response confirmation blocks.

Unit: Byte

1	**1**	**1**	**L1**	**L2**	**.....**	**Ln**
Group ID	Response confirmation code	Length (=L)	Response confirmation block-1	Response confirmation block-2	….	Response confirmation block-n

Figure 44 – Payload format of acknowledgement frame

8.4.4.1 Group ID

The group ID field consists of 1 byte and is used to send RR packets to a certain groups. The detail of the group ID, refer to 5.4.3.

8.4.4.2 Response confirmation code

Response confirmation code types are shown in Table 10.

Table 10 – Response confirmation code

Category	Reception confirmation code	Content	Remarks
Network	0x01	Association response confirmation	UID and assigned node ID transmission of MFAN-N
	0x02	Disassociation response confirmation	UID and node ID transmission of MFAN-N
	0x03	Association status response confirmation	UID transmission of MFAN-N
	0x04 – 0x0F	Reserved	-
Data	0x11	Data response confirmation	Data transmission confirmation to a joined node
	0x12 – 0x1F	Reserved	-
Set-up	0x21	Group ID set-up response confirmation	UID and group ID transmission after group ID changes
	0x22 – 0x2F	Reserved	-
Reserved	0x31 – 0xFF	Reserved	-

8.4.4.3 Length

The length field consists of 1 byte; it indicates the length of response confirmation data and is variable according to the response confirmation data.

8.4.4.4 Response confirmation block

Response confirmation block are divided into ARs confirmation, DaRs confirmation, ASRs confirmation, DRs confirmation, and GSRs confirmation. The block format of the response confirmation is as follows:

1) Association response confirmation

The block format of the ARs confirmation is shown in Figure 45. The first 8 bytes are the UID, the next 2 bytes are the assigned node ID. If the assigned node ID is 0xFFFE which is the address of un-joined node, it means the ARq has been rejected.

Unit: Byte

8	2
UID	Assigned node ID

Figure 45 – Block format of association response confirmation

2) Disassociation response confirmation

The block format of the DaRs confirmation is shown in Figure 46. The first 8 bytes are the UID, and the next 2 bytes are the node ID. The assigned node ID is used if disassociation is not allowed, the unjoined node ID, 0xFFFE is recorded if disassociation is allowed.

Unit: Byte

8	2
UID	Node ID

Figure 46 – Block format of disassociation response confirmation

3) Association status response confirmation

The block format of the ASRs confirmation is shown in Figure 47. The ASRs confirmation block consists of the 8 bytes UID.

Unit: Byte

8
UID

Figure 47 – Block format of association status response confirmation

4) Data response confirmation

The block format of the DRs confirmation is shown in Figure 48. The first 2 bytes are the Node ID, and the next 1 byte is the assigned reserved.

Unit: Byte

2	1
Node ID	Reserved

Figure 48 – Block format of data response confirmation

5) Group ID set-up response confirmation

The block format of GSRs confirmation is shown in Figure 49. The GSRs confirmation block consists of 8 bytes of UID and 1 byte of status check value.

Unit: Byte

8	1
UID	Group ID set-up status value

Figure 49 – Block format of group ID set-up response confirmation

The group ID set-up status value is shown in Table 11.

Table 11 – Group ID set-up status value

Value	Content
0x00	Change completed
0x01	Change failed
0x02 – 0xFF	Reserved

9 MAC layer function

9.1 General

In the MAC layer of MFAN, in order to manage MFAN, the association, disassociation, and ASC process for MFAN-Ns are considered. Data can be transmitted either in the response period or in the spontaneous period. In addition, the group ID set-up function is provided for the management of MFAN-N groups.

9.2 Network association and disassociation

In order for MFAN-N to communicate with MFAN-C, it first needs to be in association with MFAN. As a given, each MFAN-N looks for a pre-configured MFAN; when it finds one, it associates with that MFAN, and when it does not find, any MFAN-N can become MFAN-C by user of the application(i.e. configuration of a new MFAN which means the new MFAN-C sends the request packet periodically). However, nodes can maintain their status as MFAN-C or MFAN-N, whose status is set according to the role of nodes since MFAN is configured. In this case, if there exists MFAN that has been configured already, network configuration is cancelled because there is only one available channel.

9.2.1 Association

When MFAN-C in the request period sends the ARq packet to the unjoined MFAN-N. MFAN-N transmits the ARs packet to MFAN-C in the response period. MFAN-C decides whether MFAN-N associates in MFAN or not, and notifies this results through the ARA packet. When the association has been allowed, the assigned node ID is included in the ARA packet, and when it has been rejected, the unjoined node ID 0xFFFE is recorded. When MFAN-C does not receive the ARs packet or MFAN-N does not receive the ARA packet due to the ARA packet error, it sends the ARq packet continuously every superframe until it receives the ARA packet of all selected MFAN-N without error. The procedure of association for MFAN-N is done when MFAN-N receives the ARA packet from MFAN-C.

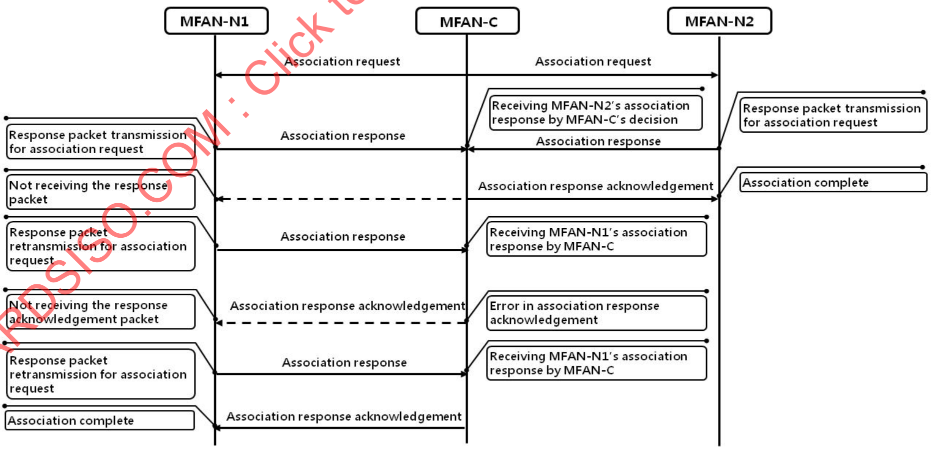

MFAN-N1
MFAN-C
MFAN-N2
Association request
Association request
Receiving MFAN-N2's association response by MFAN-C's decision
Response packet transmission for association request
Response packet transmission for association request
Association response
Association response
Not receiving the response packet
Association response acknowledgement
Association complete
Response packet retransmission for association request
Association response
Receiving MFAN-N1's association response by MFAN-C
Not receiving the response acknowledgement packet
Association response acknowledgement
Error in association response acknowledgement
Response packet retransmission for association request
Association response
Receiving MFAN-N1's association response by MFAN-C
Association complete
Association response acknowledgement

Figure 50 – Association procedure

9.2.2 Disassociation

When MFAN-C in the request period sends the DaRq packet to MFAN-N associated with MFAN, MFAN-N transmits the DaRs packet to MFAN-C in the response period. MFAN-C decides whether MFAN-N disassociates in MFAN or not, and notifies this results through the DaRA packet. When the disassociation has been allowed, the node ID on the DaRA packet is recorded as the unjoined node ID 0xFFFE, and when the disassociation has been rejected, the assigned node ID is recorded. When MFAN-C does not receive the DaRs packet or MFAN-N does not receive the DaRA packet due to the DaRA packet error, the MFAN-N retransmits the DaRs packet continuously every superframe until the MFAN-N receives the DaRA packet. Disassociation is complete when MFAN-N receives the DaRA packet from MFAN-C.

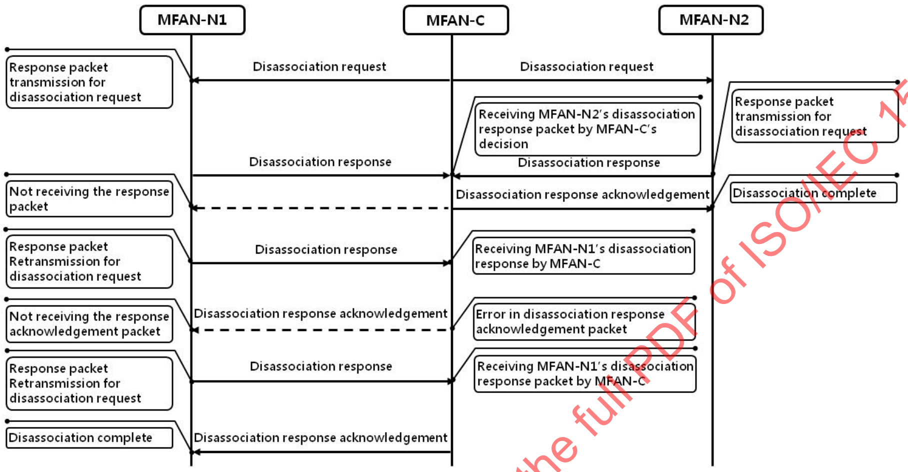

MFAN-N1
MFAN-C
MFAN-N2
Response packet transmission for disassociation request
Disassociation request
Disassociation request
Response packet transmission for disassociation request
Receiving MFAN-N2's disassociation response packet by MFAN-C's decision
Disassociation response
Disassociation response
Not receiving the response packet
Disassociation response acknowledgement
Disassociation complete
Response packet Retransmission for disassociation request
Disassociation response
Receiving MFAN-N1's disassociation response by MFAN-C
Not receiving the response acknowledgement packet
Disassociation response acknowledgement
Error in disassociation response acknowledgement packet
Response packet Retransmission for disassociation request
Disassociation response
Receiving MFAN-N1's disassociation response packet by MFAN-C
Disassociation complete
Disassociation response acknowledgement

Figure 51 – Disassociation procedure

9.2.3 Association status check

When MFAN-C in the request period sends the ASRq packet to the associated MFAN-N, MFAN-N transmits the ASRs packet to MFAN-C in the response period. MFAN-C checks and transmits the ASRA packet for the MFAN-Ns association status to MFAN. When MFAN-C does not receive the ASRs packet or MFAN-N cannot receive the ASRA packet due to the packet error, the MFAN-N transmits the ASRs packet continuously every time-slot until it receives the ASRA packet. The procedure of association status confirmation for MFAN-N's is complete when MFAN-N receives the ASRA packet from MFAN-C.

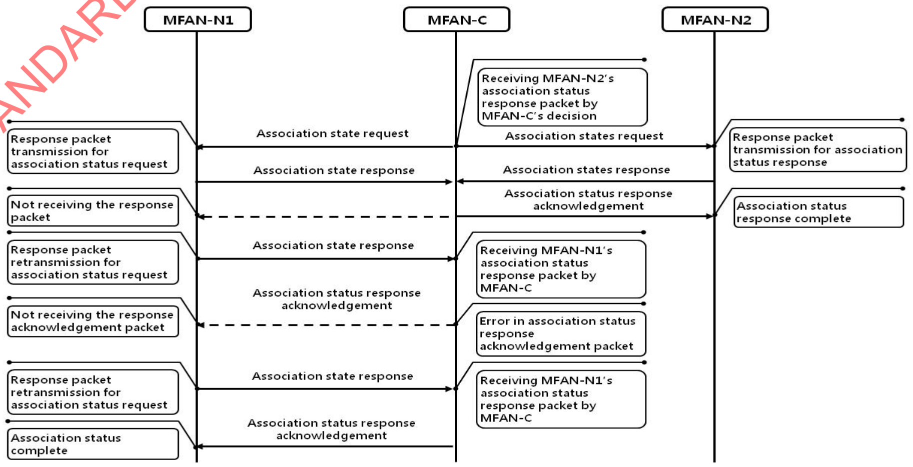

MFAN-N1
MFAN-C
MFAN-N2
Receiving MFAN-N2's association status response packet by MFAN-C's decision
Response packet transmission for association status request
Association state request
Association states request
Response packet transmission for association status response
Association state response
Association states response
Not receiving the response packet
Association status response acknowledgement
Association status response complete
Response packet retransmission for association status request
Association state response
Receiving MFAN-N1's association status response packet by MFAN-C
Not receiving the response acknowledgement packet
Association status response acknowledgement
Error in association status response acknowledgement packet
Response packet retransmission for association status request
Association state response
Receiving MFAN-N1's association status response packet by MFAN-C
Association status complete
Association status response acknowledgement

Figure 52 – Procedure of association status confirmation