Part III

Transmission Media and Lower-Layer Communication Software Specifications

Revision History Note) Version numbers except Ver.3.20 indicate Japanese editions.

・Version 1.0	March 18th	2000	Released / Open to consortium members
	July	2000	Open to the public
・Version 1.01	May 23rd	2001	Open to consortium members
			Version 1.0 addendum & corrigendum
・Version 2.00	August 7th	2001	Open to consortium members

Since the power line A and power line B methods were integrated into a single power line method (based on the power line A method), the associated descriptions were corrected accordingly.
The following table-of-contents entries were revised:

	Revised entry	Revision/addition
1	1.2, 1.3	Descriptions were deleted or corrected because the power line A and power line B methods were integrated into a single method.
2	Chapter 2	The power line communication protocol specification was renamed because the power line A and power line B methods were integrated into a single method.
3	Chapters 3 to 6	Chapter, figure, and table numbers were changed accordingly because the power line A and power line B methods were integrated into a single method.

・Version 2.01 November 9th 2001
Descriptions were corrected as needed for typographical correction, terminological standardization, and other purposes.

・Version 2.01	December 19th	2001	Open to consortium members.
・Version 2.10 Preview	December 28th	2001	Open to consortium members.
・Version 2.10 Draft	February 15th	2002	Open to consortium members.

Part 3 revisions related to the state transition related descriptions in Sections 2.5, 3.5, 5.5, and 6.5.

・Version 2.10	March 7th	2002	Open to consortium members.
・Version 2.11	April 26th	2002	Open to consortium members

The following Table-of-Contents entries were revised:

	Revised entry	Revision/addition
1	2.5, 3, 5, 4.7, 5.5, 6.5	- The typographical error “LowReset” for “LowStart” in relation to status change was corrected, and the status acquisition service return value for each was corrected.
2	2.5, 6.5	- The typographical error for “lower-layer communication software type acquisition service (LowGetDevID)” was corrected.
3	Chapters 2 through 6	- Correction of other typos.

・Version 3.00 June 12th 2002 Open to consortium members.

Chapter 1: Descriptions about IP/Bluetooth and IP/Ethernet/IEEE802.3 added.

Chapter 7: IP/Bluetooth communication protocol specifications added.

Chapter 8:IP/Ethernet/IEEE802.3 communication protocol specifications added.

The following table-of-contents entries were revised:

	Changed part (section number in the Table of Contents)	<Summary of additions/changes>
1	7.7.2	* Amendments to packet formats and the addition of new packet formats. * Amendments to formats for ECHONET frame transfer packets etc. * Addition of a new packet type (network control message packets).
2	7.7.6	* Addition of specific values to the time requirements.
3	7.9	* Addition of equipment types including "(5) Address Server Functions".
4	3.5.5.5	* Addition of the lower-layer communication software type acquisition service (LowGetDevID).
5	Chapter 2 to Chapter 8	* Correction of incorrect descriptions and the addition of descriptions.

・Version 3.10 December 18th 2002 Open to consortium members.

The following table-of-contents entries were revised:

	Changed part (section number in the Table of Contents)	<Summary of additions/changes>
1	7.6	* Amendments to port number requirements.
2	7.7	* Amendments to descriptions about the packets required for address servers.
3	7.9, 8.9	* Additions and amendments to descriptions about address servers.

・Version 3.11 March 7th 2003 Open to consortium members.

The following table-of-contents entries were revised:

	Changed part (section number in the Table of Contents)	<Summary of additions/changes>
1	7.6	* Amendments to multicast address number requirements.
2	7.7.2 (2)(c) and (3)(c)	* Correction of incorrect descriptions.

・Version 3.12 May 23rd 2003 Open to consortium members.

The following table-of-contents entries were revised:

	Changed part (section number in the Table of Contents)	<Summary of additions/changes>
1	7.4, 7.7.3 (3)	* Amendments to multicast address number requirements.
2	7.4, 7.7.2	* Amendments to port number requirements.
3	7.1, 7.1.1 (3), 7.4, Fig. 7.10, Fig. 7.12, 7.7.2 (1), 8.1, 8.1.1 (3), Fig. 8.5, Fig. 8.6	* The type of frames stored in ECHONET frame transfer packets was changed from ECHONET frame to ECHONET transmission frame.
4	Chapters 2, 3, 6 and 7	* Correction of incorrect descriptions.

・Version 3.20 Draft October 17th 2003 Open to consortium members.

・Version 3.20 December 12th 2006 Open to the public.

The following table-of-contents entries were revised:

	Changed part (section number in the Table of Contents)	<Summary of additions/changes>
1	2.5	* Addition of descriptions about LowGetProData and LowGetAddress.
2	2.5.5, 3.5.5, 4.7.4, 5.5.6, 6.5.5, 7.8.6, 8.8.4	* Terminology change LowRecvData was changed to LowReceiveData
3	3.5.4, 3.5.5, 5.5.5, 5.5.6, 7.8.5, 7.8.6	* Terminology change LowGetMacAddress was changed to LowGetAddress
4	4.2.7 and other sections	* The misleading mixed use of the names LowWakeUp and LowWakeup was corrected by changing the name LowWakeup to LowWakeUp.
5	4.4.1, 4.4.3, 4.5.5, 5.2.1, 5.4.1, Fig. 5.5, 5.5.5	* Correction of incorrect descriptions.
6	4.7	* Addition of descriptions about LowGetDevID, LowRequestRun and LowGetAddress.
7	7.7	* The descriptions about suspension requests and operation start requests were amended in Table 7.28.
8	7, Appendix 7.1	* Correction of incorrect descriptions.

・Version 3.30 Draft July 30th 2004 Open to consortium members.

The following table-of-contents entries were revised:

	Changed part (section number in the Table of Contents)	Summary of additions/changes
1	1.1	"IEEE802.11/11b" was added in Fig. 1.1.
2	1.2	The "⑧ IEEE802.11/11b-dependent lower-layer communication software" subsection was added to "1.2 Overview of the Lower-Layer Communication Software."
3	1.2	The column for IEEE802.11/11b-dependent lower-layer communication software was added to "Table 1.1 Transmission Media Supported by Individual Types of Software."
4	1.4	The "⑦ IEEE802.11/11b protocol" subsection was added to "1.4 Relationship Between the ECHONET Specification and Other Standards."
5	3.3.1 (5)	A requirement for the modulation speed at the time of registration was added.
6	3.3.2 (2)	・The specifications for communication channel groups were changed. ・Requirements for a function that allows master nodes to change communication channel groups were added.
7	3.4.2 (1)	・The specifications for "Bit synchronization 1"were changed. ・Requirements for repeat transmission count N (min. values) were added. ・A repeat transmission count requirement for ACK transmission was added.
8	3.4.3 (1), (2), (5)(A), (6)(D), (5), (6), (9)	・One of the requirements was changed to permit devices which are only equipped with master node functions. ・A requirement for the transmission speed for the registration mode was added. ・A version information flag setting requirement for the registration mode was added. ・Explanations under the heading 'Data section of data 2 for "(2) Information transmission" ' ("Setup sequence 2") were changed. ・Explanations under the heading "(3) Flag to indicate transmitting Channel No. (3 bits)" were changed. ・ The words "high-order" and "low-order" were added to explanations about flags. ・Requirements about the flag to indicate the content of the information transmission signal were changed. ・Requirements for the flag to provide information on the transmission speeds at which communication is possible were added. ・Requirements for the flag to indicate that a temporary mode change will be made to the continuous operation mode were added. ・Requirements for the flag to provide information on the transmission power output were added. ・Requirements for the flag to provide version information were added. ・ Requirements for the flag to indicate whether the link connection function is implemented were added. ・The "Time for Transmission" tables were changed.

9	3.4.4 (1), (2)	・An ACK signal requirement was added. ・Requirements for link connection were changed. ・The “Response message to be sent back upon receipt of a version information request message” subsection was added. ・The “ Response message to be sent back upon receipt of a reception level information request message” subsection was added. ・The “Vendor Messages” subsection was added. ・The “Temporary continuous reception function” subsection was added.
10	9	“Chapter 9 IEEE802.11/11b Communication Protocol Specifications” was added.

・Version 3.30 December 2nd 2004 Open to consortium members.

The following table-of-contents entries were revised:

	Changed part (section number in the Table of Contents)	Summary of additions/changes
1	9.9.3	The sentence “WEP Length is the size of the authentication key used in shared key authentication.” in the explanations about Table 9-6 (line 6 of P9-26) was changed to “WEP Length is the size of the authentication key.”
2	Appendix 9.1	The explanation about the authentication key under f. was changed from “WEP used in shared key authentication” to “WEP.”

・Version 3.40 Draft December 28th 2004 Open to consortium members.

The following table-of-contents entries were revised:

	Changed part (section number in the Table of Contents)	Summary of additions/changes
1	1.1	Power Line Communication Protocol C System and Power Line Communication Protocol D System were added in Fig. 1.1. The layout was changed.
2	Chapter 10	Specifications for the Power Line Communication Protocol C System were introduced.
3	Chapter 11	Specifications for the Power Line Communication Protocol D System were introduced.

・Version 3.40 February 3rd 2005 Open to consortium members.

The following table-of-contents entries were revised:

	Changed part (section number in the Table of Contents)	Summary of additions/changes
1	3.4.3	Explanations about the radio system identification code were amended.

・Version 3.41 May 11th 2005 Open to consortium members.

・Version 3.2 October 13th 2005 Open to the public.

・Version 3.42	October 27th 2005	Open to consortium members.
・Version 3.50 Draft	August 3rd 2006	Open to consortium members.
・Version 3.50	September 20th 2006	Open to consortium members.
・Version 3.51 Draft	February 2nd 2007	Open to consortium members.
・Version 3.60	March 5th 2007	Open to consortium members.
	December 11th 2007	Open to the public.

The specifications published by the ECHONET Consortium are established without regard to industrial property rights (e.g., patent and utility model rights). In no event will the ECHONET Consortium be responsible for industrial property rights to the contents of its specifications.

The publisher of this specification is not authorized to license and/or exempt any third party from responsibility for JAVA, IrDA, Bluetooth or HBS.
A party who intends to use JAVA, IrDA, Bluetooth or HBS should take action in being licensed for above-mentioned specifications.

In no event will the publisher of this specification be liable to you for any damages arising out of use of this specification.

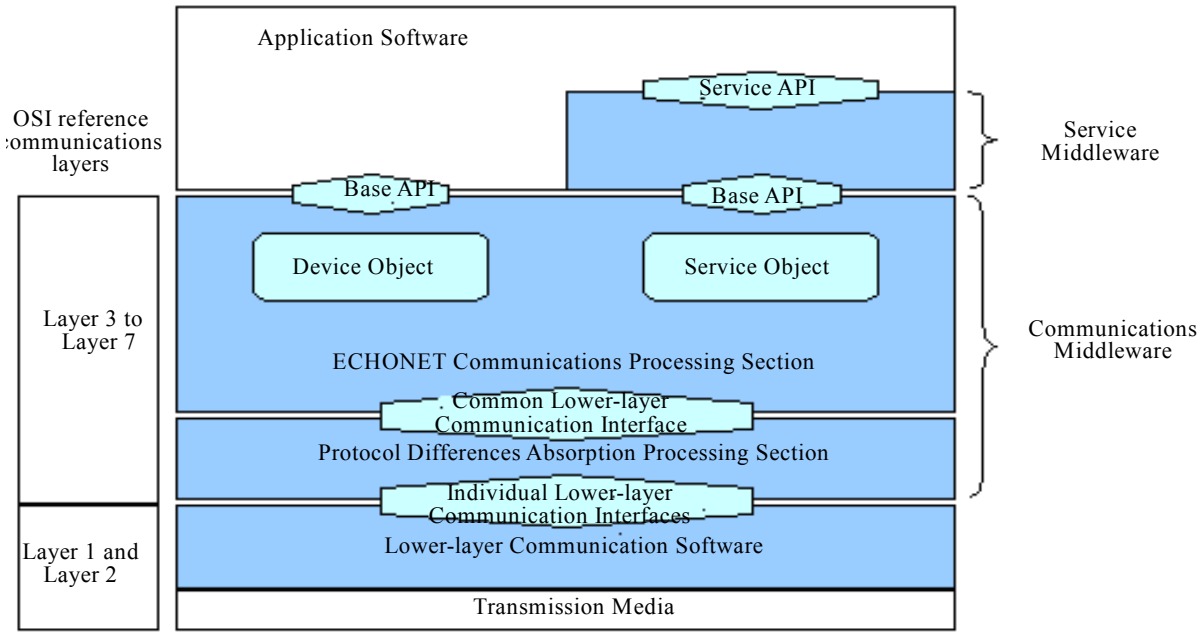
The original language of The ECHONET Specification is Japanese. The English version of the Specification was translated the Japanese version. Queries in the English version should be refereed to the Japanese version.

Contents

Chapter 1 Overview of Specifications for Transmission Media Communication Protocol and Lower-Layer Communication Software

1.1 Relationship Between the ECHONET System and Transmission Media

Figure 1.1 shows the relationship between the ECHONET system and the transmission media covered by this ECHONET Specification, in the communications layer hierarchy. Seven types of transmission media are addressed: household power distribution line, low-power radio, extended HBS, infrared IrDA, LonTalk®, Bluetooth™, and Ethernet. The requirements for connection to the Communication Middleware are specified in the section for individual lower-layer communication interfaces of Part 6 (A to G in Fig. 1.1).



Lower-layer Communication Software Supported by the Current Version

Simbol	Name of Lower-layer Communication Software	Transmission Medium
A	Power Line Communication Protocol A System Power Line Communication Protocol D System	Power distribution lines
B	Low-power Radio	Low-power radio
C	Extended HBS	Twisted-pair cables
D	IrDA Control	Infrared
E	LonTalk®	Low-power radio
F	Bluetooth® (UDP/IP)	Low-power radio (BT)
G	Ethernet IEEE802.3 (UDP/IP)	Ethernet
H	IEEE802.11 IEEE802.11b (UDP/IP)	Low-power radio (WLAN)
I	Power Line Communication Protocol C System	Power distribution lines

Fig. 1.1 ECHONET Architecture

1.2 Overview of the Lower-Layer Communication Software

This ECHONET Specification specifies the requirements for the 7 types of lower-layer communication software listed below. The detailed requirements for the individual types of lower-layer communication software are stated in Chapter 2 and succeeding chapters.

① Lower-layer communication software for household power distribution line communications

Lower-layer communication software of a direct spectrum spread type for existing indoor household power distribution lines that is in compliance with the Radio Law Enforcement Regulations.

② Lower-layer communication software for low-power radio communications

Lower-layer communication software for low-power radio communications that is in compliance with the ARIB STD-T67 and STD-30 Standards.

③ Lower-layer communication software for extended HBS

Lower-layer communication software that is designed for twisted-pair cables by accommodating the ET-2101 (HBS) Standard of JEITA (formerly EIAJ; JEIDA and EIAJ were merged on November 1, 2000, to become JEITA). The establishment of the ET-2101 Standard brought with it such changes as extending the maximum transmission distance (to 1 km), accommodating the use of one pair of media and introducing the address overlap detection function.

④ IrDA-dependent lower-layer communication software

Lower-layer communication software for infrared communications that is in compliance with the IrDA CIR Standard (IrDA Control). The standard communication distance is 8 m, the transmission speed is 75 kbps and the response time is normally 13.8 msec. The lower-layer communication software is capable of “one-to-multiple (up to 8)” communication

⑤ LonTalk®-dependent lower-layer communication software

Lower-layer communication software for low-power radio communications that is in compliance with the LonTalk® protocol. As it can accommodate a wide range of media because of the transmission media-independent protocol processing, use of this type of software for other media shall be considered in the future.

⑥ IP/Bluetooth-dependent lower-layer communication software

Lower-layer communication software for Bluetooth™ radio communications that is in compliance with the Bluetooth™ Standard and the ARIB STD-T66 Standard. This type of software uses UDP/IP as the protocol.

⑦ IP/Ethernet/IEEE802.3-dependent lower-layer communication software

Lower-layer communication software for Ethernet- or IEEE802.3-based media. This type of software uses UDP/IP as the protocol.

Table 1.1 shows the type of transmission media supported by the individual types of software.

⑧ IEEE802.11/11b-dependent lower-layer communication software

Lower-layer communication software which uses media that is based on the IEEE802.11 or IEEE802.11b standard. This type of software uses UDP/IP as the protocol.

Table 1.1 Transmission Media Supported by Individual Types of Software

Type of transmission media	Household power distribution lines	Low-power radio	Infrared	Pair cables
Lower-layer communication software for household power distribution line communications	○	○	-	-
Lower-layer communication software for low-power radio communications	-	-	-	-
Lower-layer communication software for extended HBS	-	-	-	○
IrDA-dependent lower-layer communication software	-	-	○	-
LonTalk®-dependent lower-layer communication software	-	○	-	-
IP/Bluetooth™-dependent lower-layer communication software	-	○(Bluetooth)	-	-
IP/Ethernet/IEEE802.3-dependent lower-layer communication software	-	-	-	○ (including coaxial fiber cables)
1) IEEE802.11/11b-dependent lower-layer communication software		○ IEEE802.11/11b		

The lower-layer communication software shall have the following functions:

* A function to maintain the uniqueness of the nodes' MAC addresses within the subnet
* A function to serve as a container for ECHONET messages.
* An intra-subnet communication function
* A function to allow the individual nodes to retain their own profiles and report them to the Communication Middleware

MAC address length
MAC address mask pattern
(A separate conversion rule is applied in the case of NULL.)
MAC address
Maximum message length
Lower-layer communication software ID
Transmission medium ID
Broadcast function ID
Transmission rate

* A function to allow the individual nodes to retain their own statuses and report them to the Communication Middleware

The compulsory status items are as follows:

* Stop
* Initialization
* Normal operation
* Error stop
* Suspension

For the sequences for individual types of media, refer to the basic sequences specified in the applicable chapters.

1.3 Overview of the Supported New Transmission Media

The types of new transmission media supported by the ECHONET system and their characteristics are as follows:

① Household power distribution lines

Existing indoor household power distribution lines are used as the transmission medium. Because communication signals are transmitted using existing indoor household power distribution lines, there is no need to newly install cables or wires. The consequent reduction in installation work is a major advantage of using this type of transmission medium. The ECHONET Consortium at this time proposes a direct spectrum spread method-based system. The types of facility for which household power distribution line-based data transmission systems can be used and the envisaged applications are as follows:

* Types of facility for which household power distribution line-based data transmission systems can be used: private houses, stores, small- and medium-sized concrete buildings, etc.

(100/200V lines that form part of the single-phase, 2-wire, 100V power system or the single-phase, 3-wire, 100/200V power system)

* Envisioned applications of the household power distribution line communication protocol:

EMS (efficient use of energy), centralized monitoring, control and maintenance of equipment and other similar applications

② Low-power radio

Because this type of transmission medium does not require the installation of signaling lines and is therefore easy to adopt, it is useful for both newly constructed and existing buildings. In addition, the use of batteries would allow its application for equipment installed in places with no AC power source or for portable equipment.

Low-power radio offers the following advantages:

(1) Transmission power: A transmission distance of several to several tens of meters can be achieved with the maximum output power of 10 mW (100 mW in the case of Bluetooth™).

(2) Through-wall communication between rooms or between buildings is possible.

(3) Because low-power radio is regulated by law, it is unlikely that the frequency band would be used in an indiscriminate manner.

(4) The user is not required to obtain a radio license.

Specifications complying with STD-T30 have an intermittent reception function that reduces the power consumption during the standby state waiting for a message and allows the system to be used for a long period of time with batteries.

③ Infrared

* Does not require the installation of signaling lines (which means a significant reduction in installation work).
* Has great potential for use with portable equipment.
* Offers a high level of security because there is no signal leakage to outside the building.

④ Twisted-pair cables

* Highly reliable
* Highly secure

1.4 Relationship Between the ECHONET Specification and Other Standards

① Household power distribution lines

This Specification is in compliance with the Radio Law and the Radio Law Enforcement Regulations. Copies of the Regulations can be obtained from the Association for the Promotion of Telecommunications (incorporated foundation) (Tel: +81-3-3940-3951, Fax: +81-3-3940-4055).

② Low-power radio

This Specification is in compliance with the ARB Standards that incorporate any of the following provisions:

* Laws: Radio Law, Telecommunications Business Law

* Ministerial ordinances issued by the Ministry of Internal Affairs and Communications based on applicable laws: Radio Law Enforcement Regulations, Regulations for Radio Facilities and Equipment, Regulations for Technical Standard Compliance Certification, Telecommunications Business Law Enforcement Regulations, Regulations for Terminal Equipment, Regulations for Technical Standard Compliance Accreditation

* Official notices issued by the Ministry of Internal Affairs and Communications based on applicable laws and ministerial ordinances:

Copies of ARIB Standards can be obtained from the Association of Radio Industries and Businesses (incorporated association) (Tel: +81-3-5510-8590, Fax: +81-3-3592-1103, Web site: http://www.arib.or.jp/).

③ Infrared

This Specification is in compliance with the IrDA Control Standard promulgated by the Infrared Data Association (IrDA). Copies of the IrDA Control Standard can be obtained on the IrDA web site (http://www.irda.org/).

④ LonTalk® protocol

The LonTalk® protocol is used for transmission media communication protocol Layers 1 to 3. ARIB STD-T67 is used as the standard for designated low-power radio, which is one of the types of transmission media used.

⑤ IP/Bluetooth™ protocol

This Specification is based on the Bluetooth™ Standard promulgated by Bluetooth SIG (a

special interest group) and uses the UDP/IP-related Internet standards. ARIB STD-T66 is used as the low-power radio standard.

⑥ IP/Ethernet/IEEE802.3 protocol

This Specification is based on the Ethernet and IEEE802.3 Standards and uses the UDP/IP-related Internet standards.

⑦ IEEE802.11/11b protocol

This Specification is based on the IEEE802.11 Standard, including the IEEE802.11b Standard (which is aimed at achieving higher-speed communications in the physical layer), and uses the UDP/IP-related Internet standards.

Chapter 2 Power Line Communication Protocol Specifications

2.1 System Overview

This is a high-reliability data transmission system based on the direct spread spectrum method. It has high-degree-of-freedom receiving systems, and can cope with deterioration (distortion, noise) of transmission line characteristics.

2.1.1 Scope of the Standard

This Standard consists of mechanical/physical specifications, electrical specifications, and logical specifications for Layer 1 and logical specifications for Layers 2 and 3. The mechanical/physical specifications specify connectors and intended power lines. The electrical specifications specify the modulator/demodulator unit. The logical specifications of Layers 1 to 3 specify the processing for each layer and the signal interfaces between layers.

The electrical interface specifications between the electrical specifications (modulator/demodulator unit) and logical specifications are not specified.

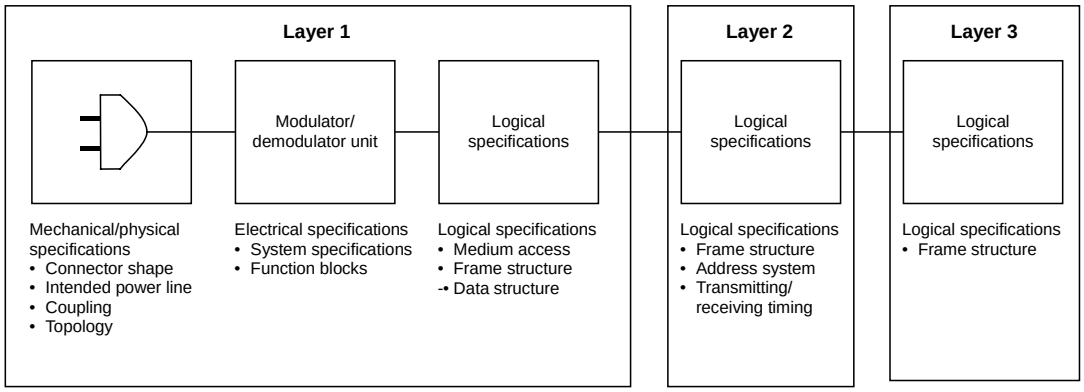


Fig. 2.1 Scope of the Standard

2.2 Mechanical/Physical Specifications

2.2.1 Connector shape

AC plug, plug socket, direct connection

2.2.2 Intended power line

The electrical system of the intended power line shall be single-phase 2-wire or single-phase 3-wire, 100 V or 200 V.

However, 3-wire lines must provide a means for transmitting signals between phases.

* Measures for 3-phase 3-wire 200 V power cable shall be discussed in the future as necessary.

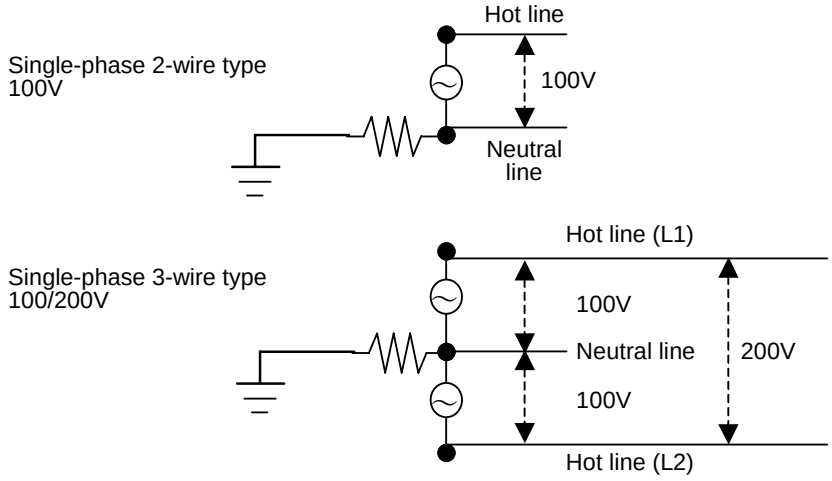


Fig. 2.2 Electrical System

2.2.3 Medium specifications

(1) Coupling system

A line coupling system for injecting signals between L1 and the neutral wire, between L2 and the neutral wire, or between L1 and L2 shall be adopted.

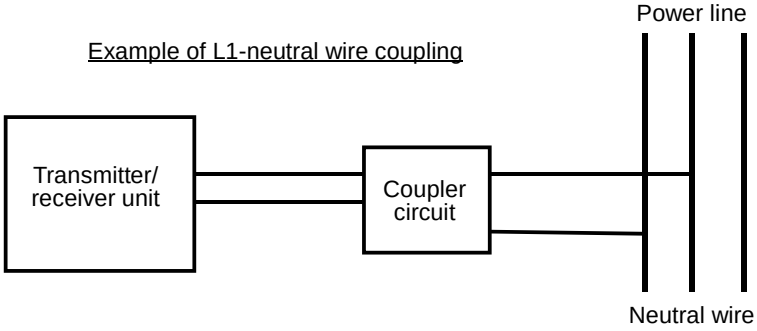


Fig. 2.3 Coupling System

2.2.4 Topology

No special restrictions shall be specified regarding the topology for working with the power line laying form of dwellings, medium/small buildings and stores.

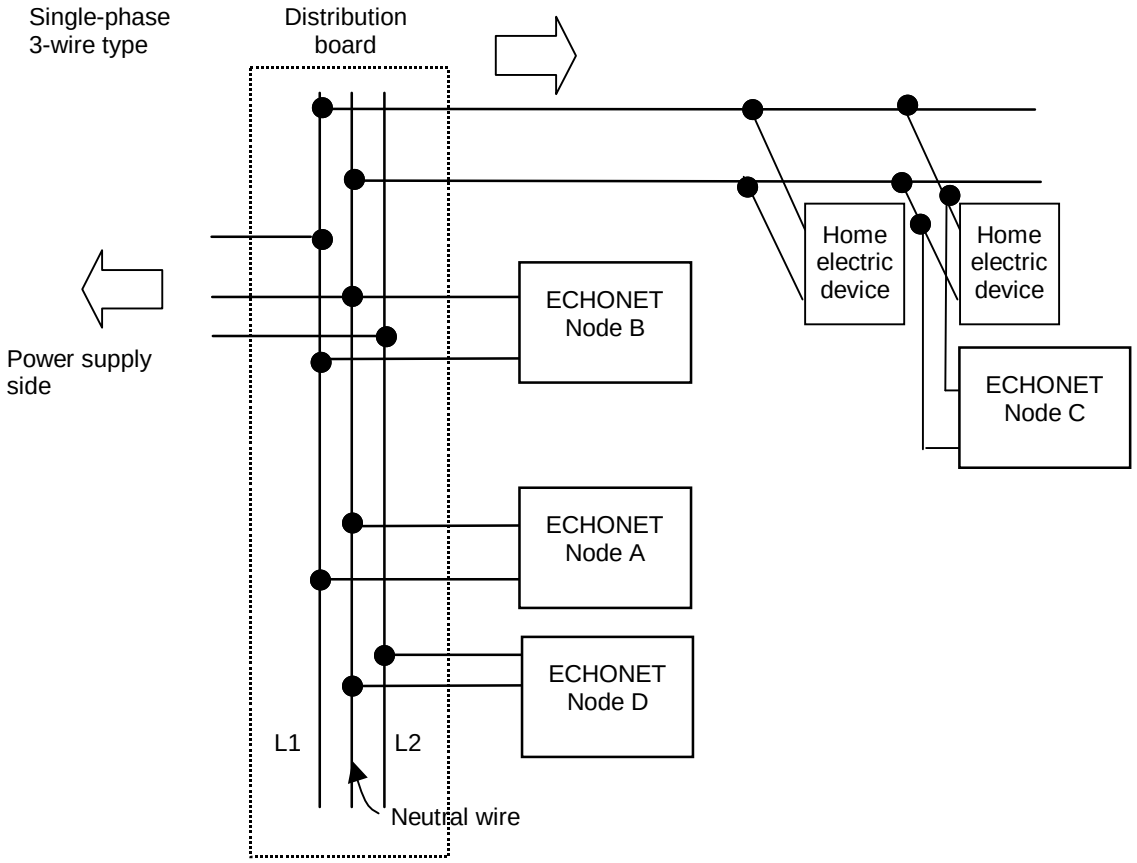


Fig. 2.4 Power Line Topology

* Communication quality between ECHONET nodes deteriorates with transmission distance and the number of home electric devices and other appliances. For example, the characteristics of the transmission path between Nodes A and C are more likely to deteriorate than those between Nodes A and B because of the influence of home electric devices between the nodes.

Additionally, it is difficult to establish communication between terminals A and D, which are connected to different lines. Here, communication can be established by connecting an HPF or the like between L1 and L2 or by connecting an ECHONET router between the terminals connected to L1 and L2 while regarding such terminals as different subnets.

Furthermore, the 200 V air conditioner power line (between L1 and L2 in Fig. 2.4) and 100 V power line, for example, do not carry communication high-frequency signals. Therefore, they are regarded as different subnets. To establish communication between such power lines, it is necessary to furnish a repeater device such as an ECHONET router. Some power lines may require the installation of a blocking filter.

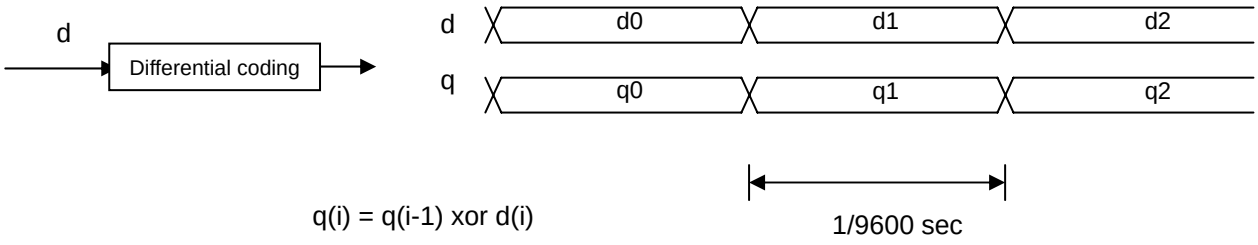
2.3 Electrical Specifications

The power line carrier system of this Standard shall conform to the Radio Law Enforcement Regulations, Article 46-2-6 “Conditions for Special Carrier Digital Data Transmitters Using the Spectrum Spread Method for Carrier Wave Modulation” (as of December 2000; the article number was changed by Posts and Telecommunications Ministry Ordinance 60 on July 12, 1999).

2.3.1 System specifications

(1) Spread spectrum system
Direct spread spectrum
Spread code: The data 1 bit and spread code shall agree in length.
The spread code group or chip length is not stipulated.

(2) Primary modulation system
Differential coding



(3) Transmission rate
9600 bps ±50 ppm

(4) Carrier sense sensitivity
Input power 0.1 mW or less

(5) Transmitting power
10 mW/10 kHz or less (Maximum value is 120% or less of rated value.)

(6) Spread frequency range
10 kHz to 450 kHz
(A spread shall take place within a frequency range of at least 200 to 300 kHz.)

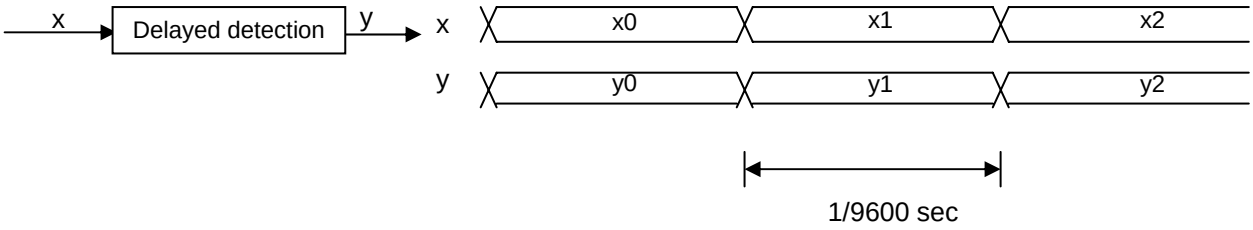
(7) Spurious signal intensity at output terminal
450 kHz – 5 MHz (inclusive): 56 dBμV or less
5 MHz – 30 MHz (inclusive): 60 dBμV or less

(8) Electric field leakage (at a distance of 30 m from the transmitter)
(A) Frequency in spread range: 100 μV/m
(B) 526.5kHz to 1606.5kHz: 30 μV/m
(C) Frequency other than A and B: 100 μV/m

(9) Receiver sensitivity
Input power 0.1 mW or less

(10) Demodulator detection method

Delayed detection



* The symbol “x” represents a received spread signal or part of it. Spread code inverse spread and differential decoding take place simultaneously.

(A) When input x is a binary signal

$y(i) = x(i)$ xor $x(i - 1)$

(B) When input x is multivalue digital signal or analog signal

$$y(i) = \begin{cases} 0 & x(i) \text{ is in phase with } x(i-1). \\ 1 & x(i) \text{ is in opposite phase with } x(i-1). \end{cases}$$

* Multivalue digital signal: Either a digital signal having a larger number of voltage levels than a binary one or a 2k-value digital signal (k = 2 or greater integer) transmitted by a bus comprising two or more binary signal lines.

Supplement 1: Example of Modulator/Demodulator Unit Configuration

The power line is not designed to transmit high-frequency signals for communication for its primary use and has noise, attenuation, and impedance variation caused by home electric devices.

The characteristics of the power line as a transmission line differ significantly depending on the place of application. For this reason, free selection of a modulation/demodulation system shall be allowed without specifying a specific demodulation system. From the viewpoint of connectivity, different demodulation systems are acceptable.

Supplement 1.1 Modulator Unit Configuration Example

One example of a modulator unit is shown in Fig. 2.5. The modulator unit consists of a data differential coder, spread code generator, and multiplier block for multiplying differential coded data by a spread code.

Any optional spread code may be freely selected and shall not be specified.

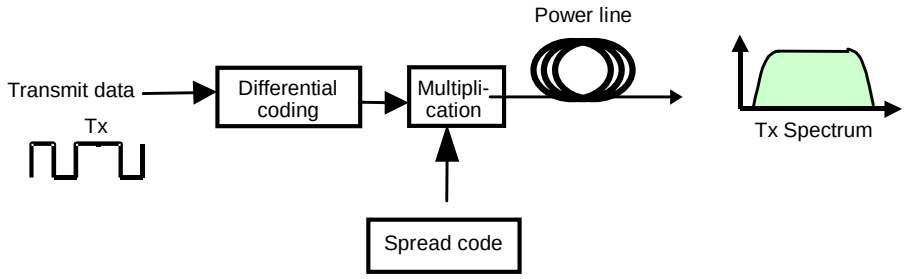
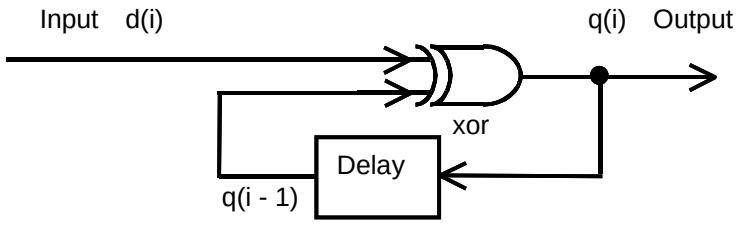


Fig. 2.5 Modulator Unit (Direct Spread Spectrum) Configuration Example

Supplement 1.2 Example of Differential Coding Block Input/Output Data

Input data		1	1	0	1	0	1	1	0	0	1	0
Output data	(0)	1	0	0	1	1	0	1	1	1	0	0

* When the input data is "0", the immediately preceding output data is output as-is.
When the input data is "1", the immediately preceding data is inverted and then output.
A differential coding block configuration example is shown below:



Supplement 1.3 Demodulator Unit Configuration Example

Figure 2.6 shows a sub-band delay detection system as an example of a demodulator.
This system uses the frequency diversity effect to obtain excellent receiving characteristics even in places with poor transmission characteristics.
As shown in Fig. 2.6, a received spectrum spread signal is frequency-divided using BPF 1 to n. The sub-band width and number of sub-bands are optional.

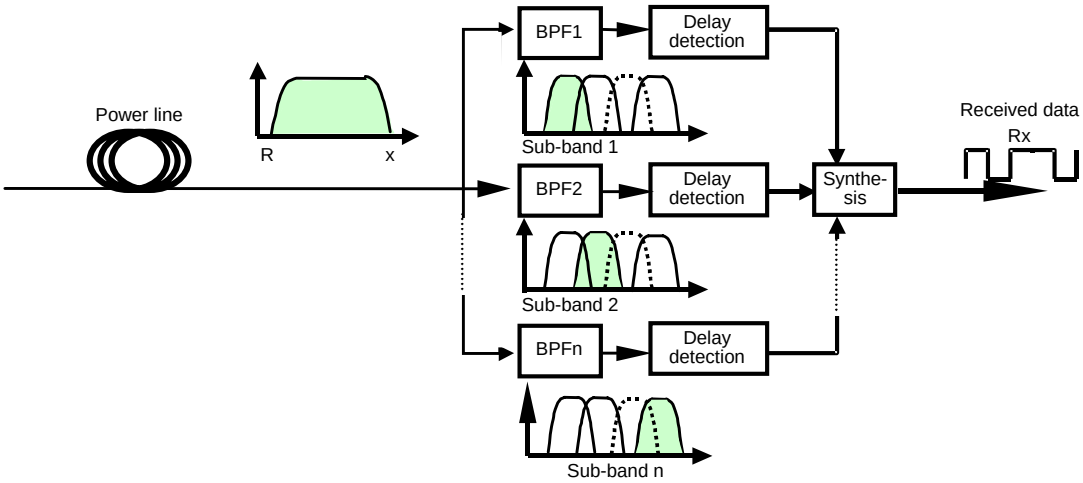


Fig. 2.6 Demodulator (Sub-band Delay Detection System) Configuration Example

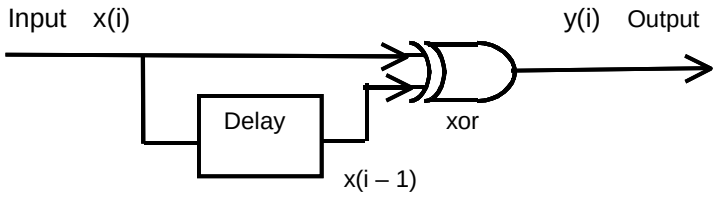
Supplement 1.4 Delay Detection Block Input/Output Data (Example 1)

(When the input is a binary signal and the spread code is 101)

Input signal	010	101	010	010	101	101	010	101	101	101	010	010
Delay signal		010	101	010	010	101	101	010	101	101	101	010
Output data		1	1	0	1	0	1	1	0	0	1	0

* The input signal and the immediately preceding input signal (delay signal) are XORred (exclusive-ORed) and then used as the output signal.

A delay detection block configuration example is shown below:



Supplement 1.5 Delay Detection Block Input/Output Data (Example 2)

(When the input is a multivalue digital signal*)

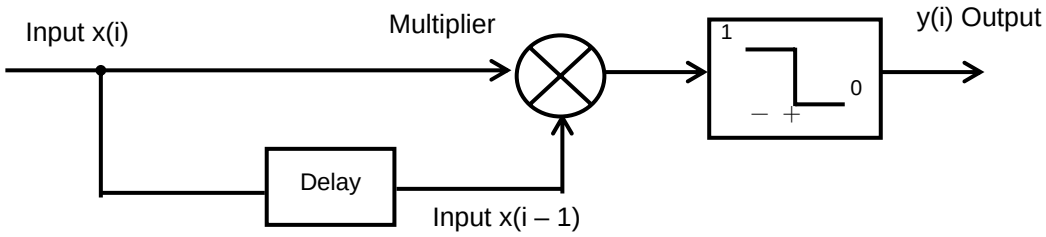
* Multivalue digital signal: Either a digital signal having a larger number of voltage levels than a binary one or a 2^k-value digital signal (k = 2 or greater integer) transmitted by a bus comprising two or more binary signal lines.

Input signal	-1, 2, -1	1, -2, 1	-1, 2, -1	-1, 2, -1	1, -2, 1	1, -2, 1	-1, 2, -1
Delay signal		-1, 2, -1	1, -2, 1	-1, 2, -1	-1, 2, -1	1, -2, 1	1, -2, 1
Multiplication result		-	-	+	-	+	-
Output signal		1	1	0	1	0	1

* The input signal is multiplied by the immediately preceding input signal (delay signal). The obtained multiplication result is converted to a binary equivalent and then output.

- 1 when a minus sign is used (when the input signal is in opposite phase with the delay signal)
- 0 when a plus sign is used (when the input signal is in phase with the delay signal)

A delay detection block configuration example is shown below (implementation is also achievable for an analog input by using an analog circuit in the same configuration):



2.4 Logical Specifications

2.4.1 Layer 1

(1) Transmission control system

CSMA system

(2) Carrier sense

Carrier sense is available. An alternative is allowed.

(3) Pause period

Pause period for normal frames (excluding response signaling and automatic retransmission): 40 ms or longer

(4) Layer 1 frame structure

Preamble	Synchronization code	Frame type	House code	Layer 1 payload

(A) Preamble: Symbol synchronization code

For synchronization between the receiving timing of the receiver and the transmission timing of the transmitter, a preamble is used.

Preamble: 010101….. 0101 (8-byte)

(B) Synchronization code: Frame synchronization code

The synchronization code is inserted between the preamble and the frame type field to indicate the beginning of data. The synchronization code shall be a fixed value. Before transmission, the synchronization code is modulated by the bit modulation system specified by the signal system and then transmitted.

Synchronization code: 1111010110010000

(C) Frame type: Frame length/type definition code

This code specifies SHORT, MIDDLE, LONG, DOUBLE LONG or ANSWER FRAME.

SHORT FRAME (frame type: 00000000)

Preamble 8 bytes	Synchronization code 2 bytes	Frame type 1 byte	House code 8 bytes	Layer 1 payload 40 bytes

MIDDLE FRAME (frame type: 00101110)

Preamble 8 bytes	Synchronization code 2 bytes	Frame type 1 byte	House code 8 bytes	Layer 1 payload 54 bytes

LONG FRAME (frame type: 01001101)

Preamble 8 bytes	Synchronization code 2 bytes	Frame type 1 byte	House code 8 bytes	Layer 1 payload 96 bytes

DOUBLE LONG FRAME (frame type: 01100011)

Preamble 8 bytes	Synchronization code 2 bytes	Frame type 1 byte	House code 8 bytes	Layer 1 payload 176 bytes

ANSWER FRAME (frame type: 10001011)

Preamble 8 bytes	Synchronization code 2 bytes	Frame type 1 byte	House code 8 bytes	Layer 1 payload 16 bytes

(D) House code: ID for house identification

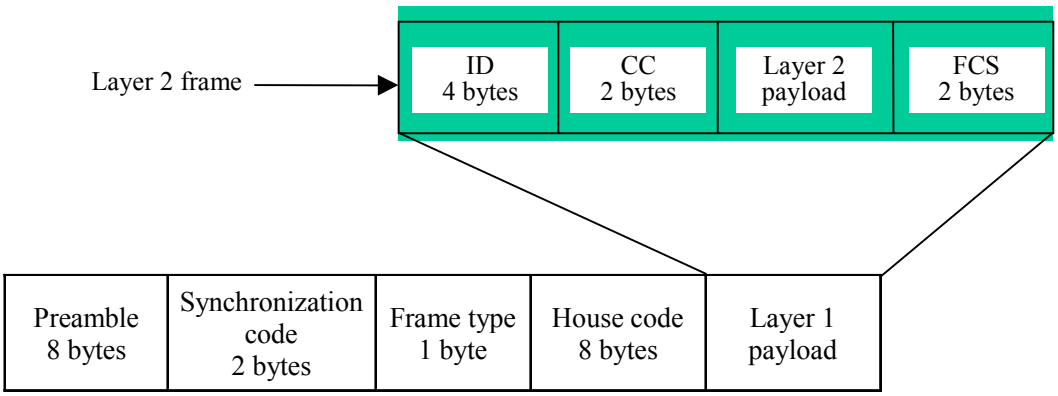
1	2	3	4	5	6		8
Manufacturer code			Identification code				

1. Manufacturer code
   – The 3 high-order bytes of the house code shall be a manufacturer code.
2. Identification code
   – The 5 low-order bytes of the house code shall be a discrete identification code.
   – The company that owns the “Manufacturer codes” shall be responsible for managing identification codes.
   – Unique numbers such as serial numbers shall be assigned.
3. P&P setup reservation code
   – For house code P&P setup purposes, the following house code shall be reserved as a common code for all nodes.
     The use of the announce address 0 for purposes other than transmission/reception shall be prohibited during P&P setup.
   – Reservation code: 0x0000000000000000

(E) Layer 1 payload: For data contents, see Section 2.4.2 (1) Layer 2 Frame Structure.

2.4.2 Layer 2

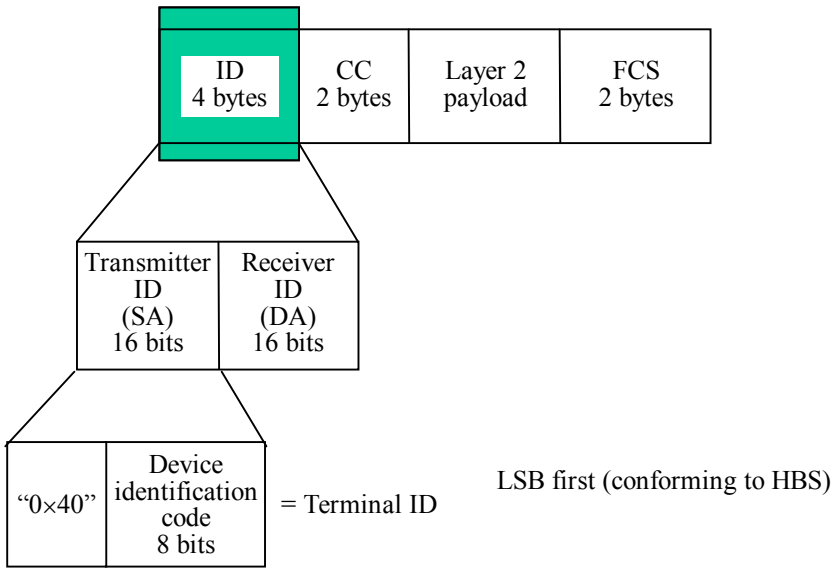
(1) Layer 2 frame structure



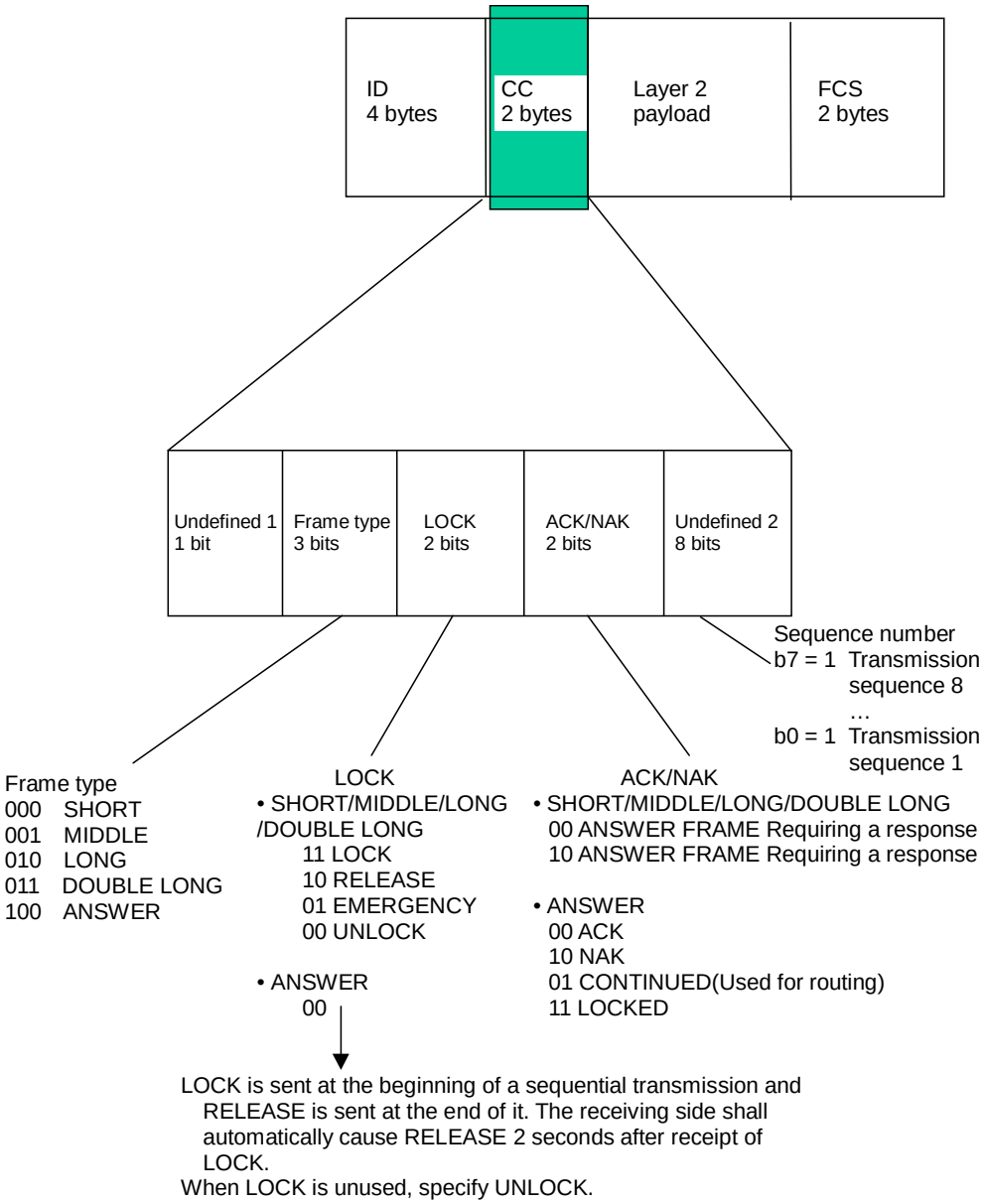
The Layer 2 frame structure is as follows:

(A) ID:

This consists of a transmitter terminal ID (physical address) and a receiver terminal ID (physical address).

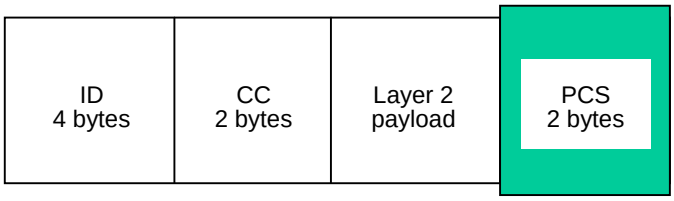


(B) CC: Control code



(C) Layer 2 payload:
For data contents, see Section 2.4.3 (1) Layer 3 frame structure.

(D) FCS: Frame inspection sequence
Items between frame type and Layer 2 payload shall be computed.



FCS determination formula
Generating polynomial G (x) = $X^{16} + X^{12} + X^5 + 1$ (CRC-CCITT recommendations)

(2) Layer 2 address system

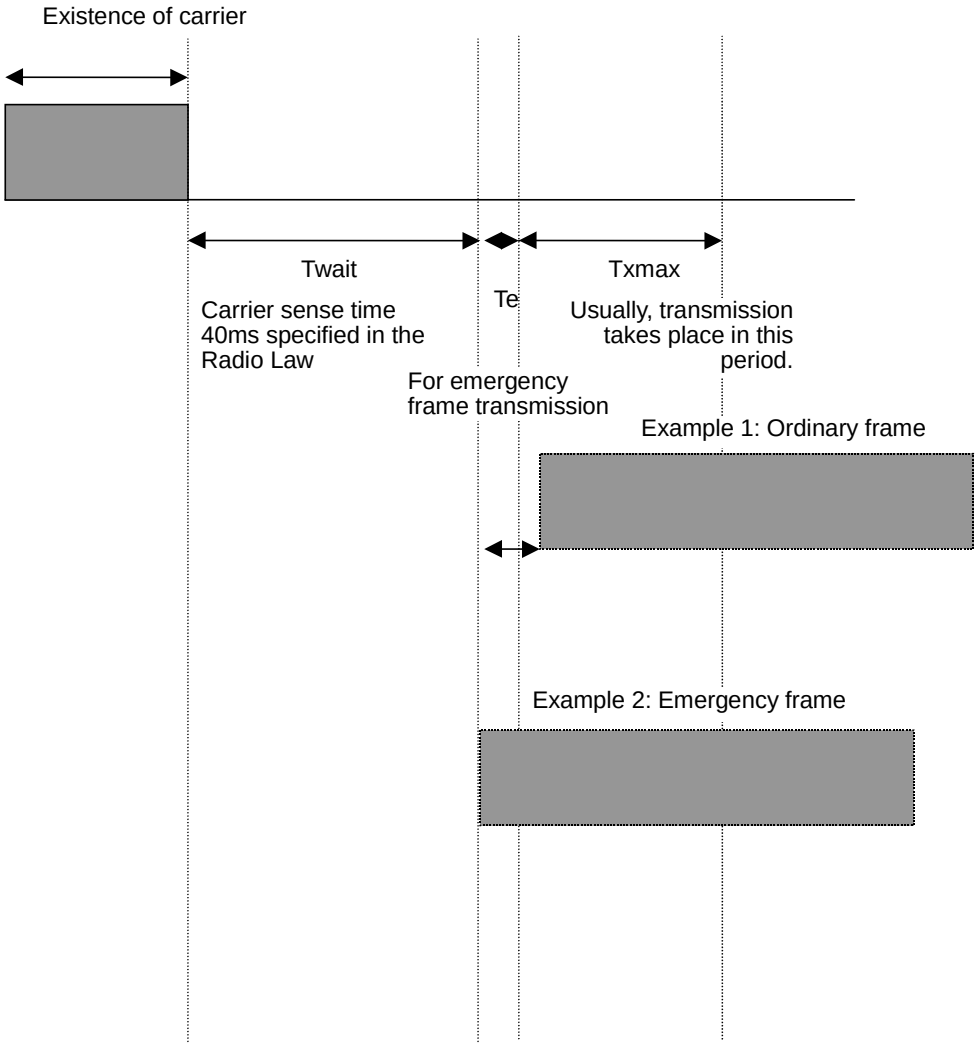
No.	Object	MAC address (HEX)	
1	Plug and play manager address	40	00
2	Discrete address	40	01 to EF
3	Broadcast address	40	F0
4	Reserved for future use	40	F1 to FE
5	Reserved for P&P	40	FF

Here, the 8 high-order bits of the terminal ID shall be fixed at 0x40 for now. (In the future, this will be extended.) The 8 low-order bits shall be a Node ID comprising an ECHONET address.

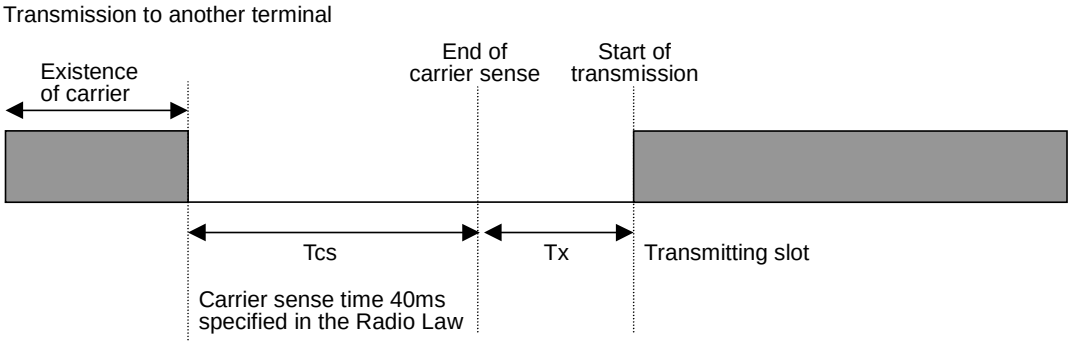
This Standard does not permit the use of a “Reserved for future use” address. However, the reception of a message destined to an “SA = Reserved for future use” address shall be permitted in consideration of future use.

(3) Transmission timing

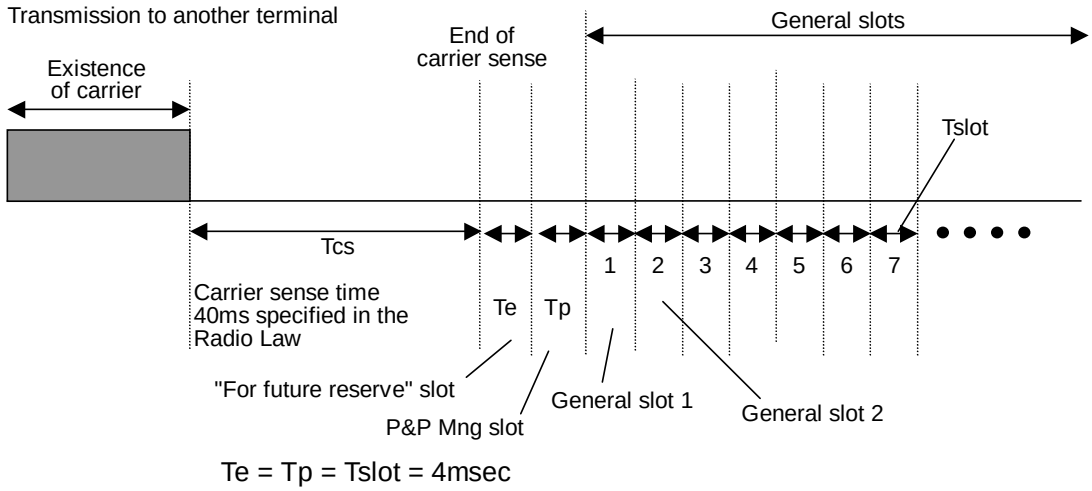
Transmitting timing



1. At the transmission timing on the power line, carrier sense of Tcs = 40 ms is performed in accordance with the Enforcement Regulations of the Radio Law. After completion of carrier sense, a TX slot waiting time is taken and then the corresponding transmission slot is transmitted. For this reason, transmission is started in Tcs + Tx ms after no carrier is available on the power line.



2. Transmission slot
   As transmission slots, a) one “Reserved for future use” slot (the use of this slot is prohibited by this Standard), b) one P&PMng slot, and c) 100 general terminal slots are available. The following timing is specified for these slots.



In the above situation, the P&PMng shall use the P&PMng slot for P&P processing. However, it shall use general terminal slots for the other normal communications.

3. General terminals determine the slots to be transmitted by the magic number Nmagic. Nmagic is generated from terminal native data and terminal independent data, with the following conditions:
   a) A different value should be generated for each transmission.
   b) Even the same types of terminals should be able to output different values.

4. When different terminals attempt to use the same slot at the same time, a collision will occur. When terminals are not provided with collision detecting capability, two types of transmission shall be performed. When an error occurs on the receiving side, the error processing procedure on the receiving side shall be observed.

Transmission timing

Figure 2.7 shows the transmission timing that prevails on the power line.

When the transmitting side transmits a SHORT, MIDDLE, LONG, or DOUBLE LONG frame, the receiving side shall receive it. When the received frame meets the Layer 1 and Layer 2 requirements and the receiver ID (DA) agrees with the receiving side MAC address, the receiver side shall start transmitting an ANSWER FRAME along the power line within the timeout time Tout1.

The Tout1 value shall be 35 ms.

If the above requirements are not met at the receiving side as shown in Fig. 2.8, the receiving side shall not send an ANSWER FRAME. The transmitting side starts a retransmission if it does not receive an ANSWER FRAME within timeout time Tout2 after completion of frame transmission. If retransmissions are to take place because messages transmitted from different terminals are lost due, for instance, to collision, each of such retransmissions shall be performed with random timing, using general terminal slots, to avoid collision.

Up to two retransmissions are to be performed. If no response is obtained after two retransmissions, the transmission process aborts. The Tout2 value shall be 100 ms.

As an exception to transmission timing, during simultaneous broadcast communication or during communication to a provisional address terminal, the receiving side need not send back an ANSWER FRAME even if the received frame satisfies the standard conditions of Layer 1 and Layer 2.

Transmission timing

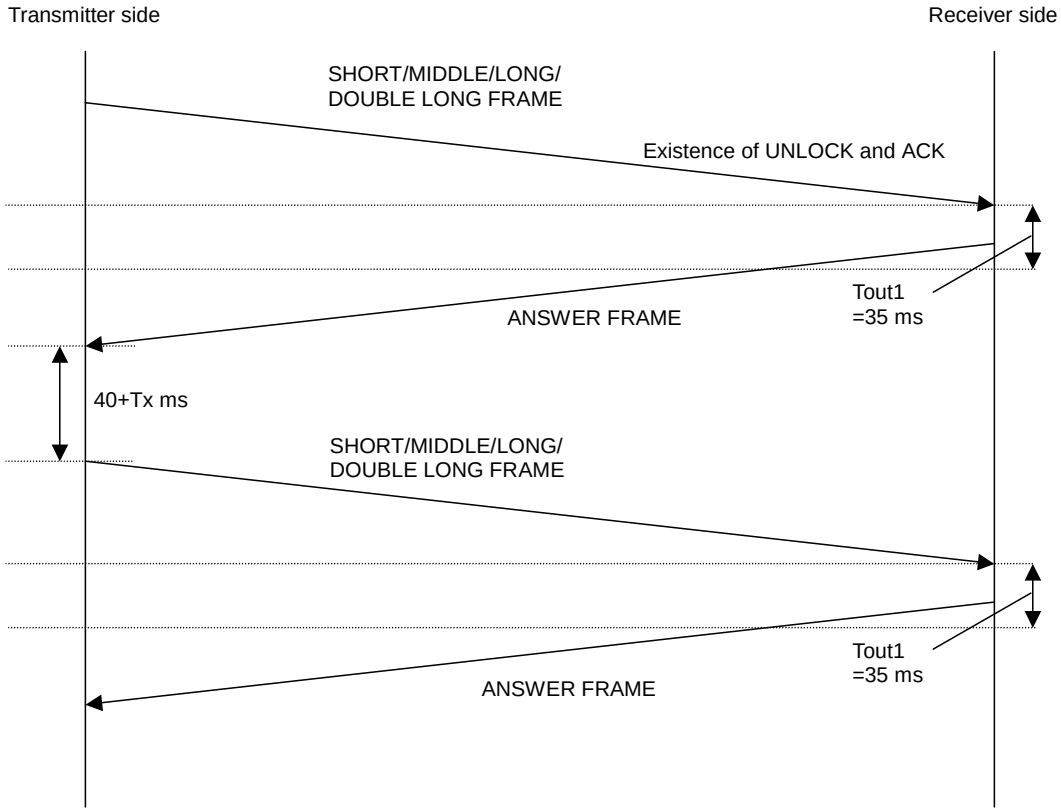
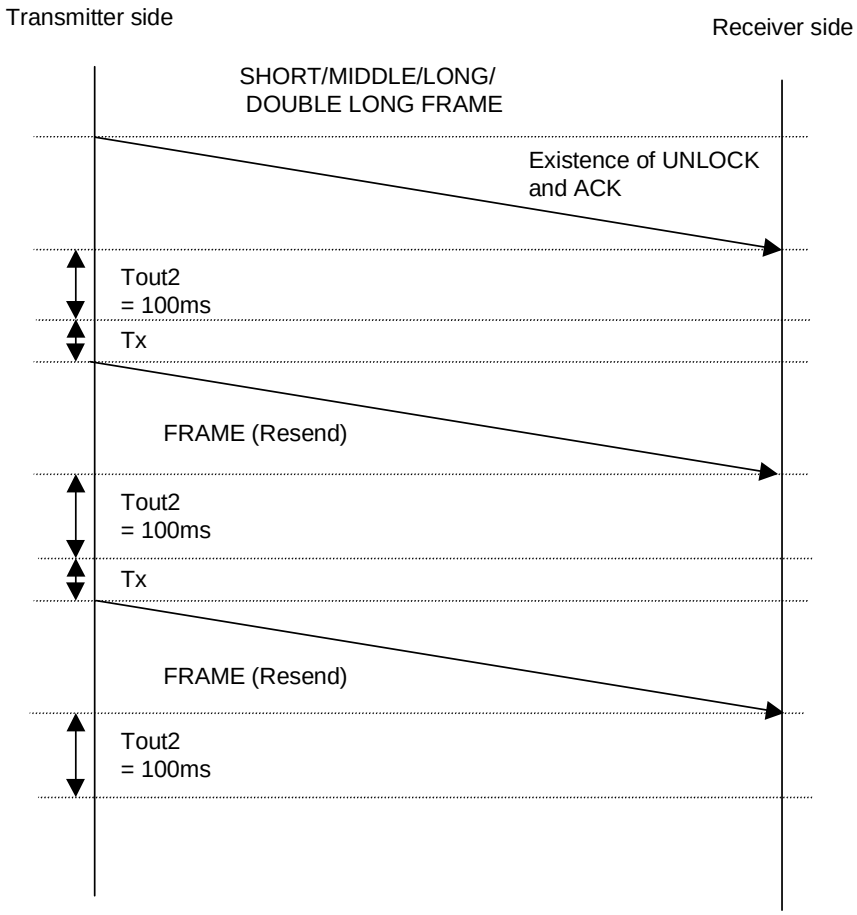


Fig. 2.7 Transmission Timing

Transmission timing (Resend)



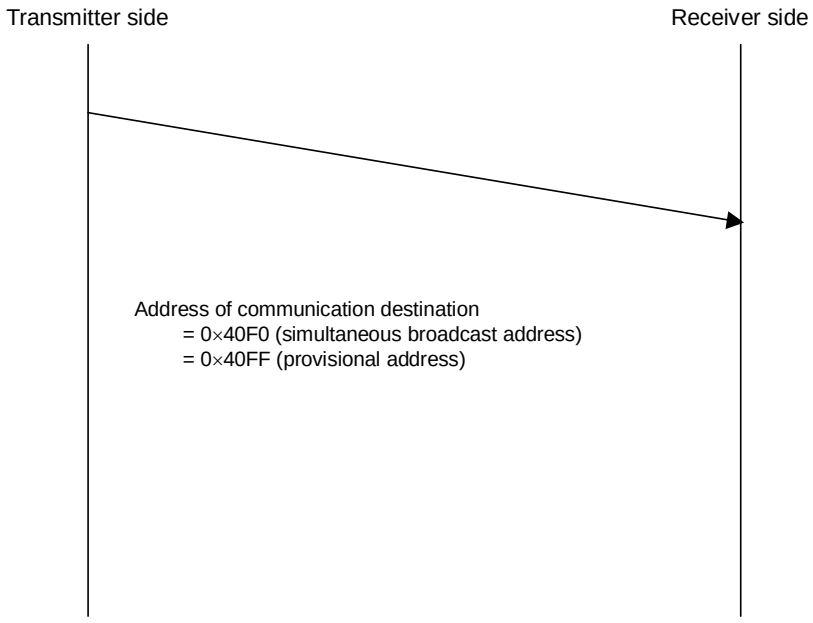
Retransmission shall take place if an ANSWER FRAME has not been received when timeout time Tout2 elapses after completion of frame output completion.

Individual retransmissions shall be performed with random timing (Tx) through the use of general terminal slots.

If no response is received despite two successive retransmissions, the application's judgment shall take precedence.

Fig. 2.8 Transmission Timing (Resend)

Transmission timing (Communication to simultaneous broadcast and provisional address terminals)

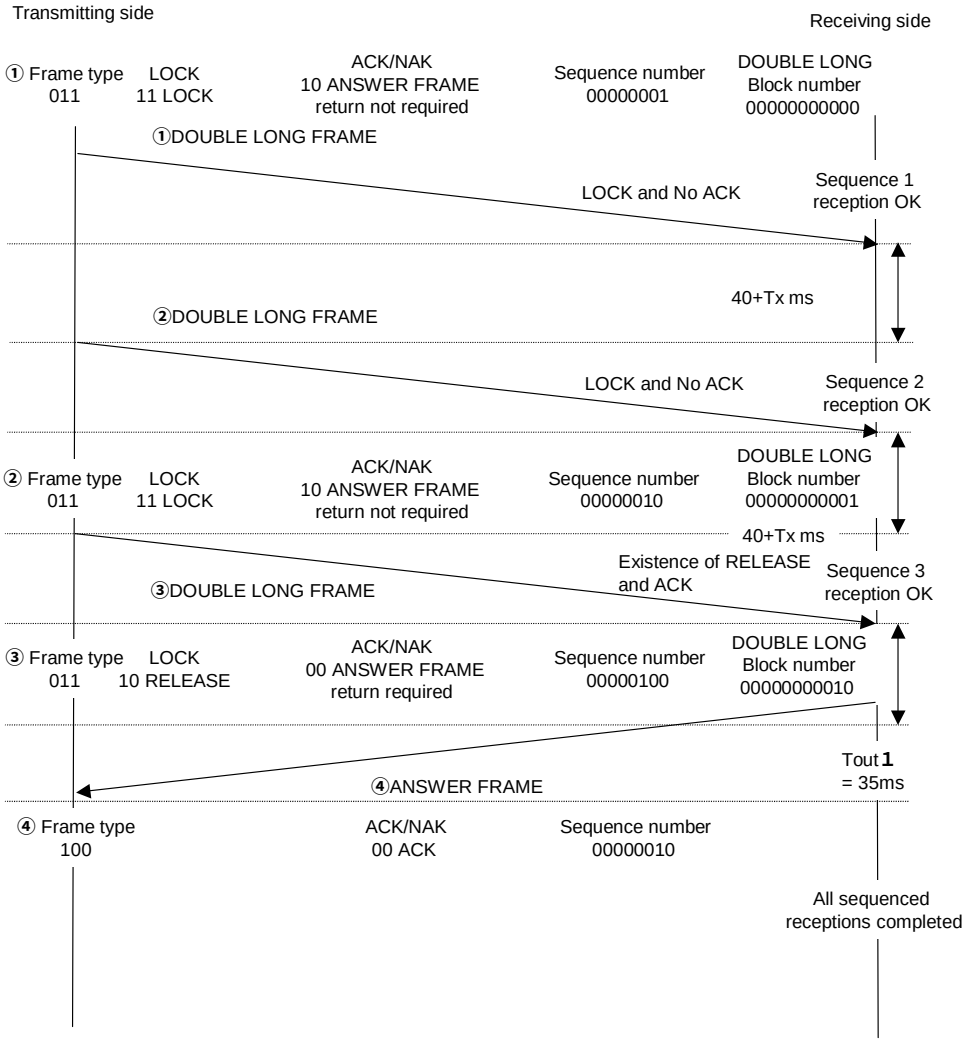


For communication to a simultaneous broadcast address and a provisional address, no ACK/NAK response shall be made in consideration of the traffic volume.

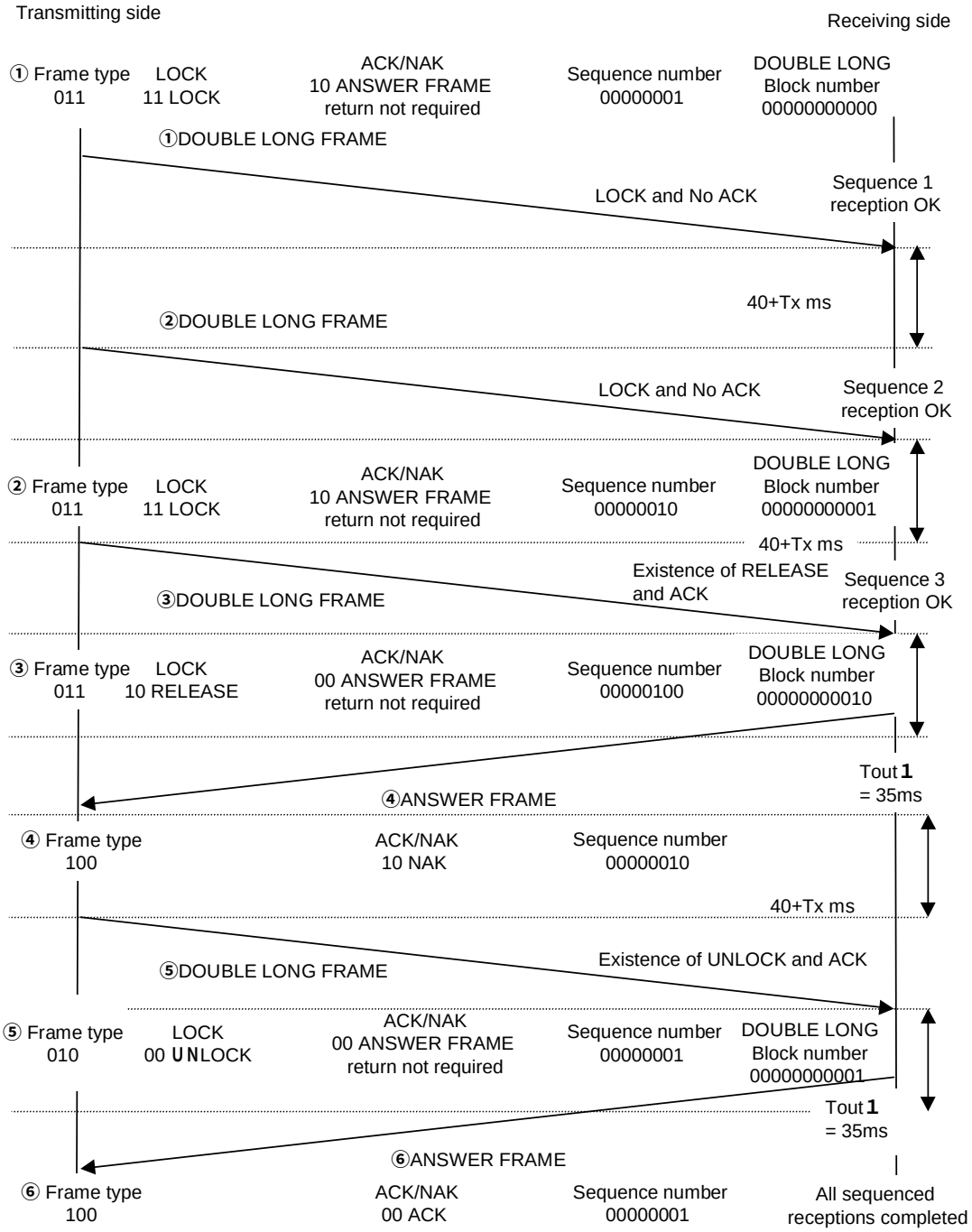
Fig. 2.9 Transmission Timing (Communication to simultaneous broadcast and provisional address terminals)

(4) Communication sequence

The figure below shows the communication sequence for frame-divided transmission (3-frame divided transmission) based on DOUBLE LONG FRAME. For frame segments, "LOCK" and "No ACK" shall be specified with CC.



Communication sequence (frame-divided transmission)



When a reception NG results, the receiving side shall return NAK with the affected sequence number attached.

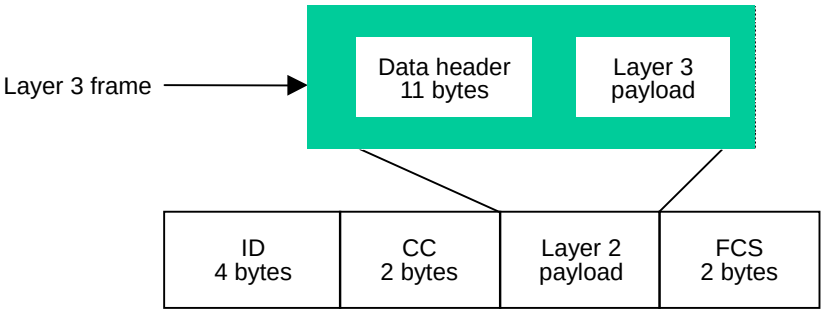
Communication sequence (frame-divided transmission with retransmission)

2.4.3 Layer 3

The data header (consisting of a routing setting code, Block No., valid byte counter (BC), and command selection switch) shall set routing, uniquely identify frames at transmission/reception of a series of frames, specify the number of valid bytes of the Layer 3 payload, and distinguish from other local commands the ECHONET commands within the Layer 3 payload.

(1) Layer 3 frame structure

The Layer 3 frame structure is as follows:

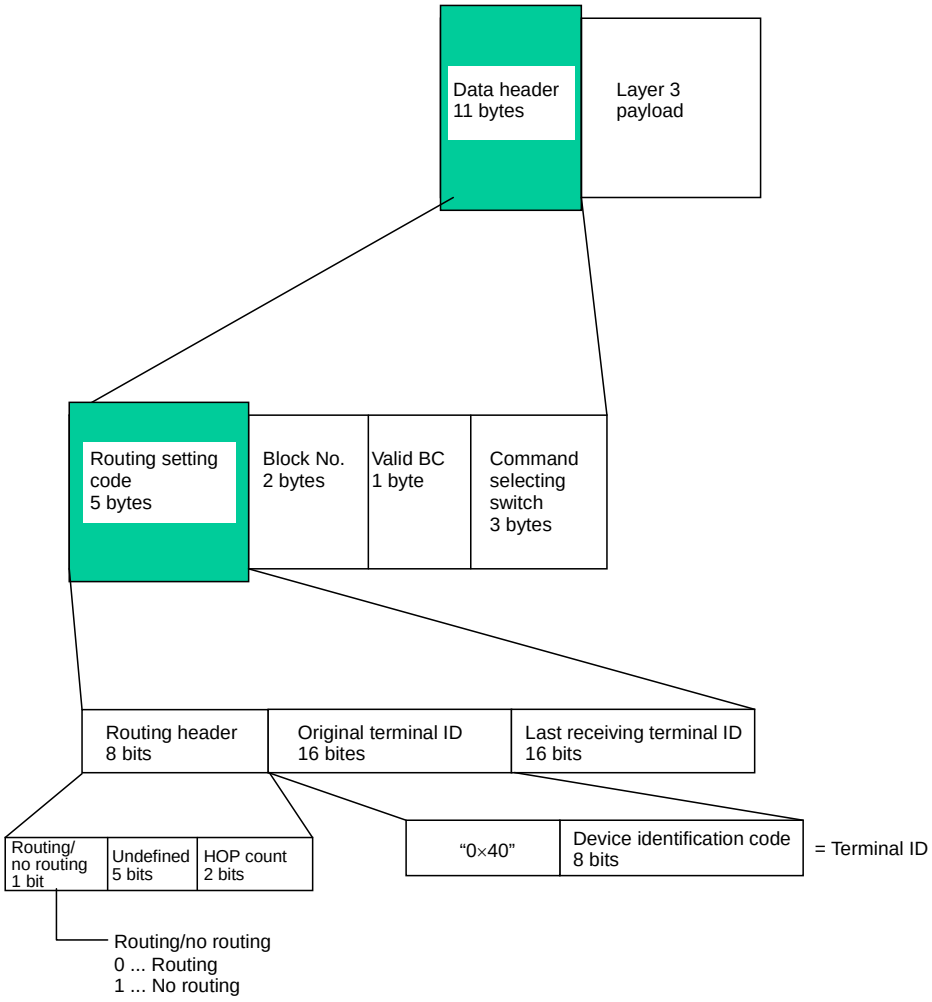


ANSWER FRAME consists only of a data header.

(A) Data header:

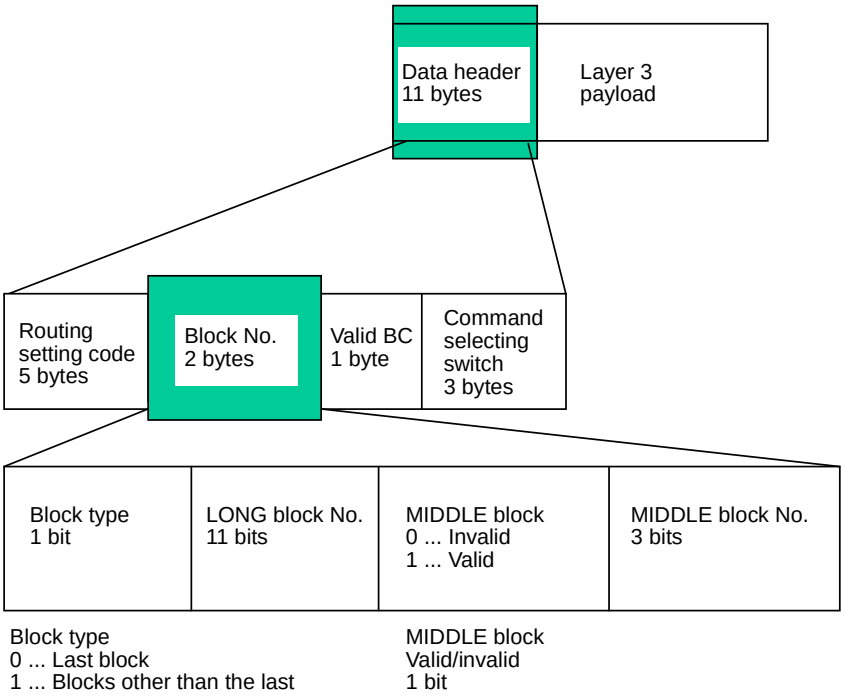
The data header consists of a routing setting code, Block No., valid byte counter (BC), and command selection switch.

- Routing setting code



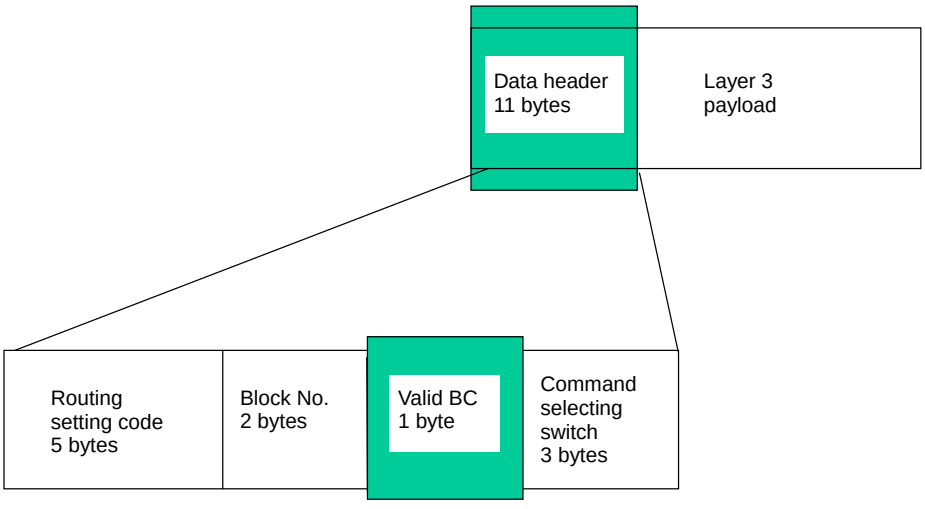
- Block No.

  The Block No. uniquely identifies frames at transmission/reception of a series of frames.



- Valid byte counter

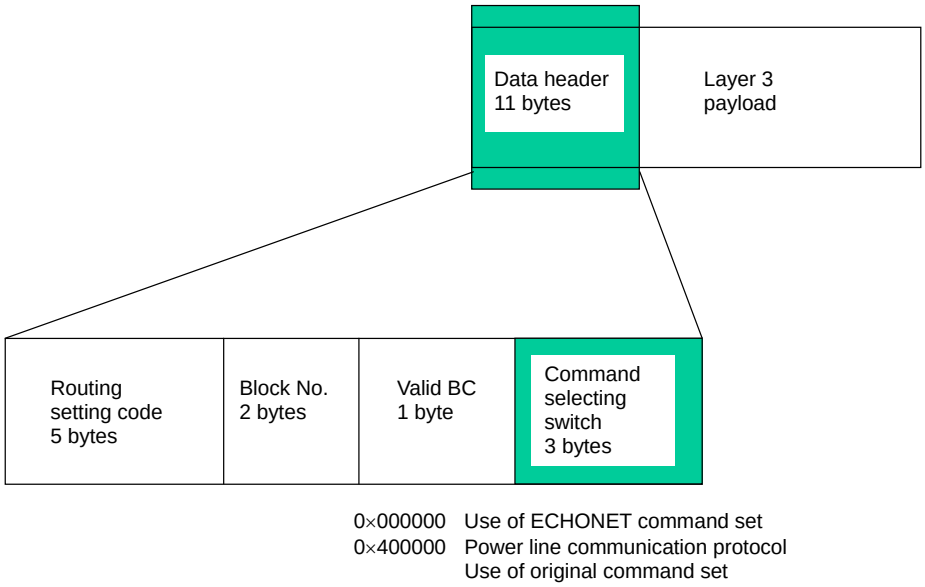
  This counter indicates the number of valid bytes of the Layer 3 payload.



- Command selection switch

  The command selection switch distinguishes ECHONET commands in the Layer 3 payload from other local commands.

  (See Supplement 2.1 “Command Set Unique to Power Line Communication Protocol”.)



2.5 Basic Sequence

This section deals with the following:

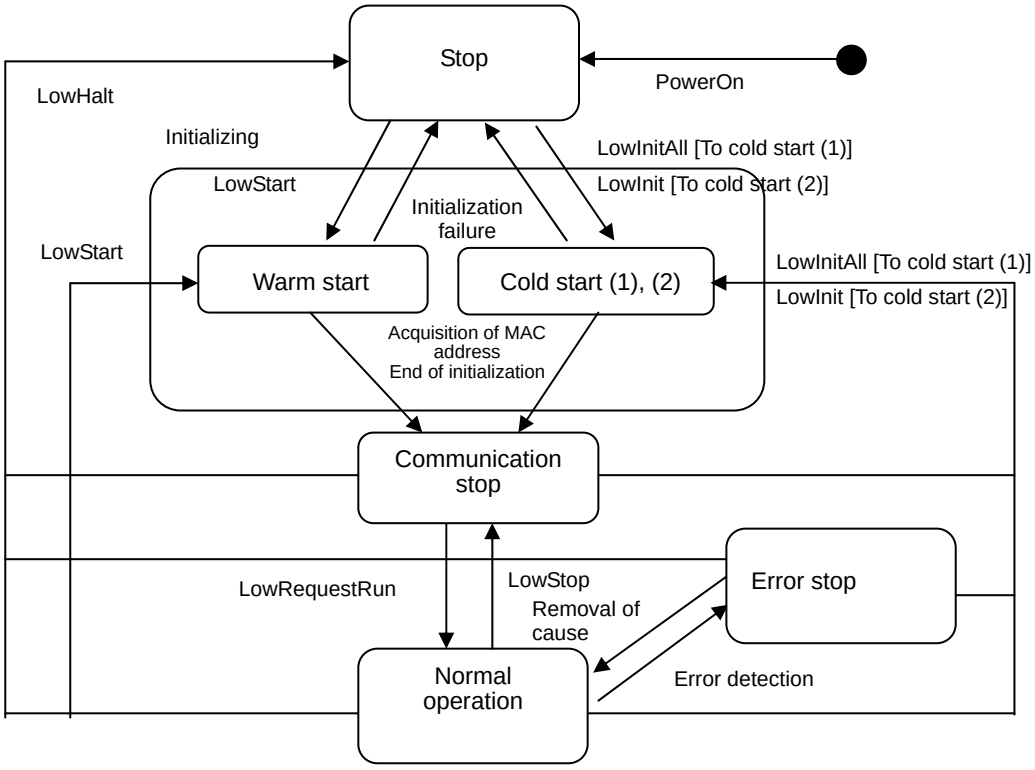
- State transition diagram
- Sequence descriptions of various states indicated in the state transition diagram

2.5.1 Basic concept

This subsection classifies the individual lower-layer communication software status as shown below, and describes an outline of the sequence in each status.

① Stop status
② Initialize processing status
③ Communication stop status
④ Normal operation status
⑤ Error stop status

The following figure shows the state transition diagram for each status.



2.5.2 Stop status

The stop status signifies a state in which no lower-layer communication software operations are performed except for the P&P settings of the lower-layer communication software's independent functions that change due to the installer's Power ON operation, etc. This status is provided immediately after Power On. An outline of the processing immediately after state transition and an outline of the discrete lower-layer communication interface services that the stop status receives, and related processing, are described below.

(1) Trigger and action

Waits for an individual lower-layer communication interface service.

For initializing the transceiver, reset processing is performed immediately after Power On.

(2) Status acquisition service (LowGetStatus)

Returns LOW_STS_STOP as the status.

(3) Lower-layer communication software type acquisition service (LowGetDevID)

Returns the lower-layer communication software type.

The triggers for state transition are as follows:

(1) Transition trigger to initialize processing status

The transition is caused by the initialization service (LowStart, LowInit, LowInitAll).

2.5.3 Initialize processing status

In the initialize processing state, the lower-layer communication software is initialized. This state can be roughly divided into warm start, cold start (1), and cold start (2).

An outline of the processing immediately after state transition and an outline of the individual lower-layer communication interface services that the initialize processing status receives, and related processing, are described below.

(1) Trigger and action

Waits for an individual lower-layer interface service.

(2) LowStart (warm start)

Terminates an initialization process and switches to the communication stop status if the MAC address and house code are retained. If the MAC address and house code are not retained, however, the initialization process fails and the status changes to the communication stop status.

(3) LowInitAll (cold start (1))

If the house code and MAC address have been retained, both are discarded. Furthermore, the following operations are performed to acquire a new house code and MAC address. When the house code and MAC address are acquired, the initialization process ends and the status changes to the communication stop status. If the house code and MAC address cannot be acquired, the initialization process fails and the status changes to the communication stop status.

① Acquiring a house code

A unique house code for power line communication protocol domain identification is acquired. The installer selects an acquisition method manually with a DIP switch, etc. An alternative is to exercise the lower-layer communication software's "Register_ID" functions described in Section 2.6 "P&P Setup of House Code and MAC Address", from the only plug-and-play manager (hereinafter abbreviated to the P&PMng) within the power line domain.

② Acquiring a MAC address unique within a subnet

A MAC address unique within the power line domain is to be acquired. As explained earlier under "1) Acquiring a house code", the installer selects an acquisition manually with a DIP switch. An alternative is to exercise the lower-layer communication software's "Register_ID" functions described in Section 2.6 "P&P Setup of House Code and MAC Address", from the only plug-and-play manager (hereinafter abbreviated as P&PMng) within the power line domain.

(4) LowInit (cold start (2))

When the house code and MAC address are retained, a new MAC address is to be acquired while the retained one is discarded The MAC address acquisition method is the same as explained earlier under "(3) LowInitAll (cold start (1))", "2) Acquiring a MAC address unique within a subnet". When the MAC address is acquired, the initialization process terminates and the status changes to the communication stop status. If the MAC address cannot be acquired, the initialization process fails and the status changes to the stop status. When the house code and MAC address are not retained, the initialization process fails and the status changes to the stop status.

(5) Status acquisition service (LowGetStatus)

In a warm start, returns LOW_STS_RST as the status. In a cold start ((1) or (2)), returns LOW_STS_INIT as the status.

(6) Lower-layer communication software type acquisition service (LowGetDevID)

Returns the lower-layer communication software type.

Triggers for state transitions are shown below:

(1) Transition trigger to communication stop status

This transition takes place upon completion of initialization.

(2) Transition trigger to stop status
This transition takes place upon an initialization failure, power ON, or any abnormal occurrence.

2.5.4 Communication stop status

In the communication stop status, an operation start request from the Communication Middleware is awaited after completion of lower-layer communication software initialization. This section outlines the process to be performed upon a state transition, describes the individual lower-layer communication interface services acceptable during the communication stop status, and gives an overview of the associated process.

(1) Trigger and action
Waits for an individual lower-layer interface service.
(2) Status acquisition service (LowGetStatus)
Returns LOW_STS_INIT as the status.

The triggers for state transition are as follows:

(1) Transition trigger to normal operation state
The transition is caused by the operation start instruction service (LowRequestRun).
(2) Transition trigger to initialize processing state
The transition is caused by the initialization service (LowStart, LowInit, LowInitAll).
(3) Transition trigger to stop status
This transition is caused by the stop service (LowHalt).
(4) Lower-layer communication software type acquisition service (LowGetDevID)
Returns the lower-layer communication software type.

2.5.5 Normal operation status

The normal operation status signifies a state in which data is transmitted to or received from a transmission medium as the primary function of the lower-layer communication software. An outline of the processing immediately after state transition and an outline of the individual lower-layer communication interface services that the normal operation status receives, and related processing, are described below.

(1) Trigger and action
Waits for an individual lower-layer communication interface service.
(2) Status acquisition service (LowGetStatus)
Returns LOW_STS_INIT as the status.

(3) Data transmission service (LowSendData)
Received Protocol Difference Absorption Processor data is divided according to the data size, and each divided data is translated into lower-layer communication software data and then output to the transmission medium.

(4) Data reception service (LowRecvData)
Lower-layer communication software data received from a transmission medium is translated into Protocol Difference Absorption Processor data and then output to the Protocol Difference Absorption Processor.

(5) Lower-layer communication software type acquisition service (LowGetDevID)
Returns the lower-layer communication software type.

(6) Physical address acquisition service (LowGetAddress)
Returns the MAC address.

The triggers for state transition are as follows:

(1) Transition trigger to initialize processing status
This transition is caused by the initialization service (LowStart, LowInit, LowInitAll).

(2) Transition trigger to error stop status
This transition is caused by the error.

(3) Transition trigger to communication stop status
This transition is caused by the end service (LowStop).

(4) Transition trigger to stop status
This transition is caused by the stop service (LowHalt).

2.5.6 Error stop status

The error stop status signifies a state in which operation is stopped by the occurrence of an error. An outline of the processing immediately after state transition and an outline of the individual lower-layer communication interface services that the error stop status receives, and related processing, are described below.

(1) Trigger and action
This transition occurs upon error detection. Error processing will be performed.

(2) Status acquisition service (LowGetStatus)
Returns LOW_STS_ESTOP as the status.

(3) Lower-layer communication software type acquisition service (LowGetDevID)
Returns the lower-layer communication software type.

Triggers for state transitions are shown below:

(1) Transition trigger to initialize processing status

This transition is caused by the initialization service (LowStart, LowInit, LowInitAll).

(2) Transition trigger to normal operation status

This transition is caused by removing the cause of the error. The cause of the house code duplicate reception error is removed by manually resetting the house code.

(3) Transition trigger to stop status

This transition is caused by the stop service (LowHalt).

2.6 P&P Setup of House Code and MAC Address

This section furnishes the information on P&P setup, which is performed from the plug-and-play manager (P&PMng) to assign a house code and MAC address to an ECHONET node newly connected to a power line domain, using the lower-layer communication software's Register_ID functions.

Note that this Standard does not stipulate a manual setup procedure. However, the processing functions provided for ordinary nodes (nodes without the P&PMng function or nodes whose P&PMng function is disabled) by Register_ID shall be implemented. Even when house code/MAC address setup is completed by a manual setup procedure, it shall be possible for P&PMng to perform resetup. In a power line domain having the same house code, however, ECHONET nodes whose house code/MAC address setup is completed by P&P setup shall not be allowed to coexist with ECHONET nodes whose house code/MAC address setup is completed manually.

(A) Man-machine interface requirements

- User operation requirements

  A means of switching to a setup mode shall be provided.

- User operation supplement

  Typical examples for switching to a setup mode are given below:

  P&PMng: Holding down a setup mode switch for several seconds, etc.

  Ordinary nodes: Turning the power ON or pressing a reset switch, etc. when MAC address setup is not completed

- Indicator requirements

  A node type indicator and operation mode indicator shall be provided.

  The node type indicator, however, shall be mandatory only for nodes having the P&PMng function.

  If an LED indicator is used, its color is not stipulated. However, due provision shall be made so that the user can identify the node type indicator, operation mode indicator, and LED indicators stipulated in Part 7, "ECHONET Communications Equipment Specifications", Chapter 3, "ECHONET Device Adapter", Section 3.3.2, "Display Block" (in terms of color, panel, etc.).

  The table below provides definitions of LED indicators:

	On	Off	Blinking
Node type	P&PMng	Ordinary node	
Operation mode	Setup mode	Normal operation	Setup error

If indicators other than LEDs are to be used, they must make it possible for the user to identify the indications as stipulated in the above table.

(B) House code and MAC address in setup mode

- House code (HC) requirements

(C) See under “Individual terminal operations”.

- MAC address requirements

  P&PMng: 0x4000 shall be used.

  However, ordinary nodes shall accept announce address 0 for addresses other than 0x4000.

  In situations where the P&PMng is installed in the local domain, the correct house code is manually set by the installer or is factory set prior to shipment.

  Furthermore, the P&PMng shall be provided beforehand so as to cover the maximum number of nodes to which MAC addresses can be assigned.

  Ordinary nodes: Address 0x40FF, which is reserved for P&P, shall be used as a tentative address when MAC address setup is not completed.

(D) Individual terminal operations

(1) P&PMng operation

- When the user performs a procedure to place the P&PMng in setup mode, it shall transmit announce address 0 at T0 intervals. 0x0000000000000000 shall be used as the house code for announce address 0.
- As a rule, to prevent the traffic from being preoccupied for the announce address 0 frame, the value T0 should be 700 msec. The operation mode indicator shall indicate that the setup process is in progress.
- The P&PMng shall operate for 5 minutes in setup mode. When a predetermined period of time elapses, the P&PMng shall exit setup mode, with the operation mode indicator indicating that a normal operation is being performed.
- An unused MAC address shall be chosen from the locally managed MAC address table (individual addresses: 0x4001 to 0x40EF) and assigned to a received Request_ID frame by means of an ID notification.

  The ID notification shall be transmitted to a tentative address (0x40FF) with a unique terminal identification code attached for identifying a terminal contained in the received Request_ID frame. If the distant party’s address (DA) is a tentative address, the terminal identification code shall be used to determine whether or not to accept the received frame.

(2) Ordinary node operation

- When an ordinary node enters setup mode, it shall wait for the reception of announce address 0. The operation mode indicator shall indicate that the node is in setup mode.

- When a predetermined timeout time elapses, setup mode shall be superseded by normal operation mode. Here, the operation mode indicator shall indicate a setup failure. Although the timeout time is not stipulated, it should be no longer than 3 minutes in order to prevent a neighboring P&PMng from performing improper setup.
- Announce address 0 is received for a period of longer than $10T_0$ (this is called "overhear"). As a result, if announce address 0 is received 10 consecutive times from the P&PMng having the same house code, the house code contained in the announce address 0 frame shall be acquired as the formal house code. The formal house code shall be used to transmit Request_ID, which contains a terminal identification code, to request that the P&PMng assign a MAC address.
- Upon receipt of the ID notification, the formal MAC address shall be acquired. The acquired formal house code and formal MAC address shall be finalized as the local codes with the mode change applied to enter the normal mode. The operation mode indicator shall indicate that a normal operation is being performed. After the P&P setup communication sequence for house code and MAC address is completed, the formal house code and formal MAC address shall be retained in nonvolatile memory.
- If the MAC address indicated by the ID notification is outside the range from 0x4001 to 0x40EF, it shall be rejected.
- If the ID notification is not received after Request_ID transmission, Request_ID shall be transmitted again.
- If announce address 0 is received from more than one P&PMng prior to the receipt of the ID notification, the previously acquired house code shall be discarded with the operation mode indicator indicating a setup failure.

(E) Terminal identification code

As the terminal identification code, the terminal attribute data (e.g., node manufacturer name, terminal type, and magic number "Nmagic" explained in Section 2.4.2 "Layer 2", "(3) Transmission timing") shall be used.

Although the terminal attribute type is not stipulated, an 8-byte value should be used to ensure that the individual terminals are unique.

(F) Communication sequence

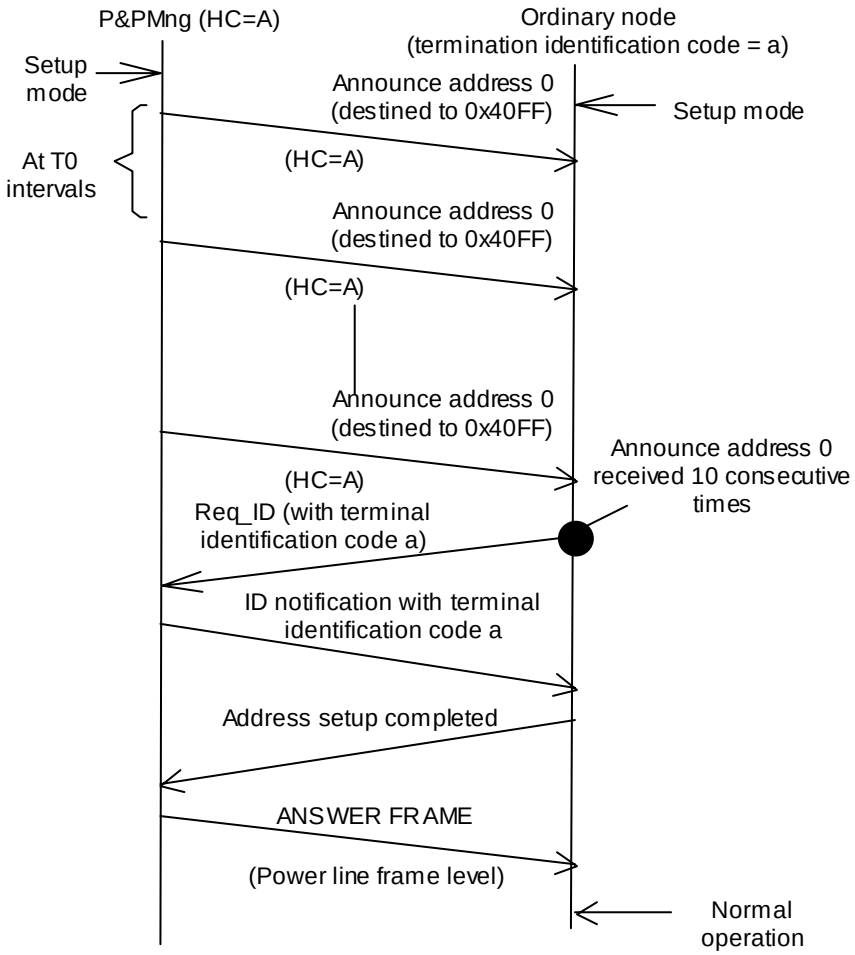


Fig. 2.10 Register_ID Basic Communication Sequence (1)

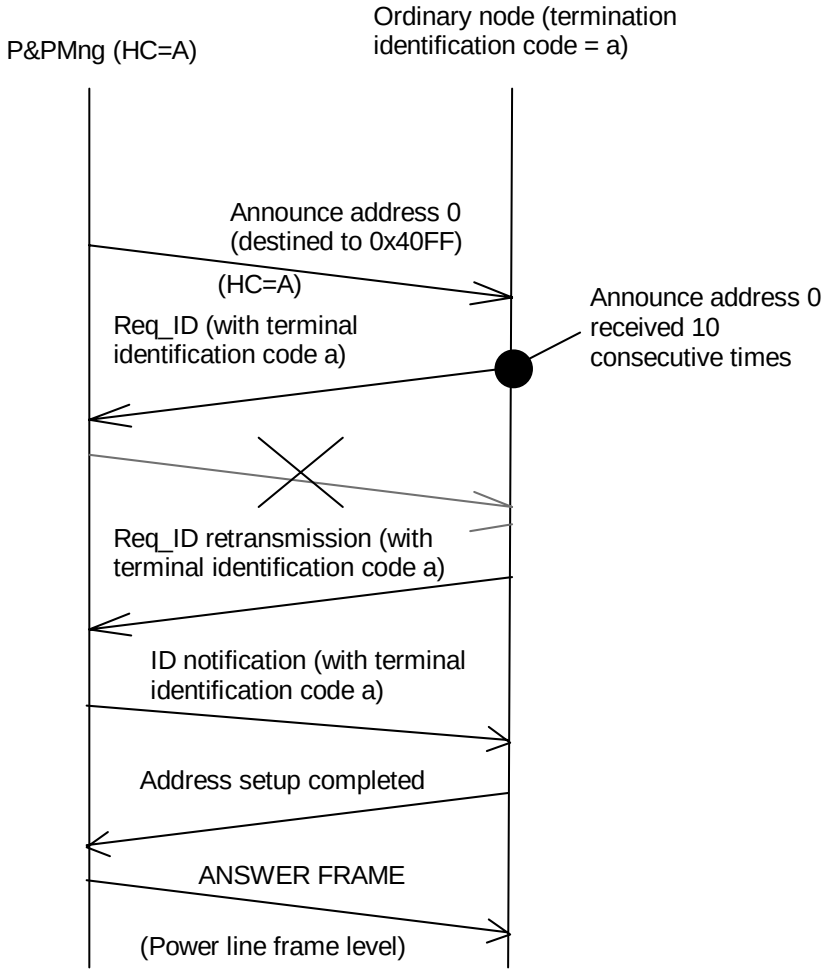


Fig. 2.11 Sequence of Req_ID, ID Notification, and Other Retransmissions

When P&PMng performs the aforementioned announce address 0 operation, "overhear" occurs. As a result, ECHONET Node A shall send a formal address request Req_ID with a terminal identification code to the P&PMng, using the formal house code received by means of announce address 0. The P&PMng assigns formal address 0x4001 to ECHONET Node A as indicated by the terminal identification code. At this stage, however, a tentative MAC address is used for ECHONET Node A so that there is no alternative but to use a tentative address as the destination for formal address assignment command. Therefore, the terminal identification code shall be attached to such a tentative address to prevent other ECHONET nodes with a tentative address from receiving the wrong data.

Here, the ECHONET node having a tentative MAC address shall transmit a formal address request, receive a formal address assignment command only when the attached terminal identification code agrees with its own terminal identification code, and use the assigned formal address as its MAC address.

After the MAC address is replaced with the formal one, the ECHONET node shall use the formal house code and formal address to send an address setup completion notification to the P&PMng. After receiving an ANSWER FRAME that the P&PMng transmits in response to the address setup completion notification, the ECHONET node shall store the formal house code and formal address in nonvolatile memory and terminal normally.

Similarly, the P&PMng shall continue distributing formal addresses to the remaining ECHONET nodes by referencing the formal addresses and individual ECHONET node information provided by the terminal attribute data representing terminal identification codes.

Although the terminal attribute type is not stipulated, an 8-byte value containing the node manufacturer name, terminal type, magic number, and other attribute information should be used to ensure that the individual terminals are unique.

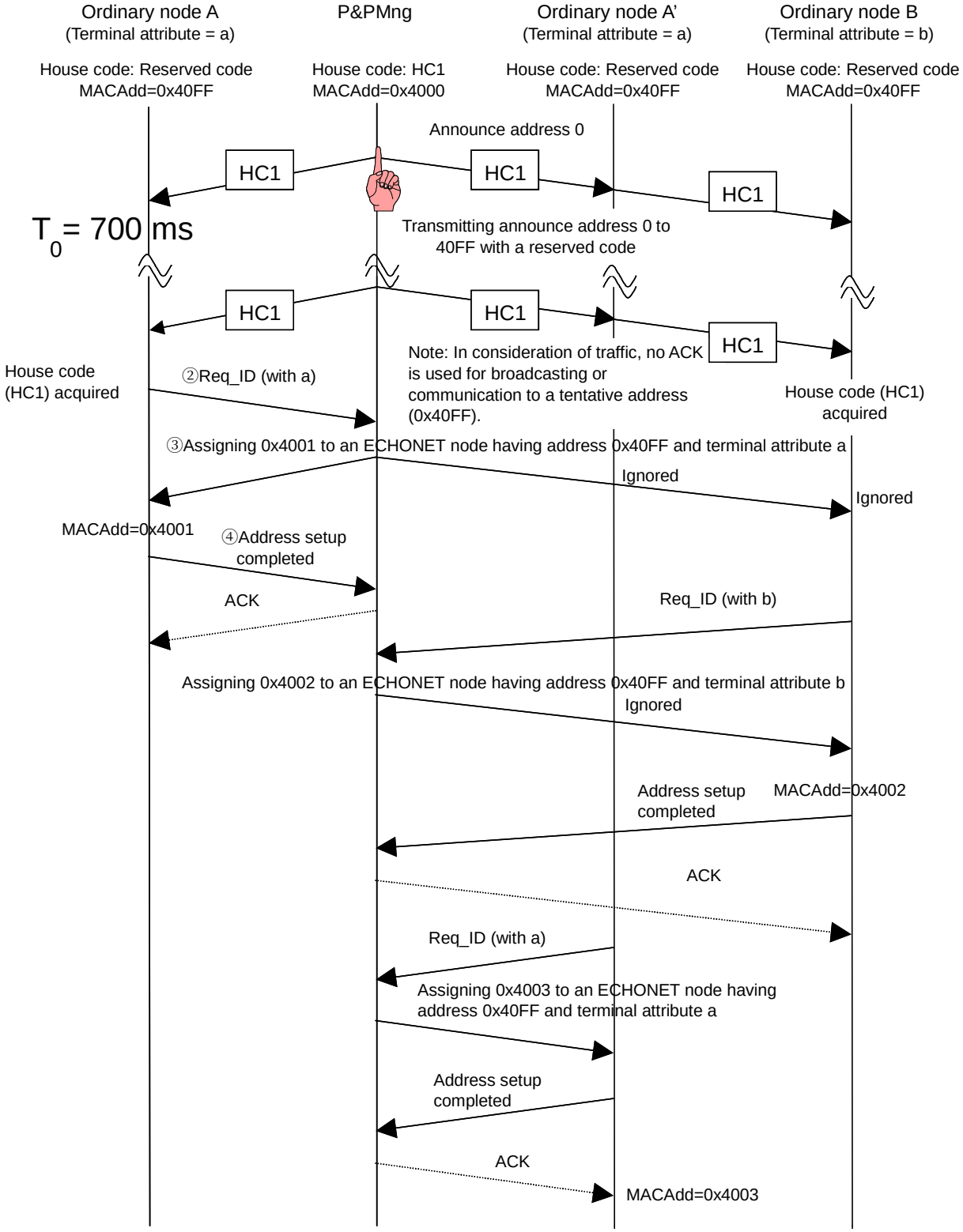


Fig. 2.12 Register_ID Basic Communication Sequence (2)

(F) P&P setup command unique to power line communication protocol

The P&PMng's house code/MAC address assignment process (Register_ID) is executed by the following unique command set. Note that Register_ID is executed by a local command unique to the power line communication protocol, which is differentiated from ECHONET commands by means of a command selector switch (see Supplement 2.1 "Command Set Unique to Power Line Communication Protocol").

For the processing sequence, see the Register_ID processing sequence in Fig. 2.10.

When an ordinary node properly receives 3) ID notification and 4) address setup completion notification, it shall return ACK to the P&PMng with an ANSWER FRAME.

Note that parenthesized numbers in the "STATE/DATA" column below indicate the number of bytes.

Also, note that the P&PMng uses a top-priority slot to transmit commands 1) and 3) below.

The commands shown below are used when an 8-byte magic number is used as a terminal identification code for terminal attribute indication. Although the terminal attribute type is not stipulated by this Standard, an 8-byte value should be used so as to ensure terminal uniqueness.

As shown in Fig. 2.12, the P&PMng uses a top-priority slot (see under "2. Transmission slots" in Section 2.4.2-(3) "Transmission timing"). Therefore, when ECHONET Node A sends Request_ID to the P&PMng, the P&PMng can transmit an ID notification frame prior to ECHONET Node A's Request_ID. Consequently, no confusion results because the same formal address will not be assigned to two or more terminals having the same terminal attribute (e.g., magic number).

1) Announce address 0

ATTRIBUTE		METHOD			STATE/DATA
TS	Property_name	methodtype	action	subtype	DATA
P&P	Housecode	INDICATE	do	normal	Housecode (8)
0x20	0x81	0x04	0x00	0x00	0x0123456789ABCDEF

2) Request ID

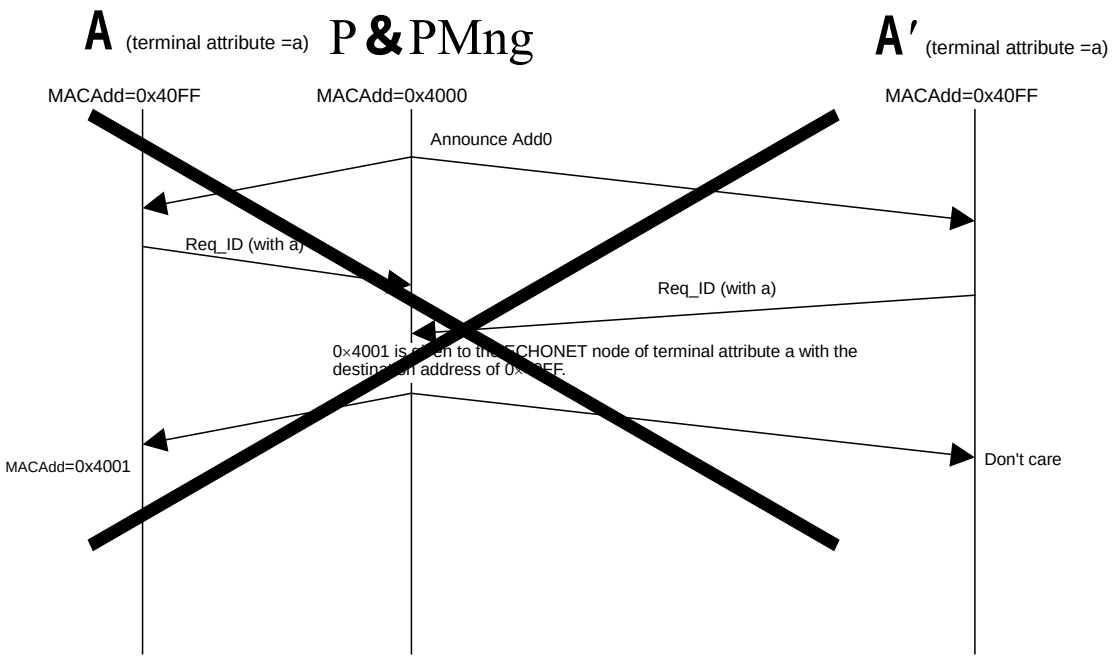
ATTRIBUTE		METHOD			STATE/DATA
TS	Property_name	methodtype	action	subtype	DATA
P&P	MacAddress	INQUIRE	do with housecode	normal	magic_number (8)
0x20	0x01	0x02	0x01	0x00	0x0123

3) ID notification

ATTRIBUTE		METHOD			STATE/DATA
TS	property_name	methodtype	action	subtype	DATA
P&P	MacAddress	WRITE	do with housecode	normal	magic_number (8) MacAddress (2)
0x20	0x01	0x01	0x01	0x00	‘0x0123, 0x4010

4) Address setup completion

ATTRIBUTE		METHOD			STATE/DATA
TS	property_name	methodtype	action	subtype	DATA
P&P	MacAddress	RESPONSE	done	normal	MacAddress (2)
0x20	0x01	0x05	0x10	0x00	0x4010



Transmissions do not take place in the above-indicated order because of slot priority for the P&PMng. Therefore, the same address will not be assigned to nodes having the same attribute. In addition, extended announce address 0 is used for terminal checkout.

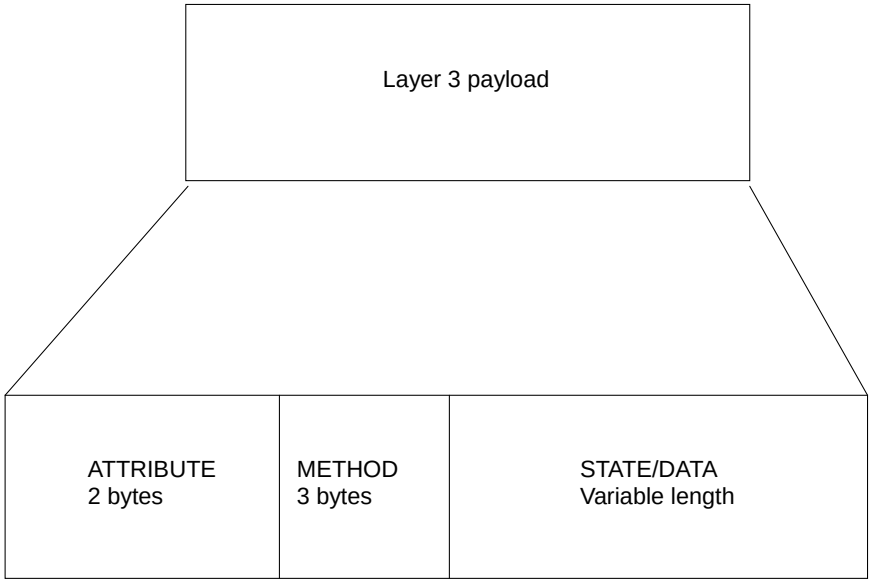
Fig. 2.13 Register_ID Basic Communication Sequence (Nonexistent in Reality)

Supplement 2.1 Command Set Unique to Power Line Communication Protocol

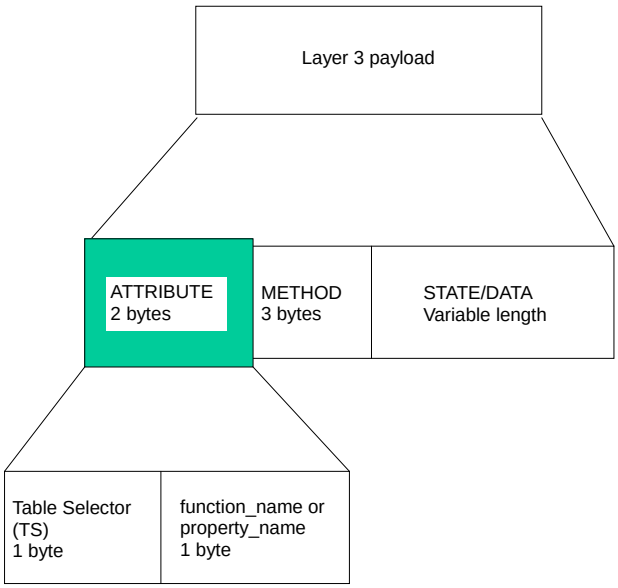
The command set unique to the power line communication protocol is described below for reference. It comprises local commands contained in the Layer 3 payload, except for ECHONET commands.

This command set consists of ATTRIBUTE (2-byte), METHOD (3-byte), and STATE/DATA (variable-length). ATTRIBUTE specifies the control target. METHOD specifies the process for the ATTRIBUTE-specified target.

Note that the command set executes the plug-and-play manager's MAC address assignment process "Register_ID", which is described under "2.5.3 Initialize processing status" in Section 2.5 "Basic Sequence".



ATTRIBUTE consists of one-byte Table Selector (TS) and one-byte function_name or property_name.

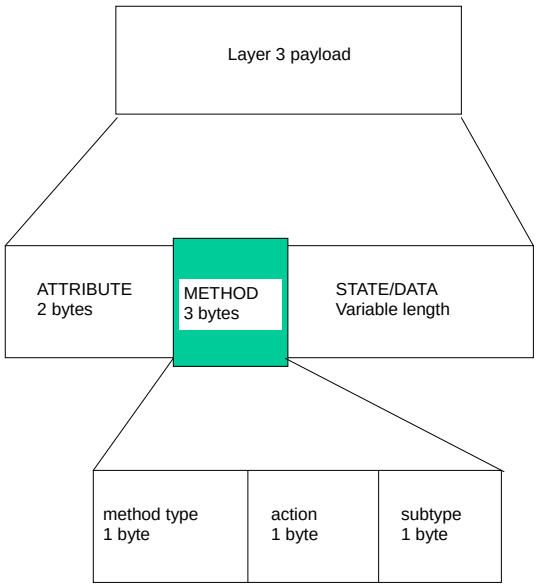


A Table Selector list and property_name for Table Selector = P&P are shown below for reference:

Type	Table Selector	Table Selector value
Plug-and-play	P&P	0×20

property_name	Value
Serial_number	0×00
MacAddress	0×01
Magic_number	0×02
Seed	0×03
maker	0×10
model	0×11
Type	0×12
Type_id	0×13
P&P	0×20

METHOD consists of one byte of method type, one byte of action, and one byte of subtype, as shown below.



A list of commands is provided below for reference.

Method type	Contents	Remarks	Value
READ	Read		0×00
WRITE	Write		0×01
INQUIRE	Request		0×02
RESET	Cancel a request		0×03
INDICATE	Indicate		0×04
RESPONSE	Response		0×05
MAKE	Add an item	Optional	0×06
REMOVE	Delete an item	Optional	0X07
OPEN	Start connection	Optional	0×10
CLOSE	End connection	Optional	0×11

Action	Contents	Remarks	Value
Do	Request for execution	To be used for transmission	0×00
Do with housecode	Request for execution	To be used for transmission	0×01
do with certification method	Request for execution	To be used for transmission	0×02
done	Complete execution	To be used for response	0×10
cannot	Cannot execute	To be used for response	0×20
busy	Cannot execute now	To be used for response	0×21
classified	Cannot execute (not qualified)	To be used for response	0×22

Subtype	Contents	Remarks	Value
normal	Normal		0×00
with certification	With certification		0×01
with encryption	With certification or encryption		0×02

Supplement 2.2 Determination of P&PMng (Set_P&PMng)

A neighboring P&PMng can presume to be the local P&PMng as a result of leakage of its declaration. Therefore, the method allowing the last P&PMng declarer to remain the final P&PMng is no longer adopted.

The installer shall have the option of effecting a P&PMng changeover.

Supplement 2.3 Extended announce address 0

This process is performed to guarantee that a unique MAC address in the subnet is given properly to an ECHONET node.

The P&PMng checks periodically to see if the ECHONET node holding the MAC address that it assigned exits correctly on the power line. This action is also meant to ensure that communication has not been disabled (despite an existing ECHONET node) by proximity or distance, noise, or power line distortion. Sometimes the ECHONET node will have been removed and will not exist. This operation is called extended announce address 0. The ECHONET node check method is not specified, but a proper command is sent out to the corresponding device and its response is checked.

Because of extended announce address 0, the MAC address of a device that does not exist is deleted from the P&PMng registered address list and may be returned to the unregistered address list.

This increases the number of MAC addresses that the P&PMng can assign. The method for deleting a MAC address is not specified in this Standard.

Chapter 3 Low-Power Radio Communication Protocol Specifications

3.1 System Overview

We now consider a low-power radio communications system using 400 MHz band radio waves and conforming to laws and the ARIB Standard.

Low-power radio waves reach a range of several meters to tens of meters, so radio interference is likely to occur among multiple radio systems and among radio devices. These specifications take the following three points into consideration with respect to radio interference:

(1) Radio interference among multiple radio systems
- Use a different channel for each radio system.
- Identify the radio system of the opposite party by using a different identification code (radio system identification code) for each radio system.

(2) Radio interference among multiple devices in the same radio system
- Use multiple channels in one radio system.
- Identify the device of the opposite party by using a different identification code (device identification code) for each device.

(3) Radio interference with radio signals other than the ECHONET radio
- Distinguish ECHONET Standard radio signals from other radio signals at an early stage using the frame synchronization signal.

3.1.1 Communication model

(1) Form

1:1 communication or 1:N or N:M communication and one-way, simplex or broadcast communication.

(2) Number of terminals

Several tens of terminals (approx.) per radio system

(3) Communication volume

One-time transmission data volume: Several tens of bytes (approx.)

(4) Transmission rate
Several kbps

(5) Transmission time
One-time transmission time: Several seconds to several tens of seconds

(6) Number of radio systems in which radio interference is supposed
Approx. 100 systems

3.1.2 ARIB Standard

The ARIB Standards include some standards by use. The low-power radio communication protocol takes into account RCR ARIB STD-T67 (STD-T67) for telemeter/telecontrol and ARIB STD-30 (STD-30) for security.

3.2 Mechanical/Physical Characteristics

This Standard adopts STD-T67 and STD-30.

3.3 Electrical Characteristics

This Standard adopts STD-T67 and STD-30.

3.3.1 Transmission system and transmitting signal

(1) Radio wave type

FID (Frequency modulation without using any sub-carrier and transmission information of data transmission, telemetering or remote indication)

(2) Communication system

One-way system, simplex system, or broadcast communication system (for the same radio system only)

(3) Antenna power

10 mW or less

(4) Modulation system

Binary FSK (Frequency Shift Keying) modulation system by direct modulation

(5) Modulation rate

2400 bps (mandatory) or 4800 bps (optional) ±100 ppm

±100ppm (The rate shall be kept at 2400bps during registration.)

(6) Modulation degree

2.1 kHz ±0.4 kHz

(7) Code type

NRZ (Non-return-to-zero) coding

3.3.2 Frequency

(1) Operating frequency

At least one of the following frequency bands shall be used:

- 429 MHz band: 46 channels of 429.1750 to 429.7375 MHz (12.5 kHz spacing)
- 426 MHz band: 48 channels of 426.2500 to 426.8375 MHz (12.5 kHz spacing)

(2) Communication channel

- The operating frequency channels are divided into multiple channel groups and different communication channel groups are assigned to individual radio systems so that multiple radio communication systems may communicate smoothly.
- The number of channels available to one radio system shall be either 3 or 5, according to the frequency of radio system communications, so that multiple devices in the same system may perform numerous communications smoothly.

(A) 429 MHz band

The STD-T67 is provided with 46 channels and divides them into two parts.

- 1 to 6 ch: Intermittent communications zone specifying a transmission time limit (40 sec. or less) and transmission pause time (2 sec. or more)
- 7 to 46 ch: Continuous communications zone without any time limit

This protocol treats both the continuous communications zone and intermittent communications zone in the same way and provides a transmission time limit and a transmission pause time for use. Accordingly, this Standard uses Channels 1 to 46 at a one-time transmission time of 40 seconds or less and a transmission interval of 2 seconds or more.

The number of channels to be used in one radio system varies with communication frequency. When communication frequency is low, 3 channels (basic channels) are used. For more frequent communication, 5 channels (basic channels + additional channels) are used.

For individual radio systems, communication channel groups A to F and communication channels for system setup are assigned as shown below. In addition to communication channel groups A to F, a channel is prepared and used for radio system setup.

Communication channel group	Channels used	
	STD-T67 intermittent communications zone	STD-T67 continuous communications zone
Group A	1 ch	8 ch, 20 ch
Group B	2 ch	14 ch, 29 ch
Group C	3 ch	10 ch, 22 ch
Group D	4 ch	16 ch, 31 ch
Group E	5 ch	12ch, 24 ch
Group F	6 ch	18 ch, 33 ch
Group G	1 ch	7 ch, 45 ch
Group H	2 ch	13 ch, 39 ch
Group I	3 ch	9 ch, 43 ch
Group J	4 ch	15 ch, 37 ch

Group K	5 ch	11 ch, 41 ch
Group L	6 ch	17 ch, 35 ch
Group M	1 ch	19 ch, 44 ch
Group N	2 ch	25 ch, 38 ch
Group O	3 ch	21 ch, 42 ch
Group P	4 ch	27 ch, 34 ch
Group Q	5 ch	23 ch, 40 ch
Group R	6 ch	30 ch, 36 ch
Group S	1 ch	28 ch, 32 ch
For system setup	6 ch	26 ch, 46 ch

In determining the above communication channel group assignment, consideration was given to the following basic rules:

(1) The 3 channels should always include one intermittent communications zone channel.

(2) Continuous communications zone channels should not be shared.

・Example: Channels 8 and 20 of Group A should not be used in other groups.

(3) Each communication channel group's continuous communications zone channels should not be located close to each other.

(4) The channels for system setup should also be used as a channel for ECHONET LonTalk® Protocol (low-power radio) group registration that uses the same frequency.

The standard frequency of communication shall be as follows (when the duration of each communication session is 1.5 seconds, the expected number of interference radio systems is 200, and the probability of wait for transmission is 0.1% or less):

・When using 3 channels: Approx. once every 3 minutes or less

The channel priority for communication shall be as follows:

(1) Continuous communication zone channels (e.g. Channels 8 and 20 in the case of Group A)

(2) Intermittent communication zone channels (e.g. Channel 1 in the case of Group A)

In addition, the transmission of the ACK signal shall use the channel that received the last incoming transmission. For resend, the transmitting channel shall be changed each time.

(B) 426 MHz band

The STD-30 is provided with 48 channels. The STD-30 specifies all 48 channels as the intermittent communications zone with a fixed transmission time limit (3 sec. or less) and transmission pause time (2 sec. or more). Accordingly, in this Standard, Channels 1 to 48 shall be used at a one-time transmission time of 3 sec. or less and a transmission interval of 2 sec. or more.

The number of channels to be used in one radio system varies depending on the frequency of communication. For low communication frequencies, 3 channels (basic channels) are used. For more frequent communication, 5 channels (basic channels + additional channels) are used.

For individual radio systems, communication channel groups A to O and communication channels for system setup are assigned as shown below. Apart from communication channel groups A to O, a channel is prepared and used for radio system setup.

Communication channel group	Channels used
Group A	1 ch, 17 ch, 34 ch
Group B	2 ch, 19 ch, 37 ch
Group C	3 ch, 21 ch, 40 ch
Group D	4 ch, 23 ch, 43 ch
Group E	5 ch, 25 ch, 46 ch
Group F	6 ch, 27 ch, 33 ch
Group G	7 ch, 29 ch, 36 ch
Group H	8 ch, 31 ch, 39 ch
Group I	9 ch, 18 ch, 42 ch
Group J	10 ch, 20 ch, 45 ch
Group K	11ch, 22 ch, 48 ch
Group L	12ch, 24 ch, 35 ch
Group M	13 ch, 26 ch, 38 ch
Group N	14 ch, 28 ch, 41 ch
Group O	15 ch, 30 ch, 44 ch
For system setup	16 ch, 32 ch, 47 ch

In determining the above communication channel group assignment, consideration was given to the following basic rules:

(1) A total of 48 channels should be used by the communication channel groups of 3 channels.

→ The total number of communication channel groups is 16.

(2) Channels should not be shared with other communication channel groups.
- Example: Channels 1, 17 and 34 of Group A should not be used in other groups.

(3) Adjacent channels should not be concentrated in one communication channel group.
- Example: Channel 1 of Group A is adjacent to Channel 2 of Group B, Channel 17 of Group A is adjacent to Channel 18 Group I and Channel 34 of Group A is adjacent to Channel 33 of Groups F and Channel 35 of Group L.

(4) Channels should not be shared with other communication channel groups.
- Example: Channels 1, 17 and 34 of Group A should not be used in other groups as one of the 3 channels.

The standard frequency of communication shall be as follows (when the duration of each communication session is 1.5 seconds, the expected number of interference radio systems is 200, and the probability of wait for transmission is 0.1% or less):
- When using 3 channels: Approx. once every 70 seconds or less.

The transmission of the ACK signal shall use the channel that received the last incoming transmission (e.g. data transmission signal, ACK signal). For resend, the transmitting channel shall be changed each time.

(C) Setting the frequencies to be used and the communication channel groups

During the initial setting process for the radio system, the frequencies to be used and the communication channel groups shall be set and stored in a non-volatile RAM, etc.

The frequencies to be used shall be set based on how the radio system will be used and the frequency characteristics of the radio transmitter and receiver circuits.

It is desirable that the communication channel groups be coordinated with nearby radio systems, but it is permissible that a default method be used whereby the communication channel groups are determined using the radio system identification code, whose value differs between radio systems.

When setting the communication channel groups using radio communication, system setup channels shall be used for communication. The setting of the communication channel groups using wire-based telecommunication, as well as the setting of the individual settings on a device-by-device basis using switches, etc., is also permitted.

Master nodes that correspond to communication channel groups A to S shall have a

function that allows them to change to communication channel groups A to F defined in Ver. 3.2* and earlier versions of the ECHONET Specification.

3.4 Logical Specifications

(1) Transmission control system

CSMA (Carrier Sense Multiple Access) system using multi-channel

(2) Carrier sense

When a radio wave of another radio system is detected by executing carrier sense before transmission, no transmission is performed. A shift is made to another communication channel and carrier sense is executed, after which transmission is performed. Or transmission is performed after completion of another radio wave. However, if STD-30 is complied with, carrier sense execution is not required.

(3) Transmission time limit and transmission pause time

- Transmission time limit: 429 MHz band 40 sec. max.; 426 MHz band 3 sec. max.
- Transmission pause time: 2 sec. or more

(4) Wait for reception

Not only a continuous wait for reception (mandatory), which gives priority to communication efficiency, but also an intermittent wait for reception (optional), which gives priority to low power consumption during standby for reception, can be provided. These are specified as follows in consideration of the transmission time limit:

- Continuous wait for reception: Available for both 429 MHz band and 426 MHz band
- Intermittent wait for reception:

  429 MHz band; 6 types of 0.5 sec., 3 sec., 5 sec., 15 sec., 25 sec., and 35 sec. intervals

  426 MHz band; 4 types of 0.5 sec., 1 sec., 1.5 sec., and 2 sec. intervals

An intermittent cycle shall be set and registered for each communication destination at the initial settings of devices so that the interval of wait for reception (intermittent cycle) may be differentiated for each device. The transmitting side switches part of the transmit signal in accordance with the intermittent cycle of the communication destination and then transmits it (See Section 3.4.2.).

This Standard provides a communication procedure to boost communication efficiency by switching over to continuous wait for reception only at communication in devices performing an intermittent wait for reception. (See Section 3.4.4.)

3.4.1 Data structure

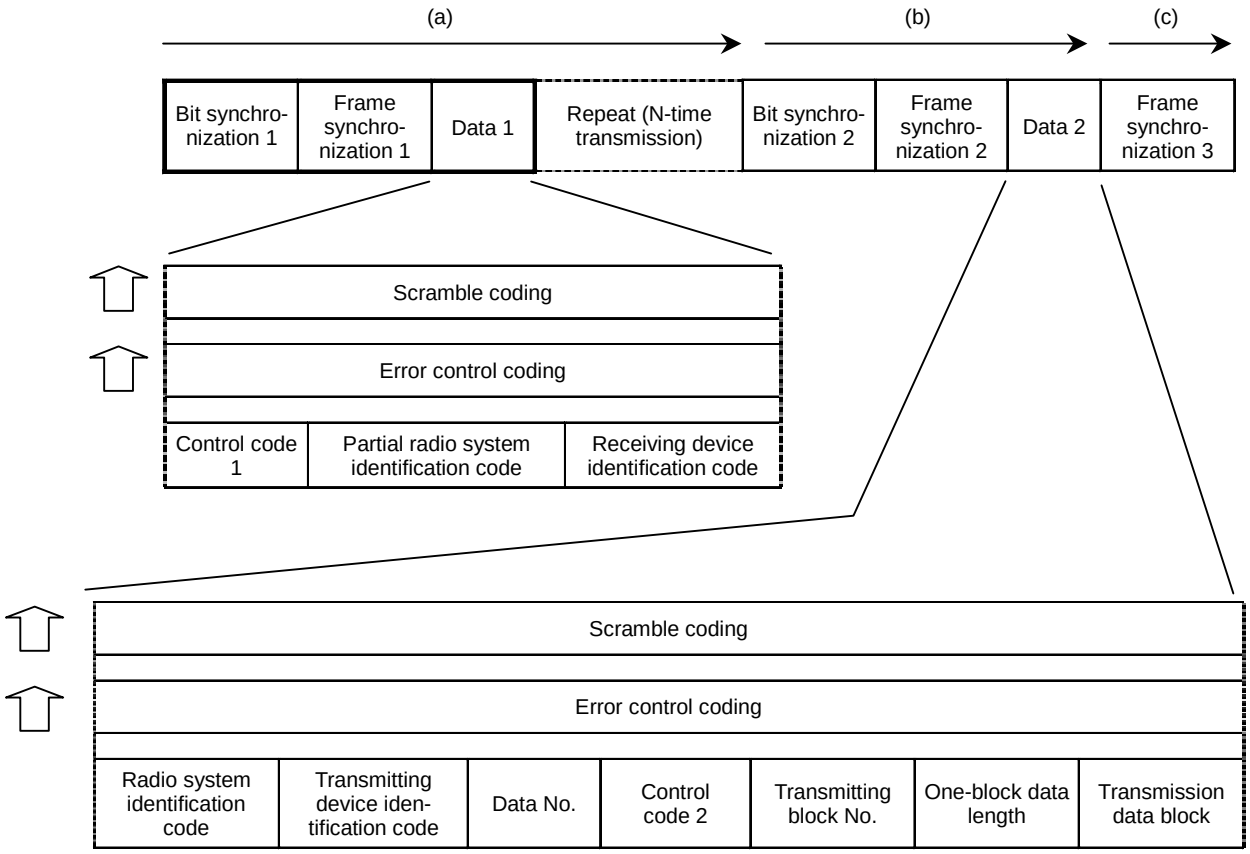
The data format to be used for radio communication is divided into the information transmitting signal (for information transmission) and the ACK signal, a response acknowledging reception. The data format for these shall be as common as possible.

The data format consists of the three elements shown below. As described later, data 1 and data 2 undergo error control coding and then scramble coding. When each byte is transmitted, it is output starting with the high-order bit (MSB).

(a) Repeat division specifying bit synchronization 2, frame synchronization 1, and data 1 as one unit. After synchronization establishment of bit synchronization and frame synchronization 1, the receiving side confirms the opposite party with data 1.

(b) This division consists of bit synchronization 1, frame synchronization 2, and data 2. After synchronization establishment of bit synchronization 2 and frame synchronization 2, the receiving side confirms the opposite party of communication with data 2 and receives the data to be transmitted.

(c) Repeat division of frame synchronization 3. When the ACK signal is transmitted from the receiving side, the transmitting side secures a communication channel.



3.4.2 Layer 1

Bit synchro-nization 1	Frame synchro-nization 1	Data 1	Repeat (N-time transmission)	Bit synchro-nization 2	Frame synchro-nization 2	Data 2	Frame synchro-nization 3

(A) Bit synchronization 1

- A signal for synchronizing the bit timing of the receiving side with the bit timing of the transmitting side in order to receive frame synchronization 1 to data 1.
- 41 bits of "01010101010101010101010101010101010101010".

However, the size of the first-time Bit synchronization 1 may be increased by the following number of bytes at the maximum:

In the case of 2400 bps: 4 bytes (total: 73 bits)

In the case of 4800 bps: 8 bytes (total: 105 bits)

(B) Frame synchronization 1

- A signal for confirming the data format position on the receiving side.
- 31-bit M series code "11100100010101111011010011000000".

(C) Data 1

- Includes the information of the opposite party of communication (details are provided in Section 3.4.3).
- Error control coding for error detection/correction is performed (details are provided in Section 3.4.3), followed by scramble code conversion for limiting the number of continuous same bits (details are provided below in item (2) Scramble code conversion).
- 64 bits (after coding)

(D) Bit synchronization 2

- A signal for synchronizing the bit timing of the receiving side with the bit timing of the transmitting side in order to receive frame synchronization 2 to data 2.
- 65 bits of "1010 ... 01".

(E) Frame synchronization 2

- A signal for confirming the data format position on the receiving side.
- To distinguish frame synchronization 2 from frame synchronization 1, this shall be a bit-inverted code.
- 31-bit M series code "00011011101010000100101100111111".
-

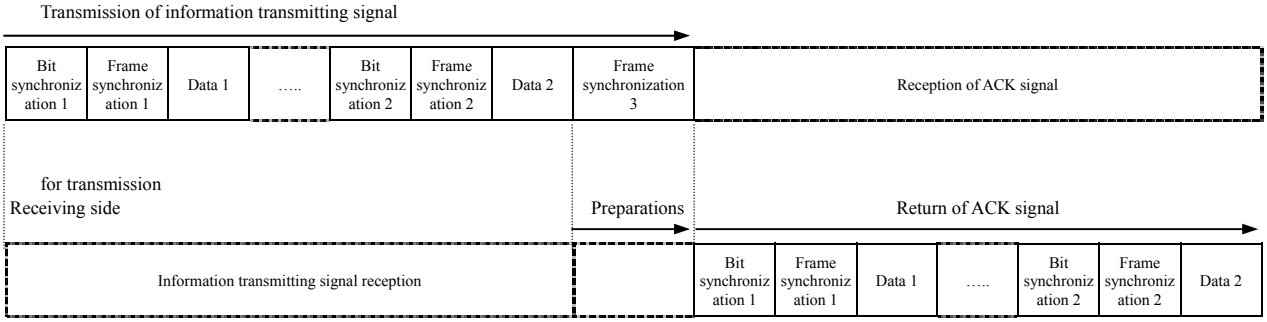
(F) Data 2

- Includes information about the opposite party of communication, data to be transmitted, etc. (details are provided in Section 3.4.3).
- Error control coding for error detection/correction is performed (details are provided in Section 3.4.3), followed by scramble code conversion for limiting the number of continuous same bits (details are provided below in item (2) Scramble code conversion).
- If the number of bits of data 2 including the error control code exceeds 2240, bit synchronization 2 and frame synchronization 2 are inserted halfway so that re-synchronization may be performed on the receiving side (details are provided below in item (3) Block division of data 2).

(G) Frame synchronization 3

- When a return of the ACK signal is required, this signal is used to secure the communication channel already used on the transmitting side while the receiving side makes preparations for transmitting the ACK signal.
- When a return of the ACK signal is not required, the transmitting side does not add frame synchronization 3 to the information transmitting signal.
- The ACK signal is returned when the receiving side has attained normal reception. The ACK signal returning side receives data 2, makes preparations for transmitting the ACK signal, checks the completion of frame synchronization 3 by carrier sense, and transmits the ACK signal.
- Only when data 2 is divided into blocks for transmission does the receiving side make a request to resend by transmitting the ACK signal, even if a receive error occurs in the process of data 2 reception of the information transmitting signal (details are provided in Section 3.4.3, item (8)). At this time, the ACK signal returning side calculates the ending time of data 2 on the basis of the transmitting Block No. in received data 2 (details are provided in Section 3.4.3, item (6)), checks the completion of frame synchronization 3 by carrier sense, and transmits the ACK signal.
- Frame synchronization 3 is a 32-bit code repeat signal with “1” added to the beginning of the 31-bit M series code “00011011101010000100101100111111” that is equal to frame synchronization 2.
- The repeat count shall be 4.

Transmitting side



(1) Receiving cycle and repeat transmission count

- The transmitting side transmits the repeat transmission division in accordance with the intermittent cycle of the receiving side. The repeat transmission division is repeated N times specifying bit synchronization 1 - frame synchronization 1 - data 1 as one unit (136 bits).
- The repeat transmission count N takes into consideration the factors shown below. Accordingly, at transmission, the repeat transmission count is selected based on the opposite party of communication.

  Repeat transmission count N ≥ (a) Count required for a wait for intermittent reception + (b) Repeat count required for identifying the opposite party of communication

  (a) Count required for a wait for intermittent reception: Differs depending on the receiving cycle and modulation rate.
  (b) Repeat count required for identifying the opposite party of communication: Differs depending on the number of channels to be used.

- The table below shows the repeat transmission count N values (min. values) and the corresponding repeat transmission part transmission time values for different intermittent communication periods.
- The number of times the ACK transmission is repeated shall be 3 or more.

Intermittent cycle (sec.)	4800 bps		2400 bps	
	Repeat transmission count (times)	Repeat transmission time (sec.)	Repeat transmission count (times)	Repeat transmission time (sec.)
0 (continuous)	6	0.2	6	0.3
0.5	24	0.7	15	0.8
1.0	41	1.2	24	1.3
1.5	59	1.7	32	1.8
2.0	77	2.2	41	2.3
3.0	112	3.2	59	3.3
5.0	182	5.2	94	5.3
15.0	535	15.2	271	15.3
25.0	888	25.2	447	25.3
35.0	1241	35.2	624	35.3

(2) Scramble code conversion

- For data 1 and data 2, data having continuous same bits (for example, 0×00 and 0×FF) is often used. Because tens of bits to hundreds of bits having continuous “0” and “1” are not desirable as radio communication characteristics, scramble conversion is performed for diffusion.
- For scramble code conversion, an XOR (exclusive OR) with a pseudo random code (M series code) is used.
- At transmission, an error control code is added and XOR is performed with the scramble code. At reception, XOR is performed with the scramble code, and then error control is exerted.
- Error detection/correction (details are provided in Section 3.4.3) is performed in units of 32 bits. Accordingly, the scramble code conversion shall be in units of 32 bits.

Example of scramble code conversion

Transmitting side

Data with an error control code (a) → Data that has undergone scramble processing (c) → Transmit processing

(a) Data with an error control code : 00000000000000001111111111111111
(b) Scramble code : 11011010011000001110010001010110
(c) Data that has undergone scramble processing : 11011010011000000001101110101001
(c) = (a) XOR (b)

Transmission → Reception

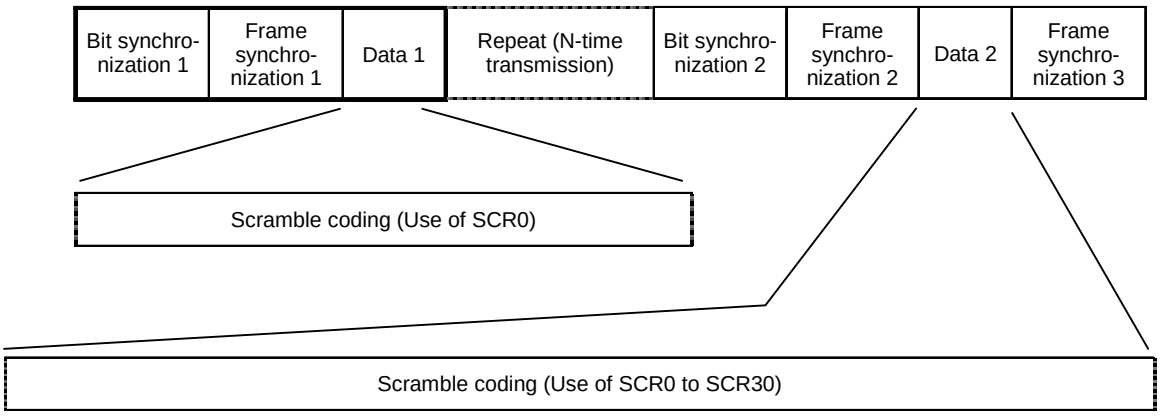
Receiving side

Received data (d) → Data that has undergone scramble processing (f) → Received contents analyze processing

(d) Received data : 11011010011000000001101110101001
(e) Scramble code : 11011010011000001110010001010110
(f) Data that has undergone scramble processing : 00000000000000001111111111111111

(f) = (d) XOR (e) = (a) holds.

- The scramble code shall be of 32 bits or 31-bit pseudo random code + 1 bit “0”, as shown in the table below. There are 31 different scramble codes with different 31-bit pseudo random code divisions.
- Usually, the common scramble code value SCR0 should be used for all radio systems. If necessary, it is permissible that different scramble code values be selected for different radio systems between SCR1 to SCR30. If the scramble code for the reception device and the scramble code for the transmission device are not the same, the confidentiality of the content of the communication is ensured.
- Set a different scramble code for each radio system in the initial settings for the radio system. The default shall be SRC0.
- The data 1 division shall use the scramble code SCR0 and the data 2 division shall use one scramble code selected from the scramble codes SCR0 to SCR30. For communication by the channel for system setup, only SCR0 shall be used as the scramble code.



Scramble Code List

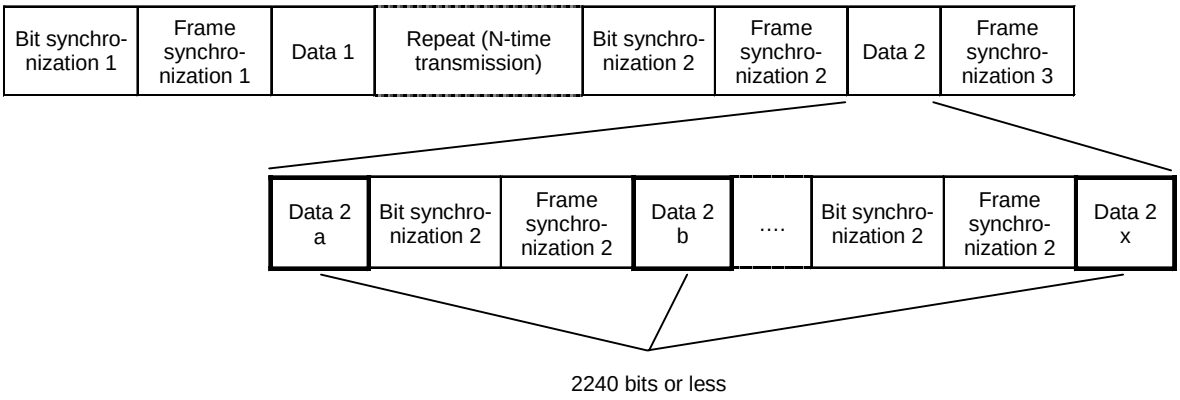
	High-order Lower-layer
SCR0	1101101001100000111001000101011 0
SCR1	1110110100110000011100100010101 0
SCR2	1111011010011000001110010001010 0
SCR3	0111101101001100000111001000101 0
SCR4	1011110110100110000011100100010 0
SCR5	0101111011010011000001110010001 0
SCR6	1010111101101001100000111001000 0
SCR7	0101011110110100110000011100100 0
SCR8	0010101111011010011000001110010 0
SCR9	0001010111101101001100000111001 0
SCR10	1000101011110110100110000011100 0
SCR11	0100010101111011010011000001110 0
SCR12	0010001010111101101001100000111 0
SCR13	1001000101011110110100110000011 0
SCR14	1100100010101111011010011000001 0
SCR15	1110010001010111101101001100000 0
SCR16	0111001000101011110110100110000 0
SCR17	0011100100010101111011010011000 0
SCR18	0001110010001010111101101001100 0
SCR19	0000111001000101011110110100110 0
SCR20	0000011100100010101111011010011 0
SCR21	1000001110010001010111101101001 0
SCR22	1100000111001000101011110110100 0
SCR23	0110000011100100010101111011010 0
SCR24	0011000001110010001010111101101 0
SCR25	1001100000111001000101011110110 0
SCR26	0100110000011100100010101111011 0
SCR27	1010011000001110010001010111101 0
SCR28	1101001100000111001000101011110 0
SCR29	0110100110000011100100010101111 0
SCR30	1011010011000001110010001010111 0

(3) Block division of data 2

When the data to be received is long, the receiving side must perform re-synchronization halfway to correct synchronization error.

If the allowable deviation of the modulation rate is 100 ppm, a single synchronization makes it possible to received data of about 5000 bits.

In consideration of a receiver synchronization error, this Standard provides that data 2 shall be divided into multiple blocks when its length exceeds 2240 bits (this number of bits contains an error control code). Bit synchronization 2 and frame synchronization 2 shall be inserted between blocks.



3.4.3 Layer 2

(1) Radio system identification code

There is a unique identification code for each radio system. Radio communication is not performed between radio systems with a different identification code. For example, there may be multiple radio system identification codes in one house. When the ECHONET subnet differs, the radio system identification code is also different.

An optional node in the subnet is specified as a master node, and the radio system identification code of the master node is specified as the radio system identification code of the subnet. All the slave nodes are standardized to the radio system identification code of the master node.

The radio system identification code shall consist of 48 bits. The 16 highest-order bits shall be the “manufacturer code for the radio system identification code” (hexadecimal code). The remaining 32 bits shall be so managed by manufacturers that duplication is prevented. To use a “manufacturer code for the radio system identification code” value, an application for use must be made to the consortium to have the value approved by the consortium in advance. Before shipping a device having a master node function, the manufacturer shall store the radio system identification code in a nonvolatile RAM, etc. It is permissible that a device be only equipped with master node functions or slave node functions.

When a new slave node is to be added to an existing radio communication system, the master node’s radio system identification code shall be written into the slave node’s nonvolatile RAM, etc. When the radio system identification code has been written, the device identification code is cleared to unset status.

For manual setting, radio communication (via the system setup channel) or wire communication may be used for setting in addition to individual setting using a switch for each device. Items to be stipulated are the radio system identification code, channel count, communication channel group, scramble code (optional), and reception cycle information (optional).

When changing the master node, the radio system identification code written in all the slave nodes within the subnet shall be replaced with the radio system identification code of the new master node.

(2) Device identification code

There is a unique identification code (MAC address) for each device comprising the radio system. In the same radio system, no duplication of device identification codes is allowed.

The device identification code of the master node shall be 0x01. The master node shall be capable of assigning device identification codes between 0x02 and 0x3F. The identification code manager shall assign device identification codes between 0x40 and 0x7F independently of the master node. If device identification code setup is not completed, however, provisional device identification codes between 0x80 and 0xFF

are stored in the slave nodes. For example, the provisional device identification codes may be randomly assigned at the factory prior to shipment.

When the master node assigns device identification codes to the slave nodes, it uses radio communication to perform setup in the following sequences:

(Setup sequence 1)

The following describes the setup sequence to be followed when the requirements (radio system identification code, etc.) for communication are established, except for the device identification code. The transmission speed in the registration mode shall be 2400 bps.

- When a switch, message, or other means is used to issue the "(a) Registration process request" from a higher layer, the slave node's low-power radio unit switches into registration mode.
- The slave node uses a provisional device identification code as the local address to set a radio system setup message flag as indicated under "control code 2" in Section 3.4.3, specifies the master node as the destination, and starts a transmission under the "Transmission data: none" condition ((1) Information transmission). A channel in the ordinary communication channel group is used for transmission.
- Upon receipt of the radioed message above, the master node switches into registration mode, views its domain table, and issues an available device identification code. Further, the master node specifies a provisional device identification code as the destination address, sets a radio system setup message flag, places the issued device identification code in the data section, and transmits it to the slave node ((2) Information transmission). The system setup channel is used for transmission. Reception occurs continuously.

(Data section of data 2 for "(2) Information transmission")

- Upon receipt of the radioed message above, the slave node sets a radio system setup message flag using the device identification code issued as the local address, places the local unit's reception cycle in the data section, and transmits it to the master node ((3) Information transmission). The system setup channel is used for transmission. Reception occurs continuously. Further, the slave node sends the "(c) Registration completion notification" to notify the higher layer of registration process completion and stores the issued device identification code in nonvolatile RAM, etc.

(Data section of data 2 for "(3) Information transmission")

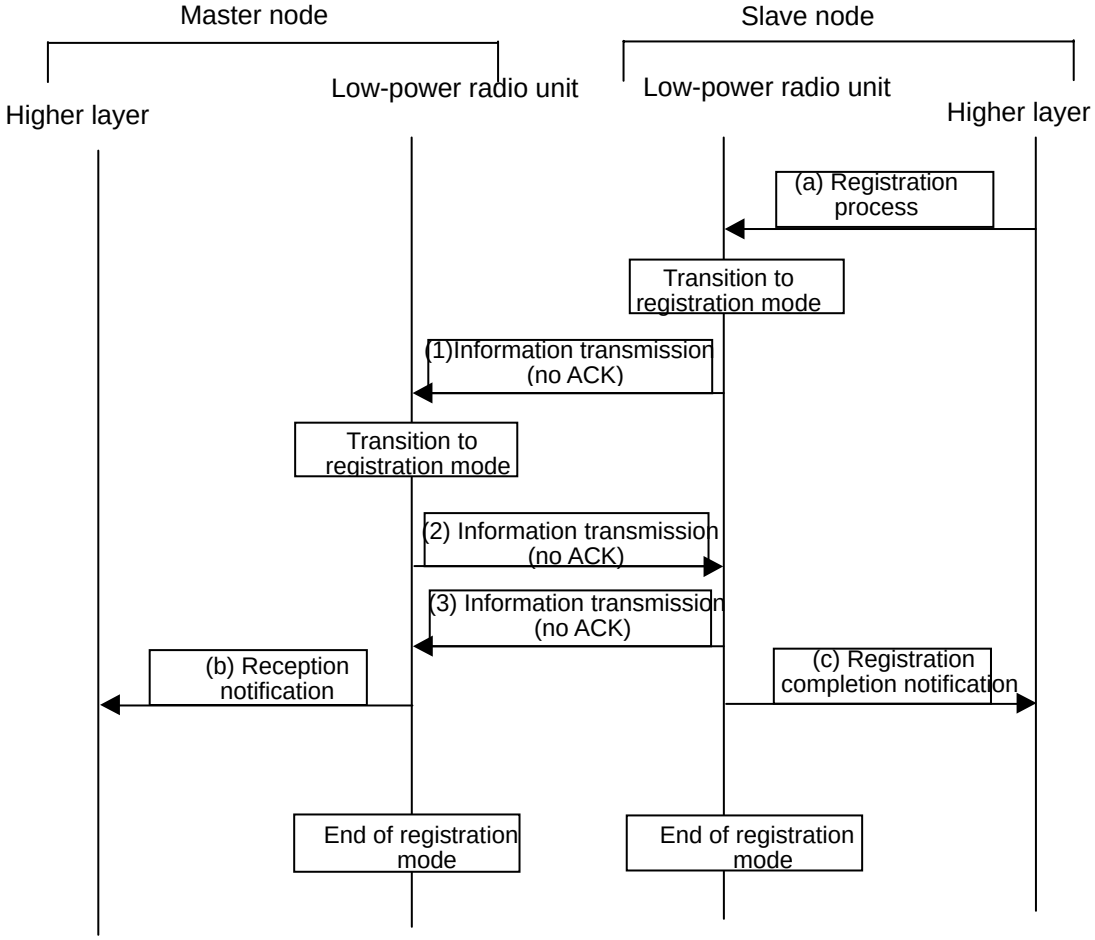
Local unit's reception cycle (8 bits)

- Local unit's reception cycle (3-bit information)

429 MHz	426 MHz	Higher bits Lower bits
Continuous	Continuous	*****000
0.5 sec	0.5 sec	*****001
3 sec	1 sec	*****010
5 sec	1.5 sec	*****011
15 sec	2 sec	*****100
25 sec	2 sec	*****101
35 sec	2 sec	*****110

Upon receipt of the radioed message above, the master node sends the "(b) Reception notification" to notify the higher layer of the slave node's reception cycle.

The sequence diagram is shown below:



(Setup sequence 2)

The following describes the setup sequence to be followed when the requirements (radio system identification code, etc.) for communication are not met (except for the device identification code) or when the slave node does not satisfy the requirements (master node reception cycle and scramble code) for master node communications. Device identification code setup can be performed via setup sequence 1 after changing settings so that the slave node satisfies the requirements for master node communications. The transmission speed in the registration mode shall be 2400 bps.

- Upon receipt of the "(a) Registration process request" from the higher layer, the low-power radio units of the master and slave nodes switch to registration mode.

- The slave node sets a radio system setup message flag using a provisional device identification code as the local address, specifies the master node as the destination, and transmits it under the "Transmission data: none" condition ((1) Information transmission). The system setup channel is used for transmission. Reception occurs continuously. Further, an all-1 radio system identification code is used.
- Upon receipt of the radioed message above, the master node views its domain table and issues an available device identification code. Further, the master node specifies a provisional device identification code as the destination address, sets a radio system setup message flag, places the issued device identification code, radio system identification code, channel count, communication channel group, scramble code, and reception cycle information in the data section, and transmits it to the slave node ((2) Information transmission). The system setup channel is used for transmission. Reception occurs continuously. Further, an all-1 radio system identification code is used.

(Data section of data 2 for "(2) Information transmission")

Device identification code	Radio system identification code	Channel count/communication channel group	Scramble code/reception cycle

Each communication channel group is represented with 5 bits; the sixth bit and the 4 lowest-order bits (The fifth bit is ignored because it is the bit that represents the number of channels).

- Channel count (1-bit information)

	Higher bits Lower bits
3ch	***0****
5ch	***1****

- Scramble code (5-bit information)

	Higher bits Lower bits
SCR0	00000***
SCR1	00001***
:	:
SCR30	11110***

- Communication channel group (4-bit information)

	Higher bits Lower bits
Group A	****0001
Group B	****0010
:	:
Group O	****1111

- Reception cycle (3-bit information)

429 MHz	426 MHz	Higher bits Lower bits
Continuous	Continuous	*****000
0.5 sec	0.5 sec	*****001
3 sec	1 sec	*****010
5 sec	1.5 sec	*****011
15 sec	2 sec	*****100
25 sec	2 sec	*****101
35 sec	2 sec	*****110

- Upon receipt of the radioed message above, the slave node sets a radio system setup message flag using the device identification code issued as the local address, places the local unit's reception cycle in the data section, and transmits it to the master node ((3) Information transmission). The system setup channel is used for transmission. Reception occurs continuously. The radio system identification code received from the master node is used.
- The "(c) Registration completion notification" is transmitted to notify the higher layer that the registration process is completed. At the same time, the device identification code and other information issued by the slave node are stored in nonvolatile RAM, etc.

(Data section of data 2 for "(3) Information transmission")

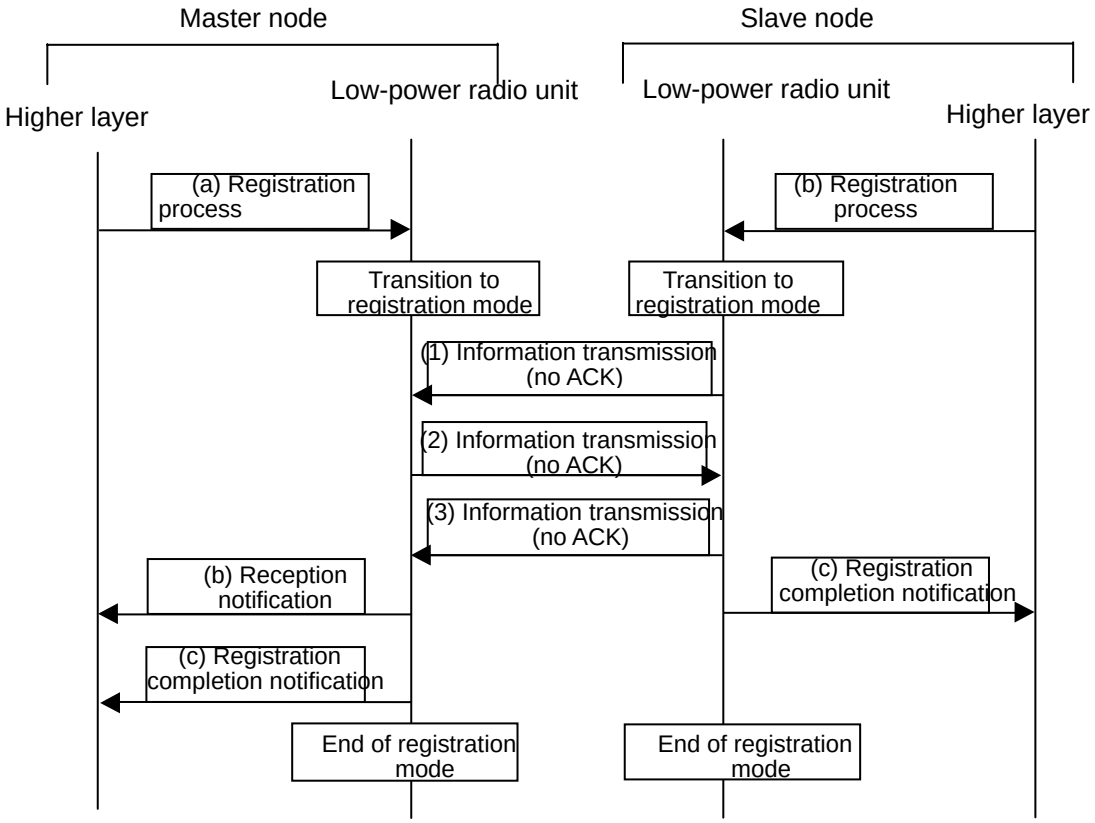
Local unit's reception cycle (8 bits)

- Local unit's reception cycle (3-bit information)

429 MHz	426 MHz	Higher bits Lower bits
Continuous	Continuous	*****000
0.5 sec	0.5 sec	*****001
3 sec	1 sec	*****010
5 sec	1.5 sec	*****011
15 sec	2 sec	*****100
25 sec	2 sec	*****101
35 sec	2 sec	*****110

- Upon receipt of the radioed message above, the master node sends the "(b) Reception notification" to notify the higher layer of the slave node's reception cycle and issues the "(c) Registration completion notification" to notify the higher layer that the registration process has been completed.

The sequence diagram is shown below:



(3) Identification code and broadcast communication or individual communication

The receiver checks that the received radio system identification code matches the self radio system identification code. In addition, the receiver checks that the received device identification code matches the self device identification code. If the codes do not match, reception is suspended.

The identification codes requiring a match check differ depending on each communication method as shown in the following table. The information to specify either broadcast communication or individual communication for the receiver is included in control code 1 in data 1.

Communication method	Radio system identification code (48 bits)	Receiving device identification code (8 bits)
Broadcast communication	Match check	Partial match check
Individual communication	Match check	Match check

Address Setting for Broadcast Communication

Broadcast group	Device identification code on the receiving side	Information on broadcast specifications to be inserted in the receiving device identification code division at transmission
0	**** 0000	**** ***1
	**** 1000	**** ***1
1	**** 0100	**** **1*
	**** 1100	**** **1*
2	**** 0010	**** *1**
	**** 1010	**** *1**
3	**** 0110	**** 1***
	**** 1110	**** 1***
4	**** 0001	***1 ****
	**** 1001	***1 ****
5	**** 0101	**1* ****
	**** 1101	**1* ****
6	**** 0011	*1** ****
	**** 1011	*1** ****
7	**** 0111	1*** ****
	**** 1111	1*** ****
Simultaneous broadcast to all groups		1111 1111

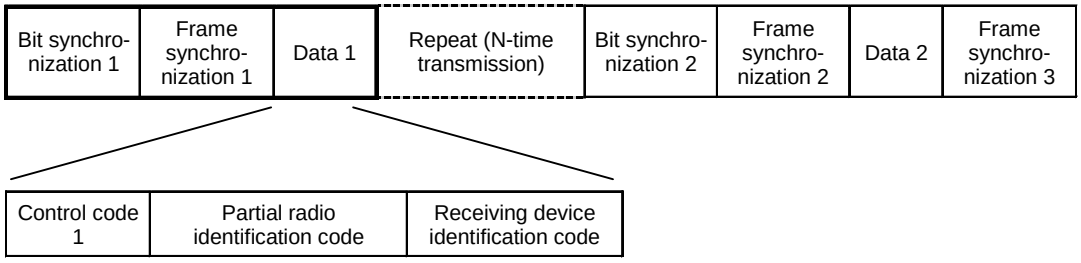
(4) Error detection/correction

- Error control is performed for data 1 and data 2.
- BCH (31, 16) is used for error control. To a 16-bit transmit signal, a 15-bit BHC error control code and a 1-bit even parity code are given.
- Error detection capability: Error detection can be performed for up to 5 bits out of 32 bits.
- Error correction capability: Error correction can be performed for up to 2 bits out of 32 bits.
- When received data corresponds to “error detection available” and “error correction enable”, it is received and error correction is performed. If received data corresponds to “error detection available” and “error correction disable”, reception is suspended.

Transmit signal 16 bits	BCH error control code 15 bits	Even parity code 1 bit

(5) Data 1 structure

- 32-bit information + 32-bit error control code = 64 bits



(A) Control code 1

- 8-bit information
- Provided with the following flags

(1) Flag to indicate broadcast communication or individual communication (2 bits)

	High-order Low-order
- Broadcast communication	: 1 0 ＊＊＊＊＊＊
- Individual communication (information transmitting signal)	: 0 0 ＊＊＊＊＊＊
- Individual communication (ACK signal)	: 0 1 ＊＊＊＊＊＊

(2) Flag to indicate contents of the following partial radio system identification code (2 bits)

16 high-order bits/16 medium-order bits/16 low-order bits of 48 bits of radio system identification code. Detailed usage is described below in item (B) Partial radio system identification code.

- 16 high-order bits	: ＊＊1 1 ＊＊＊＊
- 16 medium-order bits	: ＊＊1 0 ＊＊＊＊
- 16 low-order bits	: ＊＊0 1 ＊＊＊＊

(3) Flag to indicate transmitting Channel No. (3 bits)

The receiving side checks that the received Channel No. matches the actual receiving channel. If the codes do not match, reception is suspended.

- Channel No. : ＊＊＊＊0 0 1 ＊～＊＊＊＊1 0 1 ＊

429 MHz band

Channel number	'001'	'010'	'011'
Communication channel group	Channels used		
Group A	1 ch	8 ch	20 ch
Group B	2 ch	14 ch	29 ch
Group C	3 ch	10 ch	22 ch
Group D	4 ch	16 ch	31 ch
Group E	5 ch	12ch	24 ch
Group F	6 ch	18 ch	33 ch
Group G	1 ch	7 ch	45 ch
Group H	2 ch	13 ch	39 ch
Group I	3 ch	9 ch	43 ch
Group J	4 ch	15 ch	37 ch
Group K	5 ch	11 ch	41 ch
Group L	6 ch	17 ch	35 ch
Group M	1 ch	19 ch	44 ch
Group N	2 ch	25 ch	38 ch
Group O	3 ch	21 ch	42 ch
Group P	4 ch	27 ch	34 ch
Group Q	5 ch	23 ch	40 ch
Group R	6 ch	30 ch	36 ch
Group S	1 ch	28 ch	32 ch
For system setup	6 ch	26 ch	46 ch

426 MHz band

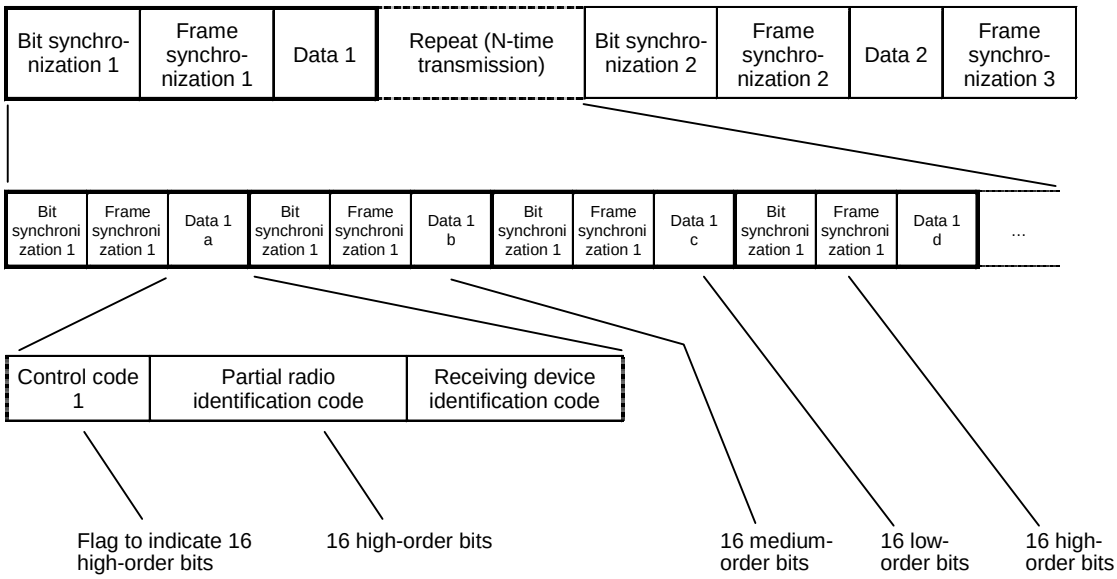
Channel number	'001'	'010'	'011'
Communication channel group	Channels used		
Group A	1 ch	17 ch	34 ch
Group B	2 ch	19 ch	37 ch
Group C	3 ch	21 ch	40 ch
Group D	4 ch	23 ch	43 ch
Group E	5 ch	25 ch	46 ch
Group F	6 ch	27 ch	33 ch
Group G	7 ch	29 ch	36 ch
Group H	8 ch	31 ch	39 ch
Group I	9 ch	18 ch	42 ch
Group J	10 ch	20 ch	45 ch
Group K	11ch	22 ch	48 ch
Group L	12ch	24 ch	35 ch

Group M	13 ch	26 ch	38 ch
Group N	14 ch	28 ch	41 ch
Group O	15 ch	30 ch	44 ch
For system setup	6 ch	32 ch	47 ch

(4) The low-order bit of control code 1 shall be 0.

(B) Partial radio system identification code

- 16-bit information
- The transmitting side transmits 48 bits of radio system identification code divided as follows: 16 high-order bits, 16 medium-order bits, and 16 low-order bits (partial radio system identification code).
- In repeat transmission, transmission is performed by switching between high-order → medium-order → low-order → high-order → etc., (or low-order → medium-order → high-order → low-order → etc.) and the partial radio system identification code.
- With received control code 1, the receiving side judges whether the partial radio system identification code is high-order/medium-order/low-order or not and checks that it matches the corresponding portion of its radio system identification code. If the codes do not match, reception is suspended.

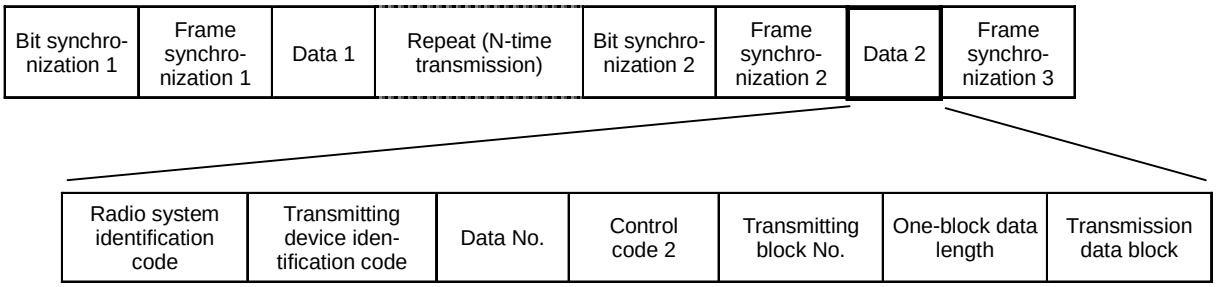


(C) Receiving device identification code

- 8-bit information
- The transmitting side sends the device identification code of the opposite party of communication.
- In individual communication, the receiving side checks that the received receiving device identification code matches the device identification code owned by it. If the codes do not match, reception is suspended.
- In broadcast communication, the receiving side checks that the received reception device identification code matches the 4 lowest-order bits of its own device identification code. If the codes do not match, reception is suspended.

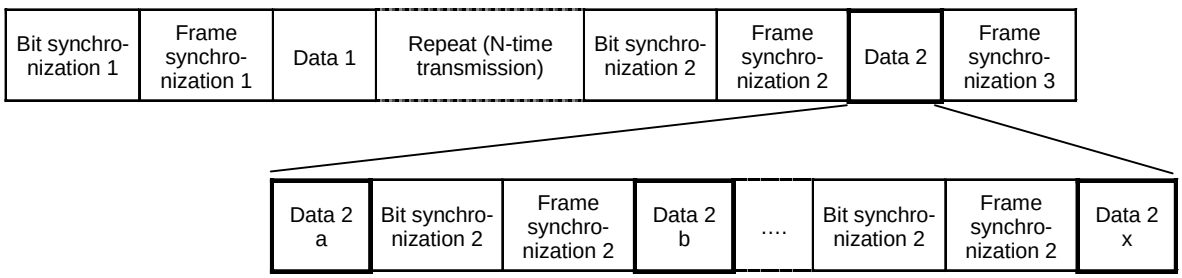
(6) Data 2 structure

- Information of up to 1120 bits + Error control code of up to 1120 bits = 2240 bits max.
- The data structure differs for lump transmission of information transmitting signal, block-divided transmission of information transmitting signal, and ACK signal.
- Lump transmission of information transmitting signal



- Block-divided transmission of information transmitting signal

  When the signal is divided into blocks, data 2 of the beginning block is the same as lump transmission. For the second block and after, the radio system identification code, transmitting device identification code, data No., and control code 2 are omitted.



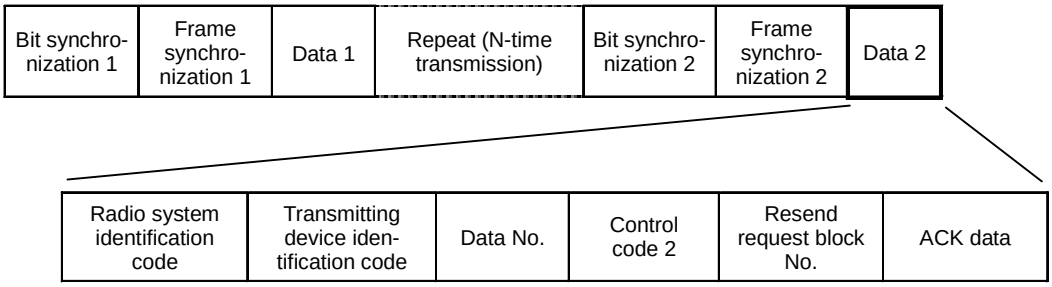
Data 2a

Radio system identification code	Transmitting device identification code	Data No.	Control code 2	Transmitting block No.	One-block data length	Transmission data block

Data 2b to data 2x

Transmitting block No.	One-block data length	Transmission data block

- ACK signal
  Basically, this is the same as lump transmission. The ACK data has a fixed length, so “One-block data length” is omitted. Instead of transmitting Block No., a resend request block is included and used for block-divided transmission.



(A) Radio system identification code

- 48-bit information
- The transmitting side sends its 48 bits of radio system identification code.
- The receiving side checks that the received radio system identification code matches its radio system identification code. If the codes do not match, reception is suspended.

(B) Transmitting device identification code

- 8-bit information
- The transmitting side sends its device identification code.
- The receiving side obtains the information transmitting signal or the identification code of the device that transmitted the ACK signal.

(C) Data No.

- 8-bit information
- The receiving side checks duplicate reception at re-transmission by the data No. If the same data No. is received from the same opposite party of communication in succession, the received data is not notified to the

high-order layer (i.e., higher than the lower-layer communication software in the communication layers).

- The transmitting side changes the data No. each time new transmission data is transmitted. For example, suppose that the previously transmitted or received data number is +1. The data number for re-transmission shall be the same as that previously transmitted.
- When the data No. exceeds 0×FF, it is returned to 0×00.

(D) Control code 2

- 16-bit information
- Provided with the following flags. For detailed usage, see Section 4.4.4.

(1) Flag to indicate whether or not a request for link connection exists in the communication procedure (2 bits)

	High-order Low-order
- Link connection data	: ＊＊＊＊＊＊＊＊＊＊1 0＊＊＊＊＊
- Link disconnection data	: ＊＊＊＊＊＊＊＊＊＊0 1＊＊＊＊＊
- Linking data	: ＊＊＊＊＊＊＊＊＊＊0 0＊＊＊＊＊
- Single-shot data (without link)	: ＊＊＊＊＊＊＊＊＊＊1 1＊＊＊＊＊

(2) Flag to indicate whether or not the request for the return of the ACK signal is made to the receiving side (1 bit)

- Request for ACK signal : ＊＊＊＊＊＊＊＊＊＊＊1＊＊＊＊
- No request for ACK signal : ＊＊＊＊＊＊＊＊＊＊＊0＊＊＊＊

(3) Flag to indicate the content of the information transmission signal (2-bit)

The term “radio system setup messages” used below shall refer to messages used to make confirmations and set settings for the radio section (e.g. registration messages, reception level measurement messages, version information messages).

The term “ECHONET messages” used below shall refer to such messages as information transmission messages, link messages and ACK signal messages.

- The term “vendor messages” used below shall refer to messages that are based on individual vendors’ specifications.

	High-order Low-order
- ECHONET messages:	＊＊＊＊＊＊＊＊＊＊＊＊0 0＊＊
- Radio system setup messages:	＊＊＊＊＊＊＊＊＊＊＊＊0 1＊＊
- Vendor messages:	＊＊＊＊＊＊＊＊＊＊＊＊1 1＊＊

- Reserved: ＊＊＊＊＊＊＊＊＊＊＊＊１０＊＊

(4) Flag to provide information on the transmission speeds at which communication is possible (2-bit)

If 4800 bps is supported, this flag shall be set and sent even if the transmission speed to be used for the transmission is 2400 bps.

	High-order	Low-order
- 2400 bps is supported:	＊＊＊＊＊＊０	０＊＊＊＊＊＊＊＊
- 4800 bps is supported:	＊＊＊＊＊＊０	１＊＊＊＊＊＊＊＊

(5) Flag to indicate that a temporary mode change will be made to the continuous operation mode (1-bit)

If a temporary mode change is going to be made to the continuous reception waiting mode after transmission of the message, this flag shall be set and sent.

	High-order	Low-order
- No mode change is going to be made:	＊＊＊＊＊＊＊＊＊＊＊＊＊＊＊	０＊
- A mode change is going to be made:	＊＊＊＊＊＊＊＊＊＊＊＊＊＊＊	１＊

(6) Flag to provide information on the transmission power output (1-bit)

Indicates the transmission power output for the message to be sent. If the power is 1mW or less, this flag shall be set and sent.

	High-order	Low-order
- 1mW to 10mW:	＊＊＊＊＊＊＊＊	０＊＊＊＊＊＊＊
- 1mW or less:	＊＊＊＊＊＊＊＊	１＊＊＊＊＊＊＊

(7) Flag to provide version information (1-bit)

Provides version information. This flag must be set without exception regardless of the type of the message to be sent.

	High-order	Low-order
- Version 3.2* and earlier versions:	＊＊＊＊	０＊＊＊＊＊＊＊＊＊＊＊＊
- Version 3.30 and later versions:	＊＊＊＊	１＊＊＊＊＊＊＊＊＊＊＊＊

(8) Flag to indicate whether the link connection function is implemented (1-bit)(optional)

Indicates whether the link connection function is implemented. If the link connection function is implemented, this flag must be set without exception regardless of the type of the message to be sent.

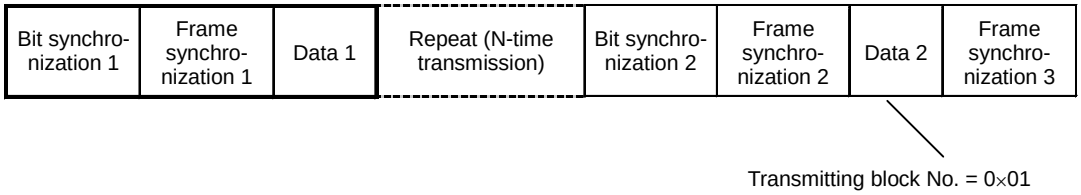
	High-order	Low-order
- Link connection function is not implemented:	＊＊＊＊＊0	＊＊＊＊＊＊＊＊＊＊
- Link connection function is implemented:	＊＊＊＊＊1	＊＊＊＊＊＊＊＊＊＊

(9) All other bits are reserved bits and shall be set to 0.

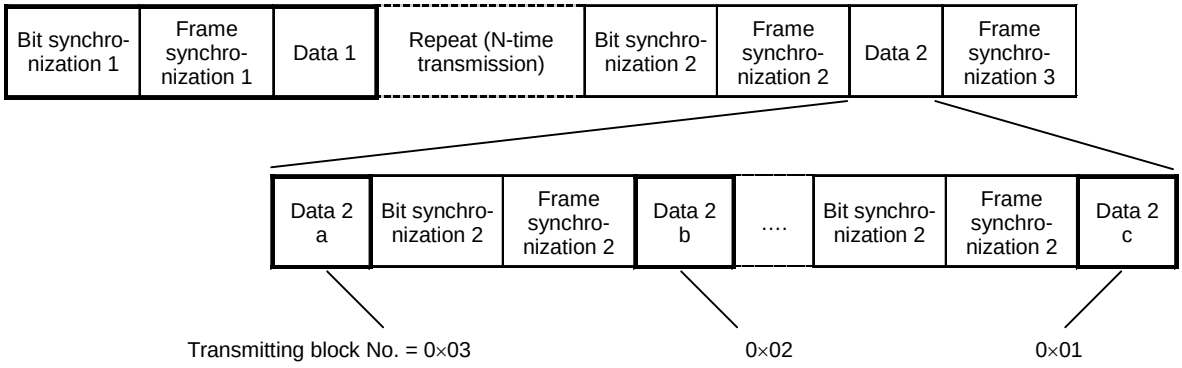
(E) Transmitting Block No.

- 8-bit information
- The transmitting Block No. indicates the number of remaining transmission data blocks to be received.
- 0×01 for lump transmission of data 2
- For block-divided transmission of data 2, the number becomes a decremented value during transmission, as shown in the figure below, and the last transmission data block is 0×01.
- To prevent endless reception, it is desirable that the receiving side check the decremented value of the transmitting Block No. and suspend reception if the check result is not normal.

<Example 1> When transmitting data 2 in a lump transmission



<Example 2> When transmitting data 2 in 3-divided form



(F) One-block data length

- 8-bit information
- The transmission data length (not including the error control code) in the next transmitting data block is included in units of bytes for transmission.
- The data length of one block is 1 byte to 128 bytes.
- When "0×00" is set in a one-block data length, the subsequent transmission data block does not exist.

(G) Transmission data block

- 2048 bits (256 bytes) max.
- The information volume to be transmitted by the high-order layer (higher than the lower-layer communication software in the communication layers) shall be in units of 8 bits, and the transmission data block shall be up to 1024 bits (128 bytes).
- A 16-bit error control code shall be added to each 8-bit transmission data x 2 units (= 16 bits).
- When the transmission data is of an odd number of bytes, [0] of 8 bits is added to the end into 16 bits for transmission. The receiving side determines this based on whether the data length of one block is odd or even.

(H) Resend request Block No.

- 8-bit information
- 0×00 for no receive error.
- In the ACK signal during block-divided transmission, the beginning number of the transmission data block for the resend request shall be sent.

  0×00 for no receive error.
- The receiving side that received the resend request by the ACK signal resends the transmission data blocks subsequent to the resend request Block No. Details are provided below in item (7) Transmitting Block No. and resend request Block No.

(I) ACK data

- 8-bit information
- 0×06 for normal reception. (ACK)
- 0×15 upon occurrence of a receive error at block-divided transmission. (NAK)

(7) Duplication check for radio system identification code and device identification code

To perform a duplication check for the radio system identification code and device identification code in broadcast communication, the receiving side should check that the received transmitting device identification code matches its own device identification code and report it by some means.

(8) Transmitting Block No. and resend request Block No.

Usually, when the received information transmitting signal corresponds to “error detection available” and “error correction disable”, the receiving side does not send back the ACK signal but instead waits for re-transmission from the transmitting side. However, if the transmission data volume is substantial and block-divided transmission must be performed, the following ACK signal shall be transmitted to shorten the data to be resent under “error detection available” and “error correction disable”.

The receiving side of the information transmitting signal sends the resend request Block No. of the data transmitting block together with the resend request Block No. in the ACK signal to be resent to the transmitting side. If no resend request is made, the Block No. is 0×00.

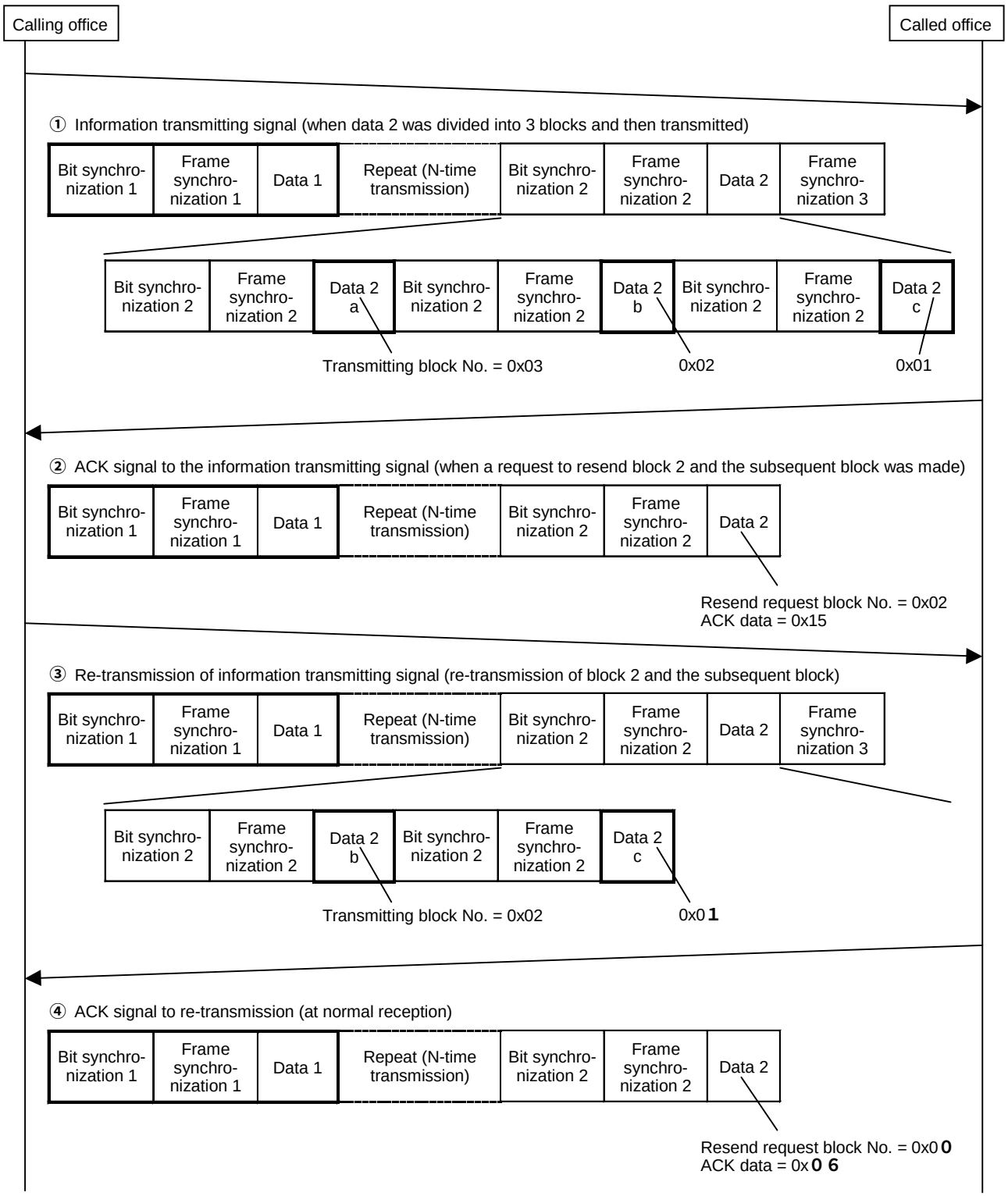
The transmitting side of the information transmitting signal that received this ACK signal resends the data transmitting blocks subsequent to the resent request Block No. in the ACK signal.

The receiving side of the information transmitting signal receives this re-transmission and integrates it with the previous received contents on the basis of the data transmitting Block No.

However, even in block-divided transmission, if the first block corresponds to “error detection available” and “error correction disable”, the receiving side does not send back the ACK signal but instead waits for re-transmission.

Re-transmission of the information transmitting signal to the ACK signal of the resend request is the same as ordinary resend processing. Here, the data No. shall be the same as the data No. that was previously transmitted.

<Example> Data 2 was divided into 3 blocks and then transmitted. A request to resend block 2 and the subsequent block was made. As a result of re-transmission, all 3 blocks have been received.



(9) Communication time and transmission data volume (for reference)

The reference values for the data (transmission data) volume to be transmitted by the high-order layer (higher than the lower-layer communication software in the communication layers) and transmission time are shown below. The transmission time includes the repeat transmission time of bit synchronization 1 to data 1, so that it differs depending on the intermittent cycle and number of channels used on the receiving side.

STD-T67 and STD-30 establish a limit on transmission time, thereby limiting the volume of information that can be sent in a single transmission.

As an example, the transmission time for a case in which the transmission data is 16 bytes and 256 bytes is shown below.

(A) 429 MHz band

Duration of Transmission of Transmission Data (seconds)

Intermittent cycle (sec.)	**4800 bps**		**2400 bps**	
	Duration of transmission of 16-byte data (sec)	Duration of transmission of 256-byte data (sec)	Duration of transmission of 16-byte data (sec)	Duration of transmission of 256-byte data (sec)
0 (continuous)	0.3	0.7	0.5	1.3
0.5	0.7	1.1	1.0	1.8
3.0	3.2	3.6	3.5	4.3
5.0	5.2	5.6	5.5	6.3
15.5	15.2	15.6	15.5	16.3
35.0	25.2	25.6	25.5	26.3

(B) 426 MHz band

Duration of Transmission of Transmission Data (seconds)

Intermittent cycle (sec.)	**4800 bps**		**2400 bps**	
	Duration of transmission of 16-byte data (sec)	Duration of transmission of 256-byte data (sec)	Duration of transmission of 16-byte data (sec)	Duration of transmission of 256-byte data (sec)
0 (continuous)	0.3	0.7	0.5	1.3
0.5	0.7	1.1	1.0	1.8
1.0	1.2	1.6	1.5	2.3
1.5	1.7	21	20	2.8
2.0	2.2	2.6	2.5	***3.3***

Because the oblique font portion exceeds the transmission time limit, 256 bytes cannot be transmitted.

(10) Overall system configuration

In the radio system, the system identification code, communication channel groups, number of channels, and scramble code to be provided by each device shall be common. Using simultaneous broadcast communication enables the devices in the same radio system to receive data simultaneously.

Groups sharing the 3 low-order bits of the device identification code shall be specified as broadcast groups. Using group broadcast communication enables the devices in a specified broadcast group to receive data simultaneously.

For radio devices, the device identification code to be prepared for each device differs with the device. The receiving cycle can be different for each device. Using individual communication enables only a specified device to receive data.

3.4.4 Layer 3

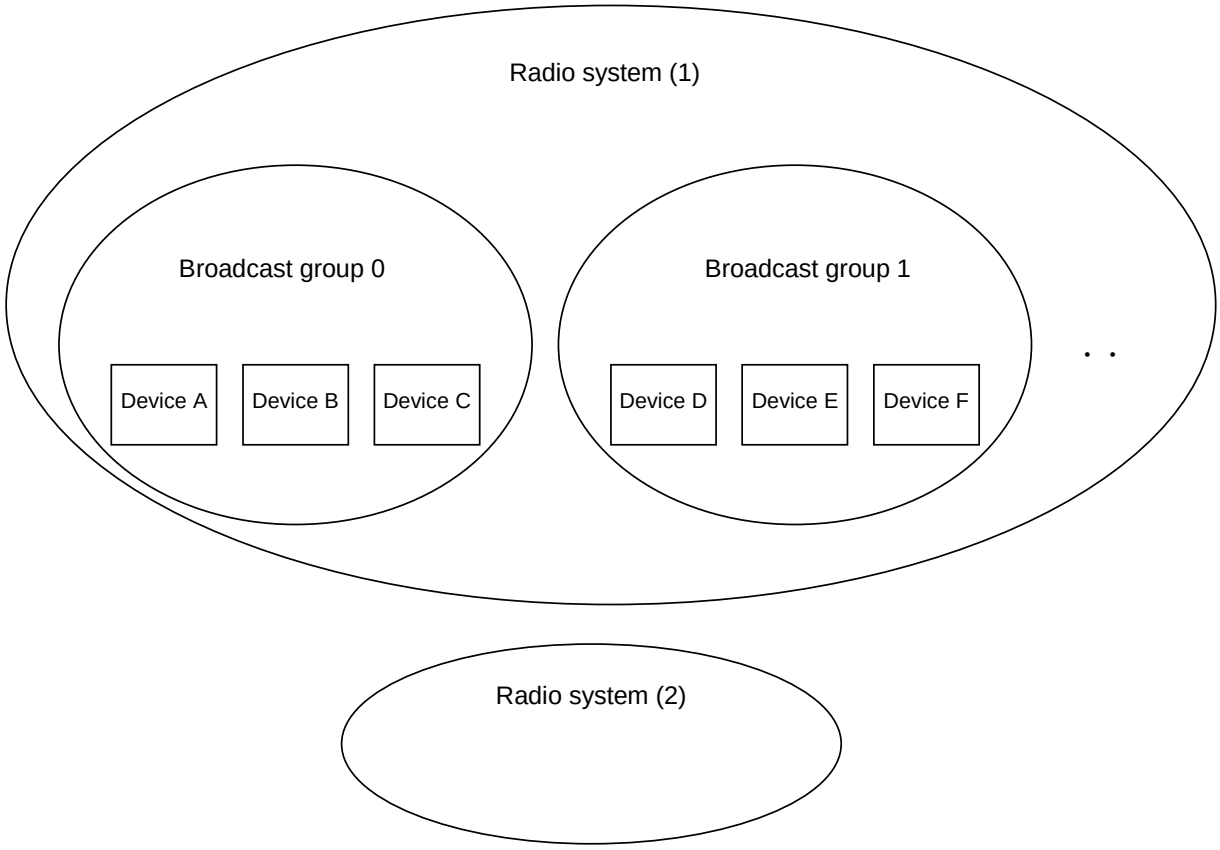
(1) Individual communication

(A) Basic procedure

The “ACK provided” and “Link connection provided” features indicated in the procedure set forth below are available as options. However, the transmission of a response upon receipt of an “ACK signal request” is mandatory.

When communication is performed between a calling office and a called office in 1:1 form, this is called individual communication. Figure 3.1 shows a basic communication procedure. In this figure, “high-order” means higher than the lower-layer communication software in the communication layers. This corresponds to the portion that is higher than the ECHONET Communication Middleware. The numbers ((1) (2) ...) described in the data in the radio communication section indicates the data number (see Section 3.4.3). The data number is provided only as an example.

The data to be transmitted is created by the high-order portion. The low-power radio unit on the calling office side transmits the created data according to the request to send (a) and notifies the high-order portion whether or not the transmission has been successful ((1) Information transmission in the figure). ((b) Notice of transmission.)



When the transmitted signal is received correctly, the low-power radio unit on the called office side informs the high-order portion of the received contents ((c) Notice of received contents) and also transmits the ACK signal to the calling office ((2) ACK signal). In Fig. 4.1, (2) ACK signal is transmitted after (c) Notice of received contents. However, this order may be reversed. The calling office side receives the ACK signal transmitted from the called office side, transfers (d) Notice of reception to the high-order portion, and informs the high-order portion that the called office side received the data.

The data ((1) Information transmission) transmitted from the calling office side includes frame synchronization 3, and the corresponding channel is secured while the receiving side prepares for the ACK signal (see Section 3.4.2). In the transmitting operation for a return of the ACK signal, the same channel as the data (1) transmitted from the calling office has priority, and carrier sense is executed for transmission.

Single data transmission is completed according to the above procedure.

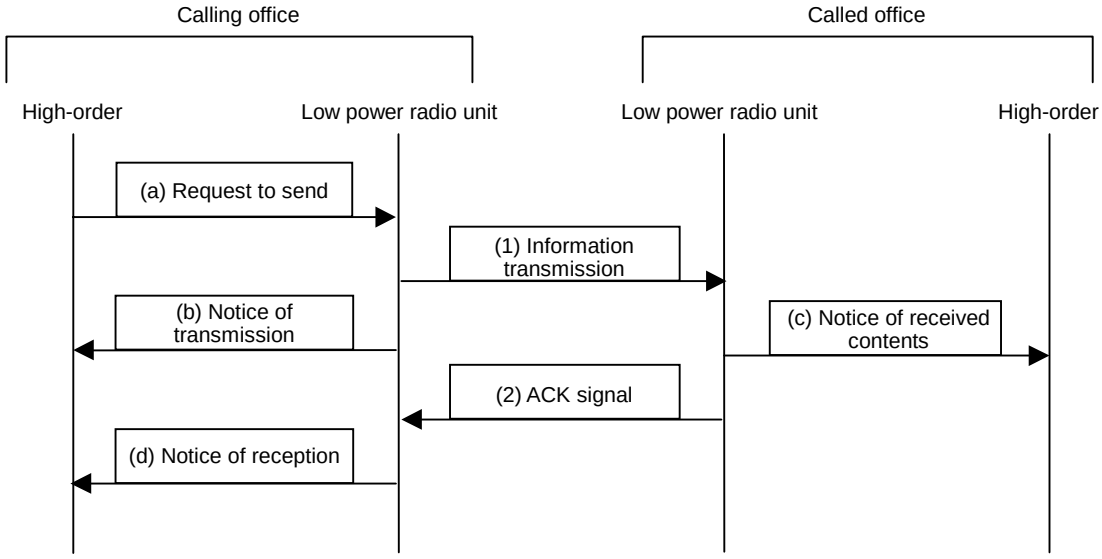


Fig. 3.1 Basic Communication Procedure

(B) Link connection

When continuous multiple communications are performed in the form of 1 calling office:1 called office, link establishment shall be enabled between the calling office and the called office to increase communication efficiency. This link establishment means that (1) Opposite party of communication is fixed by the low-power radio unit, and (2) Switching is performed to a wait for continuous reception at wait for intermittent reception.

In particular, at a wait for intermittent reception, the frequency of the repeat transmission division is raised and the data length increased by the data of the basic procedure. Link establishment can minimize the frequency of the repeat transmission division.

For link establishment, a link connection is made between the calling office and the called office. Figure 3.2 shows a communication procedure at a link connection. In the data in the radio communication section in Fig. 3.2, “Link connection: Yes”, “ACK request: No”, etc. are described in a shortened form of flag information in control code 2 in data 2.

For example:

“Link connection: Yes” means “In control code 2, the flag to indicate the request for radio link connection is set to Yes”.

“ACK request: No” means “In control code 2, the flag to indicate the request for ACK signal transmission is set to No”.

“Transmission data: No” means that the transmission is performed with a one-block data length of “0x00” (see Section 3.4.3).

As shown in Fig. 3.2, the ACK signal is not sent back to link connection data (1). Usually, after link establishment, the calling side transmits information according to the communication procedure shown in Fig. 3.3.

It is also permissible that information be sent in link connection messages.

(C) Link establishment

In link establishment status, communication is performed according to the procedure shown in (A) Basic procedure. The communication procedure in link establishment status is shown in Fig. 3.3. In link establishment status, efficient communication can be performed because of a wait for continuous reception.

When the status is “link establishment,” all types of incoming messages addressed to the home device can be received, including link connection messages, link disconnection messages, “in link” messages and single-transmission messages.

The established link shall be maintained for 10 seconds or more.

(D) Link disconnection

To terminate link establishment, link disconnection is performed between the calling office and the called office. The communication procedure at link disconnection is shown in Fig. 3.4. The ACK signal response to a Link disconnection message (1) shall be as per the “ACK request” settings. When the link is disconnected, the low-power radio section shall change the reception waiting period back to the period that was being used before the establishment of the link.

It is also permissible that information be sent in link disconnection messages.

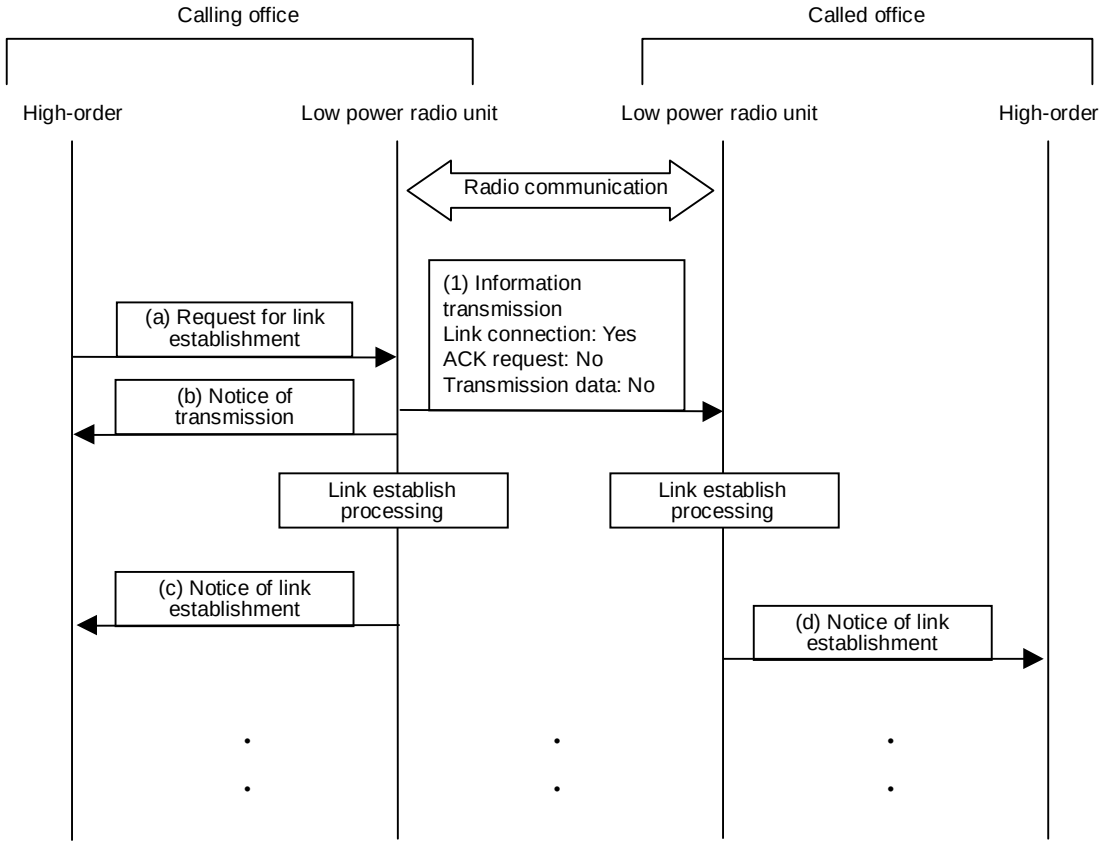


Fig. 3.2 Communication Procedure at Link Connection

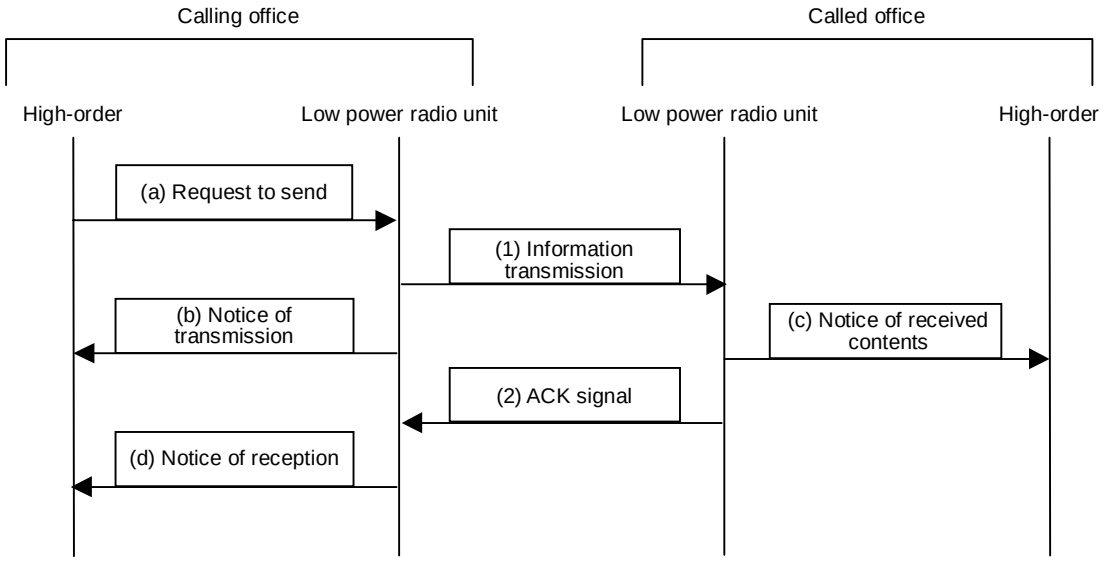


Fig. 3.3 Communication Procedure in Link Establishment Status

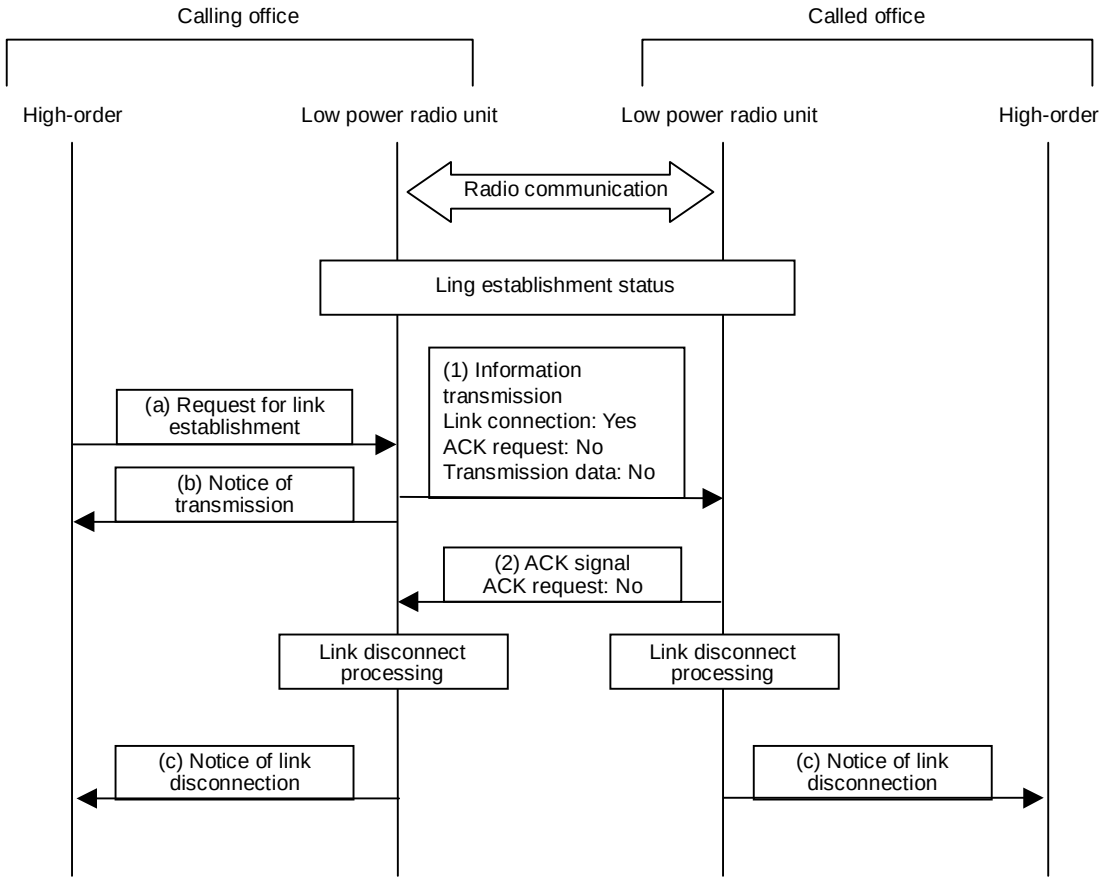


Fig. 3.4 Communication Procedure at Link Disconnection

(E) Response message to be sent back upon receipt of a version information request message (The transmission of a response message upon receipt of a version information request message is mandatory.)

Software version information can be obtained by using a radio message.office and When a master or slave node needs to have software version information, it shall set the radio system setup message flag and send a request message that contains “Transmission data: 0xE1.” The transmission channel shall be the system setup channel.

A master or slave node that received such a radio message shall set the radio system setup message flag and send back a response message that contains “Transmission data: 0xF1, 0xab, 0xcd” (“ab” represents the lower-layer communication software standard version number and “cd” represents the vendor’s software version number). The transmission channel shall be the system setup channel.

(F) Response message to be sent back upon receipt of a reception level information request message (optional)

Reception level information can be obtained by using a radio message.

When a master or slave node needs to have reception level information, it shall set the radio system setup message flag and send a request message that contains “Transmission data: 0xE0” and “No ACK request.” The transmission channel shall be the system setup channel.

A master or slave node that received such a radio message shall set the radio system setup message flag and send back a response message that contains “Transmission data: 0xF0, 0xRECEPTION LEVEL” and “No ACK request.” The transmission channel shall be the system setup channel (The RECEPTION LEVEL value shall be the reception level in dB μ V of the received message). It must be possible to indicate the reception level within the range between 0 and 30dB μ V or within a wider range.

As this function is optional, devices that are not equipped with this function shall not return a response.

(G) Vendor Messages

When sending vendor messages, which are messages that are not radio system setup messages or ECHONET messages (in terms of the information transmission signal content), the vendor message flag shall be set.

(2) Broadcast communication

When communication is performed between a calling office and called offices in the form of 1:N, this is called broadcast communication. Figure 3.5 shows the communication procedure.

The control codes for data to be transmitted from the calling office are set as follows:

Control code 1:

Opposite party classification: Broadcast communication

Control code 2:

Flag to indicate the request for ACK signal transmission: No

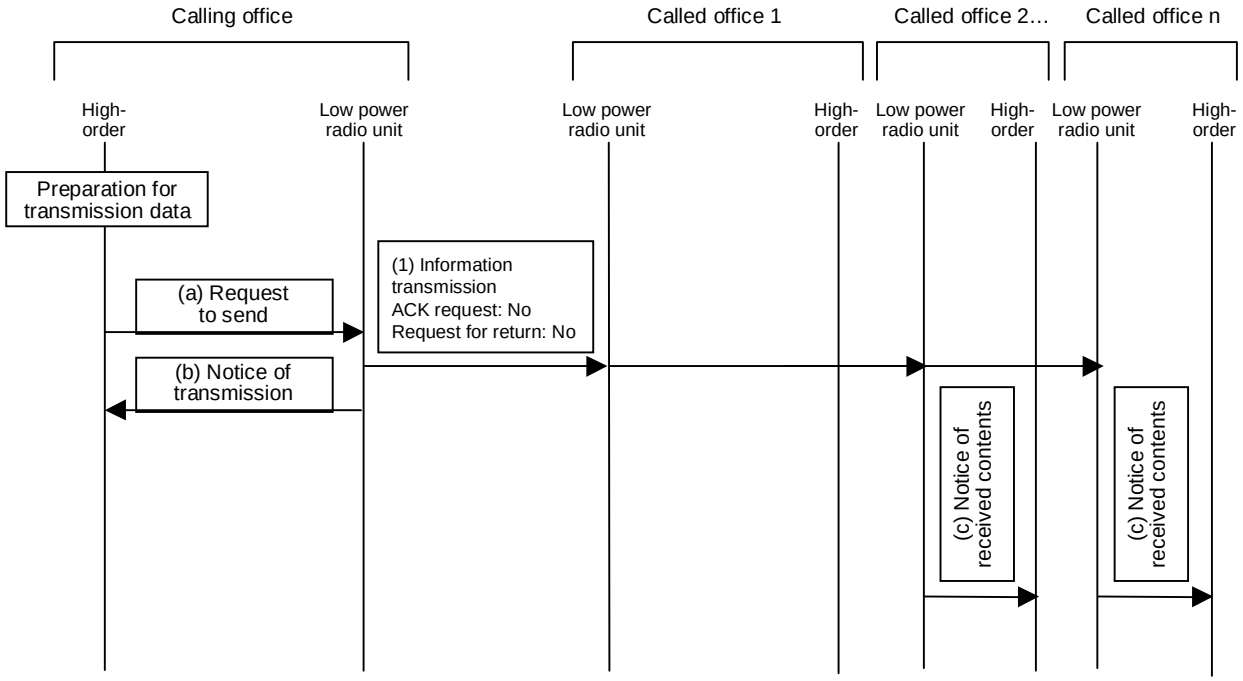


Fig.3.5 Broadcast Communication

(3) Temporary continuous reception function (optional)

This is a function whereby a device that is in the intermittent reception waiting mode can notify the device on the other end of communication that it will temporarily change to the continuous reception waiting mode by sending a message with the temporary continuous reception information flag set (Control Code 2).

If a message is sent with the temporary continuous reception information flag set, the T1 timer shall be reset-started (ACK shall not be included) and the device shall operate in the continuous reception waiting mode until the T1 timer expires (regardless of whether the type of communication is non-broadcast or broadcast).

If a message is sent or received for which the temporary continuous reception information flag has not been set (ACK shall not be included), the T1 timer shall be stopped and this function shall be deactivated. The expiration time for the T1 timer shall be 2 seconds or longer.

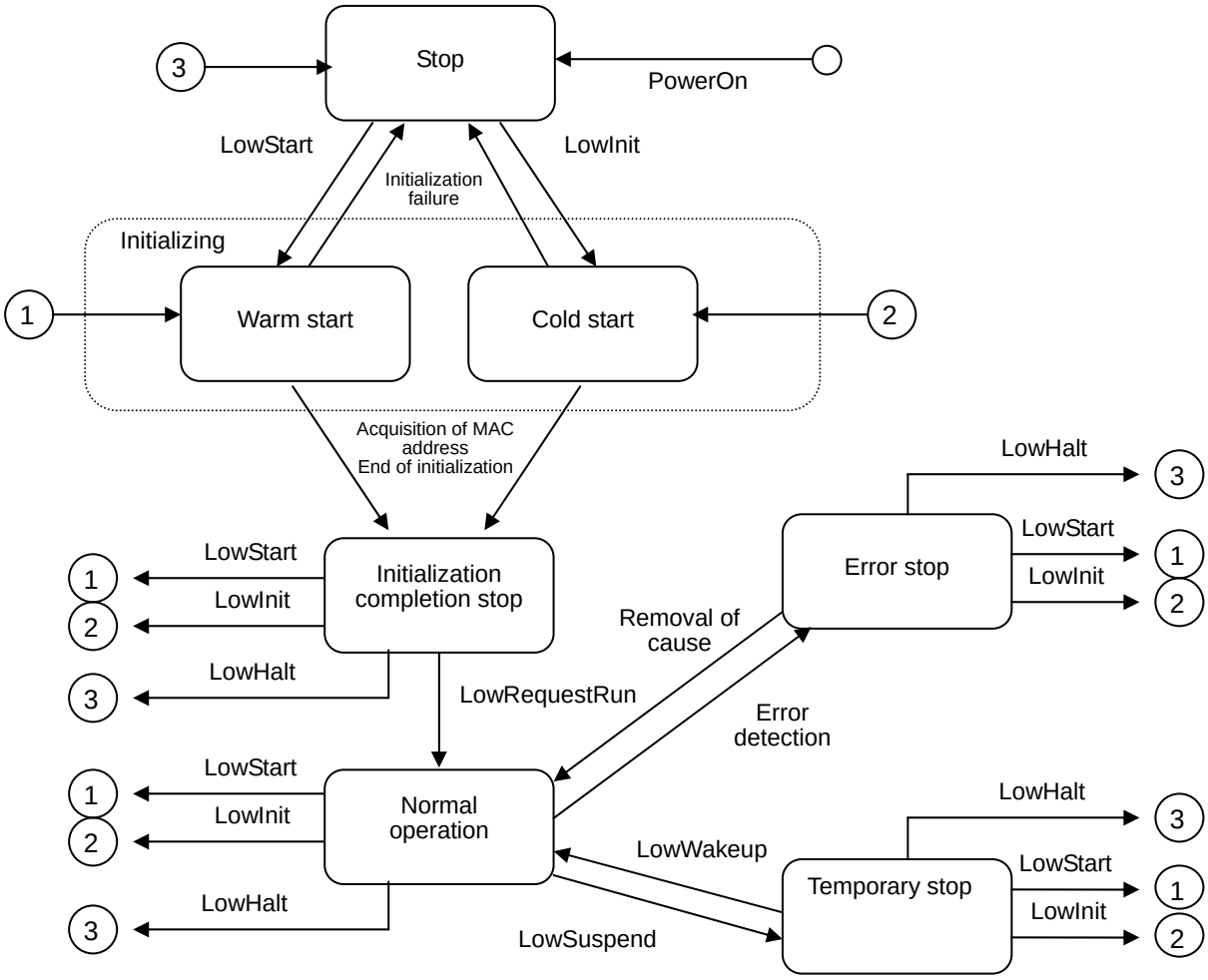
3.5 Basic Sequence

3.5.1 Basic concept

This subsection classifies the discrete lower-layer communication software status as shown below, and describes an outline of the sequence in each status.

① Stop status
② Initialize processing status
③ Normal operation status
④ Error stop status

The following figure shows the state transition diagram for each status.



3.5.2 Stop status

Stop status signifies a state in which no lower-layer communication software operations are performed. This status is established immediately after Power On. An outline of the processing immediately after state transition and an outline of the individual lower-layer communication interface services that the stop status receives, and related processing, are described below.

(1) Trigger and action
Waits for an individual lower-layer communication interface service.

(2) Status acquisition service (LowGetStatus)
Returns LOW_STS_STOP as the status.

(3) Lower-layer communication software type acquisition service (LowGetDevID)
Returns the lower-layer communication software type.

The triggers for state transition are as follows:

(1) Transition trigger to initialize processing status
This transition is caused by an initialization service (LowStart, LowInit).

3.5.3 Initialize processing status

The initialize processing status signifies that the lower-layer communication software is initialized.

An outline of the processing immediately after state transition and an outline of individual lower-layer communication interface services that the initialize processing status receives, and related processing, are described below.

(1) Trigger and action
① Initializes the transceiver.
② Obtains a unique MAC address in the subnet.
- When a warm start is used, the retained MAC address is used to start a MAC acquisition process.
- When a cold start is used, the retained MAC address is discarded, and the master node newly performs a new MAC address acquisition procedure. The retained radio system identification code remains unchanged.

③ Obtains a radio system identification code.

(2) Status acquisition service (LowGetStatus)
In a cold start, returns LOW_STS_INIT as the status. In a warm start, returns LOW_STS_RST.

(3) Lower-layer communication software type acquisition service (LowGetDevID)

Returns the lower-layer communication software type.

Triggers for state transition are as follows:

(1) Transition trigger to initialization completion stop status
The transition is caused by initializing the transceiver, getting a MAC address, and getting a radio system identification code.
(2) Transition trigger to stop status
This transition is caused by the initialization failure.

3.5.4 Initialization completion stop status

The initialization completion stop status signifies a state waiting for a request for operation start from the Communication Middleware after the lower-layer communication software is initialized. An outline of the processing immediately after state transition and an outline of the individual lower-layer communication interface services that the initialization completion stop status receives, and related processing, are described below.

(1) Trigger and action
Waits for an individual lower-layer communication interface service.
(2) Status acquisition service (LowGetStatus)
Returns LOW_STS_INIT as the status.
(3) Physical address acquisition service (LowGetMacAddress)
Returns a MAC address.
(4) Profile data acquisition service (LowGetProData)
Returns profile data.
(5) Lower-layer communication software type acquisition service (LowGetDevID)
Returns the lower-layer communication software type.

Triggers for state transition are as follows:

(1) Transition trigger to initialization processing state
This transition is caused by the initialization service (LowStart, LowInit).
(2) Transition trigger to normal operation status
This transition is caused by the operation start instruction service (LowRequestRun).
(3) Transition trigger to stop status
This transition is caused by the end service (LowHalt).

3.5.5 Normal operation status

Normal operation status signifies a state in which data is transmitted to or received from a transmission medium as the primary function of the lower-layer communication software. An outline of the processing immediately after state transition and an outline of the individual lower-layer communication interface services that the normal operation status receives, and related processing, are described below.

(1) Trigger and action
Waits for an individual lower-layer communication interface service.

(2) Status acquisition service (LowGetStatus)
Returns LOW_STS_RUN as the status.

(3) Physical address acquisition service (LowGetMacAddress)
Returns a MAC address.

(4) Profile data acquisition service (LowGetProData)
Returns profile data.

(5) Data transmission service (LowSendData)
Translates the received Protocol Difference Absorption Processing Block data into lower-layer communication software data and outputs it to the transmission medium.

(6) Data reception service (LowRecvData)
Translates the lower-layer communication software data received from the transmission medium into the Protocol Difference Absorption Processing Block data and outputs it to the Protocol Difference Absorption Processing Block.

(7) Lower-layer communication software type acquisition service (LowGetDevID)
Returns the lower-layer communication software type.

Triggers for state transition are as follows:

(1) Transition trigger to stop status
The transition is caused by the end service (LowReset).

(2) Transition trigger to initialize processing state
This transition is caused by the initialization service (LowStart, LowInit).

(3) Transition trigger to error stop status
The transition is caused by the occurrence of an error.

(4) Transition trigger to suspension state
This transition is caused by the suspension service (LowSuspend).

3.5.6 Error stop status

The error stop status signifies a state in which operation is stopped by the occurrence of an error. An outline of the processing immediately after state transition and an outline of the individual lower-layer communication interface services that the error stop status receives, and related processing, are described below.

(1) Trigger and action
Performs error processing.

(2) Status acquisition service (LowGetStatus)
Returns LOW_STS_SUSPEND as the status.

(3) Lower-layer communication software type acquisition service (LowGetDevID)
Returns the lower-layer communication software type.

Triggers for state transition are as follows:

(1) Transition trigger to stop status
The transition is caused by the end service (LowHalt).

(2) Transition trigger to initialize processing status
This transition is caused by the initialization service (LowStart, LowInit).

(3) Transition trigger to normal operation status
The transition is caused by removing the cause of the error.

3.5.7 Suspension status

The suspension status signifies a state in which operation is paused by an instruction from the Communication Middleware. An outline of the processing immediately after state transition and an outline of the individual lower-layer communication interface services and its processing are described below.

(1) Trigger and action
Stops the operation of the lower-layer communication software.

(2) Status acquisition service (LowGetStatus)
Returns LOW_STS_SUSPEND as the status.

(3) Lower-layer communication software type acquisition service (LowGetDevID)
Returns the lower-layer communication software type.

Triggers for state transition are as follows:

(1) Transition trigger to normal operation status
This transition is caused by the end service (LowHalt).

(2) Transition trigger to normal operation state

This transition is caused by the operation restart service (LowWakeUp).

(3) Transition trigger to initialize processing state

This transition is caused by the initialization service (LowStart, LowInit).

Chapter 4 Extended HBS Communication Protocol Specifications

4.1 System Overview

This Specification provides the extended HBS communication protocol for pair cable as an ECHONET transmission medium. The specifications of the communication protocol to be used for this medium were already established in 1988 as “ET-2101 Home Bus System (HBS)” by the Electronic Industries Association of Japan (EIAJ: predecessor to JEITA, which was formed by the merger of EIAJ and JEIDA in November 1, 2000). After that, “Addresses and Commands related to AVC Service of the ET-2012 Home Bus System” was released in January 1990, and “ET-2101-1 Home Bus System (Supplement)” was published in November 1990. The EIAJ Standard specifies “Twisted pair cable” and “Coaxial cable” as transmission media.

As a rule, if standards established in the past are available and still effective, ECHONET uses them where applicable. The EIAJ Standard includes Layers 1 to 7 of the OSI communication layer configuration as well as multiple information channels for the transmission of audio and video, in addition to equipment control channels. The pair cable protocol for ECHONET includes the lower-layer layers (Layers 1 to 3) and provisions on control channels and part of the provisions of Layer 7. However, the ET-2101 Standard relates to specifications of the lower-layer medium specifying the co-existence of both CT and AVC systems (e.g., the mounting of four sets of twisted pair cables, the use of 8-pin modular jacks as connectors, etc.) and includes some excessive specifications for actual system construction that place constraints on the equipment. When adopting the ECHONET Standard, new provisions have been added for these portions, and additional provisions have been specified for insufficient portions.

In the following sections in this Chapter, the applicable portions have been extracted from the ET-2101 Standard and new portions added. The principal differences with ET-2101 are as follows:

(1) One pair of twisted pair cable is allowed.
(2) Regarding socket shape, a screw-fixing specification has been added in addition to the 8-pin modular jack.
(3) The allowable transmission distance (cable length) for pair cables is 1 km, and specifications for related signal levels have been added.
(4) The data area specifications, including command specifications (related to Layer 7), are newly provided as the extended HBS. As a rule, however, ET-2101 specifications are followed whenever possible.
(5) New specifications are added for address redundancy detection.

4.2 Mechanical and Physical Characteristics

The following six items are specified as mechanical and physical specifications for the extended HBS. For Specifications 3) and 5), the EIAJ ET-2101 Standard (HBS standard) shall be fully applied. The applicable specifications are described below.

Note: Principal differences with ET-2101

1) One pair of pair cables shall also be allowed.
2) A maximum length of 1 km shall be allowed in consideration of medium and small buildings.
5) Because a specification for the number of pairs of pair cables has been added, a specification for information sockets corresponding to the addition has also been added.

1) Transmission media and number of transmission pairs
   - The ET-2101 Standard “3.1.1 Transmission media and number of transmission pairs” is applied, and a part is additionally specified.
   - “5.2.1 Transmission media and number of transmission pairs” provides detailed specifications.

2) Cable length
   - The ET-2101 Standard “3.1.2 Cable length” is applied, and a part is additionally specified.
   - “5.2.2 Cable length” provides detailed specifications.

3) Topology
   - The ET-2101 Standard “3.1.3 Topology” is applied.

4) Information socket shape (including compatibility with signals)
   - The ET-1201 Standards “3.1.4 Information socket shape” and “3.1.6 Compatibility between information sockets and signals” are applied. A part is additionally specified.
   - “5.2.3 Information socket shape (including compatibility with signals)” provides detailed specifications.

5) Number of information sockets
   - The ET-2101 Standard “3.1.5 Number of information sockets” is applied.

4.2.1 Transmission media and number of transmission pairs

(1) Cable type: Twisted pair cable

(2) Number of pairs: 1 pair (1 pair for control and 3 pairs for information in the case of HBS)

4.2.2 Cable length

The maximum cable length shall be 1 km per cluster. However, the applicable cable diameter is specified as follows:

Twisted pair cable length: 1 km max.	However, when the cable length is 200 m or less, the cable diameter shall be 0.65 mm. When the cable length exceeds 200 m but does not exceed 1 km, the cable diameter shall be 1.2 mm.

4.2.3 Topology

Bus system

4.2.4 Number of terminals to be connected

When the cable length is 200 m, the number of terminals to be connected shall be 64 per cluster.

When the cable length exceeds 200 m but does not exceed 1 km, the number of terminals to be connected shall be 128 per cluster. Logically, the maximum number in one system shall be 256.

4.2.5 Information socket shape (including compatibility with signals)

The ET-2101 Standards “3.1.4 Information socket shape” and “3.1.6 Compatibility between information sockets and signals” are applied. In the case of one pair of twisted pair cable, screw fixing shall be allowed.

4.2.6 Compatibility between information sockets and signals

The ET-2101 Standard “3.1.6 Compatibility between information sockets and signals” is applied.

4.3 Electrical Characteristics

With the exception of "Load resistance of control channel cable", the ET-2101 Standard "3.2 Electrical Characteristics" is applied. In this section, the specification for the cable diameters specified in "4.2.2 Cable length" is also provided for "Load resistance of control channel cable" to be additionally specified.

4.3.1 Characteristic impedance of cable

Short-conductor cable with cable diameter of 0.65 mm:	300Ω
Short-conductor cable with cable diameter of 1.2 mm:	150Ω
Stranded cable with nominal sectional area of 0.75 mm^2:	200Ω

However, when the cable diameter is 1.2 mm and the nominal sectional area is 0.75 mm^2, the transmission distance shall be more than 200 m and as much as 1 km.

4.3.2 Load resistance of control channel cable

The processing method for the load to be applied to the cable shall be as outlined below. For the load resistance, a condenser is connected in series and the direct current is cut in consideration of power feed.

(1) For cable length of 200 m or less
A 75Ω resistor or a 39Ω load resistance is connected to the terminal.

(2) For cable length of more than 200 m and as much as 1 km
A 100Ω resistor is connected to each terminal.

4.3.3 Transmission rate of control signal

9600 bps ± 0.13%

4.3.4 Transmission system and transmission waveform of control signal

(1) Transmission system: Base-band transmission

(2) Transmission waveform: AMI (Alternate Mark Inversion) shown in Fig. 4.1 and negative logic with a duty ratio of 50%.

The start bit of each byte shall be transmitted from the 0 (+) side for collision detection.

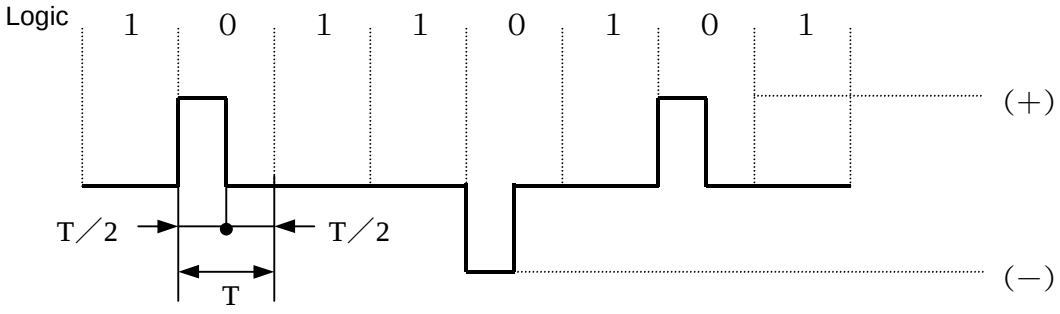


Fig. 4.1 Transmission Waveform of Control Signal

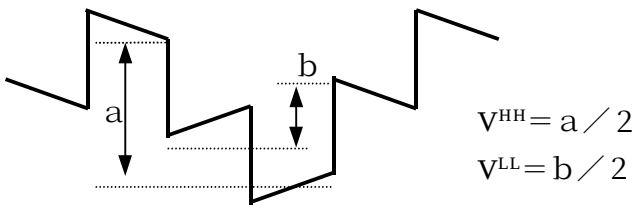
4.3.5 Transmitting/receiving level of control signal

The transmitting/receiving level of the control signal shall be as shown in Table 4.1.

Table 4.1 Transmitting/Receiving Level of Control Signal

Logic	Receiving level	Transmitting level
1	V^{LL} = 0.6 V or less	V^{LL} = 0.6 V or less
0	V^{HH} = 1.4 V or more	V^{HH} = 2.5 V or more

Note: The voltage represents the control signal level on the cable.



4.3.6 Impedance and power feed voltage of terminals to be connected

(1) Input impedance: 10 kΩ or more for a frequency of 5 kHz

(2) Output impedance: 40 Ω or less for a frequency of 5 kHz

Remarks: In consideration of power feed, a condenser must be connected in series. The above value shall include the condenser for cutting this direct current.

4.3.7 Power feed voltage of control channel

Power feed is allowed. The maximum power feed voltage shall be 36 V DC.

4.4 Logical Layers (Layer 1 Specifications)

This section provides the logical specifications for Layer 1 of the pair cable communication protocol.

The following eight items are specified as logical specifications for Layer 1. EIAJ ET-2101 (HBS Standard) is fully applied for all of these items. An outline of the specifications is provided below. (For details, see “ET-2101”.)

1) Control system
2) Synchronization system
3) Basic format of control signal
4) Pause time and pause period
5) Packet priority
6) Collision detection procedure
7) Synchronization recovery procedure
8) Short data interruption procedure

4.4.1 Control system

Survival type CSMA/CD (Carrier Sense Multiple Access with Collision Detection)

4.4.2 Synchronization system

Start-stop synchronization. The configuration shall be as follows:

(1) Character configuration: 11-bit configuration, namely, start bit (1 bit), data (8 bits), parity (1 bit), and stop bit (1 bit). The parity shall be even parity (see Fig. 5.2).
(2) Start bit transmission: (+) side
(3) Data transmission: LSB first (negative logic)
(4) Parity: Even parity
(5) Character spacing: No spacing between the stop bit and the next character

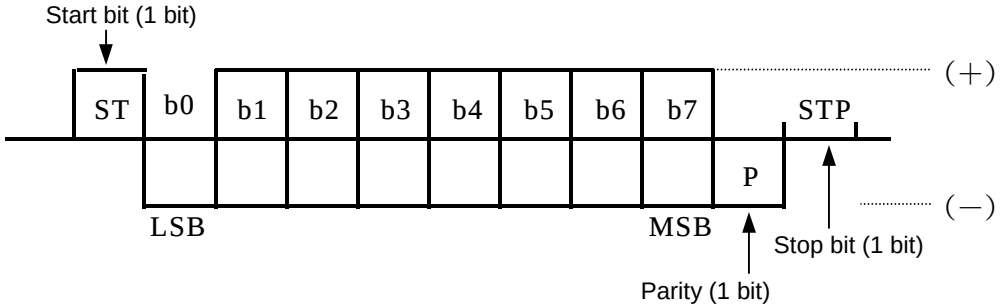
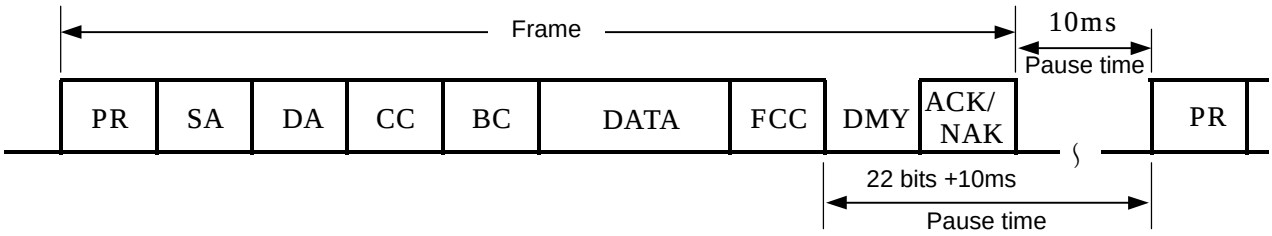


Fig. 4.2 Character Configuration

4.4.3 Basic format of control signal

Figure 4.3 shows the basic format of the control signal.



PR(Priority)	: Priority code	(11bit)
SA(Source Address)	: Self code	(11bit)
DA(Destination Address)	: Destination code	(11bit)
CC(Control Code)	: Control code	(11bit)
BC(Byte Count)	: Data length code	(11bit)
DATA(Data Field)	: Data field	(11bit＊Max.256)
FCC(Frame Check Code)	: Frame check code	(11bit)
DMY(Dummy Code)	: Dummy	(11bit)
ACK/NAK(ACK/NAK Code)	: ACK/NAK code	(11bit)

Fig. 4.3 Basic Format of Control Signal

4.4.4 Pause time and pause period

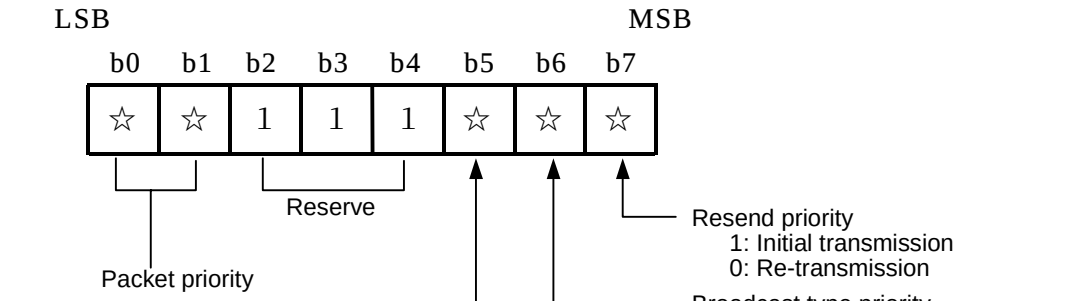
(1) Pause time: 10 ms (96-bit time) from the end of the stop bit of the ACK/NAK code

(2) Pause period: 10 ms + 22-bit time from the end of the stop bit of the check code

Note: The terminal that intends to transmit new data monitors the pause time on the bus and then transmits the data according to the synchronization recovery procedure.

4.4.5 Packet priority

Packet priority is performed by contention between the priority code PR and the self address SA. Figure 4.4 shows the bit configuration of the priority code (for details, see ET-2101 “3.3.5 Priority code”).



Priority level	b0	b1	Contents
1	0	0	System privilege packet
2	0	1	Line switching packet
3	1	0	Short-data ordinary packet
4	1	1	Long-data ordinary packet

Fig. 4.4 Priority Code Bit Allocation

4.4.6 Collision detection procedure

Collision detection is performed according to the following procedure to determine the surviving packet (for details, including concrete examples, see ET-2101 “3.3.6 Bit collation at contention and collision detection”).

(1) The terminal that intends to transmit data collates transmit data with receive data in “bit” units (bit collation).

(2) When a mismatch is detected between transmit data and receive data in bit units (at collision detection), the transmission is immediately stopped and reception is started. When transmission is re-enabled, the data is transmitted again. (At this time, the flags related to re-transmission are not changed.)

(3) Bit data with logic 0 has priority over bit data with logic 1.

(4) As a result of contention of the priority code and the self address division, the terminal with higher priority survives.

(5) Collision detection is performed when 45.5μs have elapsed from the starting time of each bit. (In the case of HBS, it is performed when 26μs have elapsed.)

(6) To reduce collisions in the bus idle status and to detect collisions securely when they occur, the delay time (Td) from free channel check to start of transmission is defined as the allowable transmission time. At 50% AMI, the Td value shall be 4.0μs or less from the setting of the start bit.

4.4.7 Synchronization recovery procedure

Synchronization between multiple IFUs is recovered according to the following procedure (for details, including figures, see ST-2101 "3.3.7 Synchronization recovery procedure"):

(1) Synchronization recovery: Monitors bus from time TF (synchronization recovery monitoring time) ahead of the end of the pause time.

(2) Synchronization recovery monitoring time (TF): Time equivalent to 2 bits (= 1/9600 × 2μs)

(3) Reception: Enters a receive enable status after a lapse of (10 ms - TF) time subsequent to the end of data.

(4) Transmission: If another terminal starts transmission in the TF period, transmission is started in synchronization with it. If no other terminal starts transmission in the TF period, transmission is performed after the pause time (10 ms). In the period of (10 ms - TF) time from the end of data, transmission is inhibited.

(5) Allowable transmission delay time (Td): Delay time from start of transmission by another terminal in the synchronization recovery monitoring time (TF) until start of transmission in synchronization with it, and delay time from a free channel check until start of transmission. 1/8 bit time (13μs) or less after the rise of the start bit.

4.4.8 Short-data interruption procedure

The short-data interruption procedure is as follows:

(1) If a request for short-data frame transmission is made during long-data frame transmission, the data field of the long data can be interrupted.

(2) As a break signal, "logic 0 (+)" is transmitted in synchronization with the stop bit of long data. However, break signal transmission is allowed only within the allowable transmission delay time (Td).

(3) The break enable period is from the end of long data code BC of the long-data frame until the start of the check code FCC.

(4) For processing after a break, as soon as the terminals on the transmitting side and the receiving side that perform long-data communication detect a break signal, they stop transmit and receive processing and enter the pause time (10 ms + 22 bits) from the end of the break signal.

(5) The terminal that transmitted a break signal and has a request for short-data frame transmission transmits a short-data frame after the pause period.

(6) The terminal that stopped transmitting long data because of the break signal does not increment the control code re-transmission count after the pause time, but instead re-transmits the long-data frame that was at a break.

(7) After the pause period, a short-data frame and a long-data frame are transmitted simultaneously. However, the short-data frame receives priority because of contention by the priority bit of the priority code.

After the short-data frame that survived as a result of contention has been transmitted, the long-data frame is transmitted again in or after the pause period.

4.5 Logical Specifications (Layer 2 Specifications)

The eight items below are specified for Layer 2 specifications. For specifications 2), 6) and 7), EIAJ ET-2101 (HBS Standard) is fully applied.

Note: The principal differences with the standard are as follows:

1) The address area of the router is additionally specified.
3) Bit settings for application to ECHONET are specified.
4) The size specified for short data is increased from 16 bytes to 32 bytes.
5) Specifications for starting lower-layer transmission media and using maintenance commands are provided.
8) The contents of error detection and NAK code are additionally specified.

1) Address
2) Broadcast, simultaneous broadcast, group broadcast
3) Control code
4) Long-data code
5) Data area
6) Check code
7) Dummy
8) Error detection and error control

4.5.1 Address

The size of the self address (SA) and the destination address (DA) shall be 1 byte and conform to Table 4.2 Address Code Allocation Table. Regarding the priority of the address code, the 4 low-order bits have the same priority at collision as the 4 high-order bits. Since bit transmission is started from the low-order bits, the 4 low-order bits have priority over the 4 high-order bits.

Table 4.2 Address Code Allocation

4 high-order bits / 4 low-order bits	0,8,4,C,2,A,6,E,1,9,5,D,3,B,7,F (Order of descending priorities from the left at collision)
0	For router, GW
8, 4. C	For security device
Others	For devices other than the above

4.5.2 Broadcast, simultaneous broadcast, and group broadcast

Broadcast shall be performed according to the procedure shown below. Here, the address code shall be as shown in Fig. 4.5.

(1) Simultaneous broadcast (transmission to all terminals connected to the transmission line)
- Set b6 of the priority code (PR) to 1 and all the bits of the destination address (DA) to 1, and specify all groups.

(2) Group broadcast (transmission to a part of address groups)
- Set b6 of the priority code to 1 and specify a group (0 to 7) specified in each bit of the destination address (DA). (See Fig. 4.5.)

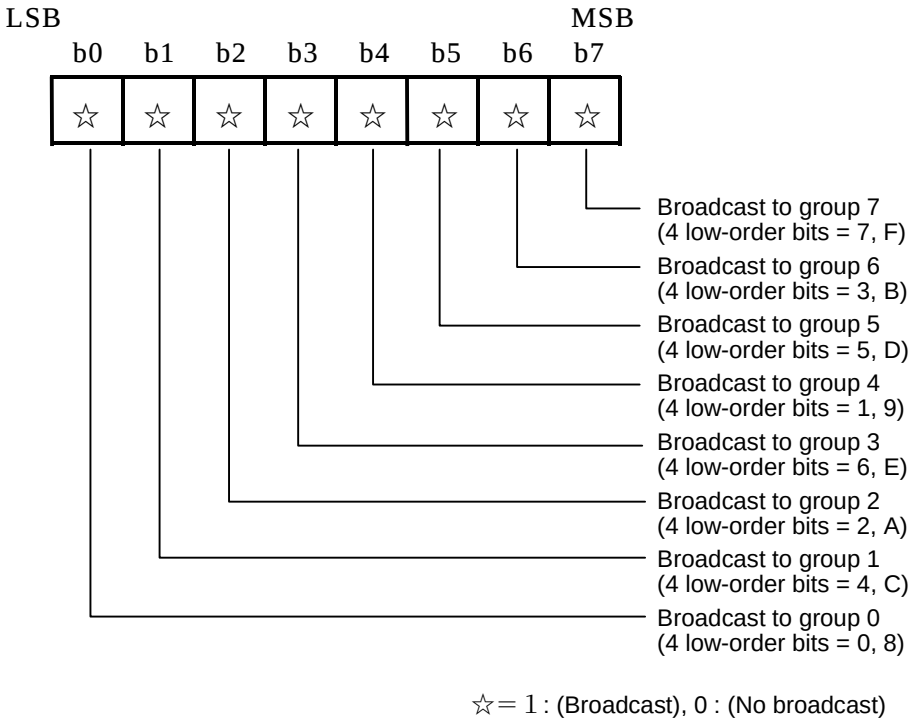


Fig. 4.5 Bit Specification for Group Broadcast

4.5.3 Control code

The allocation of the control code shall be as shown in Fig. 4.6. The ET-2101 specifies that bits (b2 to b5) specified for protocol expansion are positioned for protocol specification, and both ECHONET data and extended HBS can be set in the DATA area. Messages designated as an extended HBS message shall not be delivered to an individual lower-layer communication interface (they shall be processed as extended HBS messages). Messages designated as an ECHONET message (Version 1.0) shall be delivered to the ECHONET Communication Middleware via an individual lower-layer communication interface. For detailed requirements for b0, b1, b6, and b7, see under

"3.3.11 Control code (CC)" in the ET-2101 standard.

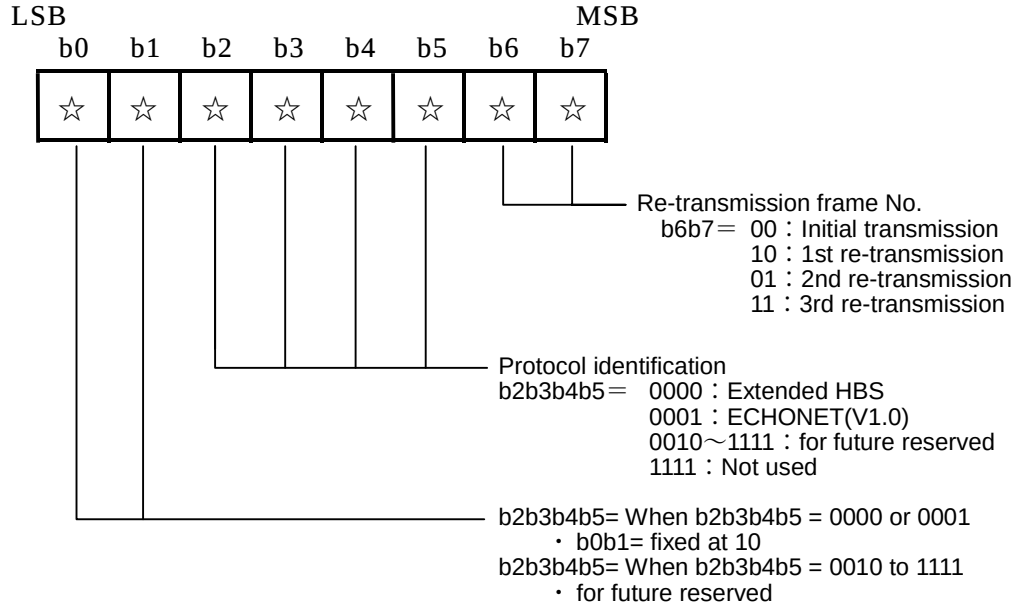


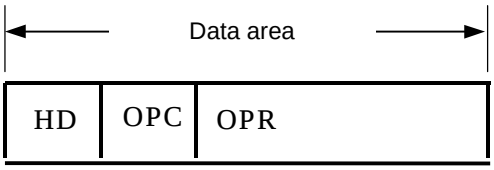
Fig. 4.6 Control Code Bit Allocation

4.5.4 Data length code

The data length code indicates the number of characters in the data field. The data length code x '01' to x 'FF' indicates 1 to 255 characters. The data length code x '00' indicates 256 characters. In this extended HBS Standard, when the data field length is 32 characters or less, it is specified as short data, and the others are specified as long data.

4.5.5 Data area

The data area structure depends on the values of b2 to b5 in the control code. This Standard specifies both the extended HBS specifications (control code b2:b3:b4:b5 = 1:1:1:0 specification) and the ECHONET V1.0 specifications (control code b2:b3:b4:b5 = 1:1:0:1 specification). In the ECHONET Specification, the data area adopts the data structure specified in the Communication Middleware specifications (see Part 2). Figure 4.7 shows a data area structure in the extended HBS specifications.



HD (extend Heder) : Header code 1 (11bit)
OPC (extend Operation Code) : Operation code (11bit)
OPR (extend Operand Code) : Operand code (11bit*Max.30)

Note: Details of each code are specified in the Layer 3 specifications.

Fig. 4.7 Data Area Structure in Extended HBS Specifications

4.5.6 Check code

For frame transmission error detection, a 2's complement value of the sum from the self address to the last character of the data area is transmitted at the end of the frame. However, the check code is a value of 1 low-order byte obtained by the aforementioned calculation.

4.5.7 Dummy code

For the dummy code, one character is assigned as an error check calculation time. During this time, bus idle status is continued without data or characters. The receiving device calculates the check code of the received frame in this period and performs one-byte response processing after a lapse of 11 bits.

4.5.8 Error detection and error control (ACK/NAK response)

"Error detection" is executed to increase the reliability of a received frame by providing one bit as parity for each byte or one byte as the check code for the whole frame to detect a transmission error due to data change or a lack of data. The parity shall be even parity.

The ACK/NAK response processing shall be as follows:

(1) For data addressed to the self address that does not correspond to simultaneous broadcast or broadcast, one byte of ACK/NAK code is transmitted as a response after the dummy code. However, on detection of the following address redundancy, a code to indicate address redundancy is transmitted as the ACK/NAK code even if the data is not addressed to the self address.

(2) The transmitting terminal transmits the control signal frame (from priority code PR to check code FCC), and the receiving terminal side performs signal frame error detection. When the control signal is received correctly, the receiving terminal side transmits the ACK signal to the transmitting terminal side.

(3) When the control signal cannot be received correctly, the receiving terminal side transmits the NAK signal to the transmitting terminal side.

(4) At data transmission other than broadcast, if the data transmitting side received the NAK response after the dummy code, it resends the frame after the pause period. At this time, the re-transmission frame number (b6, b7) of the control code is changed based on the re-transmission count. The maximum re-transmission count shall be 3.

(5) When a code other than the ACK/NAK code is received after the dummy code, it is always regarded as NAK. No response at data transmission except broadcast is also regarded as NAK.

(6) At address redundancy detection, the NAK signal to indicate address redundancy is transmitted even if it is broadcast.

(7) When receiving the NAK signal to indicate address redundancy, the processing to be performed by the transmitting terminal side is specified by the basic sequence in the Layer 3 specifications.

ACK/NAK code	ACK	: x ‘06’	
	NAK	: x ‘15’	(Parity error or FCC error)
		: x ‘00’	(Address redundancy detection)
		: x ‘11’	(Receiving buffer full)
		: x ‘12’	(Terminal [application] failure)

When errors of the above four types of NAK are detected in redundant form, the code is determined with the following priority and returned (in the order of subsequent priorities):

x ‘15’ → x ‘00’ → x ‘11’ → x ‘12’

When FCC error and address redundancy detection occur simultaneously, it is indicated that the FCC error notice has priority.

4.6 Logical Specifications (Layer 7 Specifications)

Extended HBS does not provide any sub-bus specifications. Extended HBS is intended to specify pair cables as lower-layer transmission media in the ECHONET Standard, and the high-order layer processing is realized as processing in the ECHONET Communication Middleware. Extended HBS provides specifications that take into consideration maintenance of lower-layer transmission media as the Layer 7 specifications. The specifications in this section relate to data contents (see Fig. 4.7) and data sequence in cases where extended HBS has been selected. They consist of the following items:

1) Header code (HD)
2) Command (OPC, OPR)
3) Communication sequence

4.6.1 Header code (HD)

The header code allocation shall be as shown in Fig. 4.8.

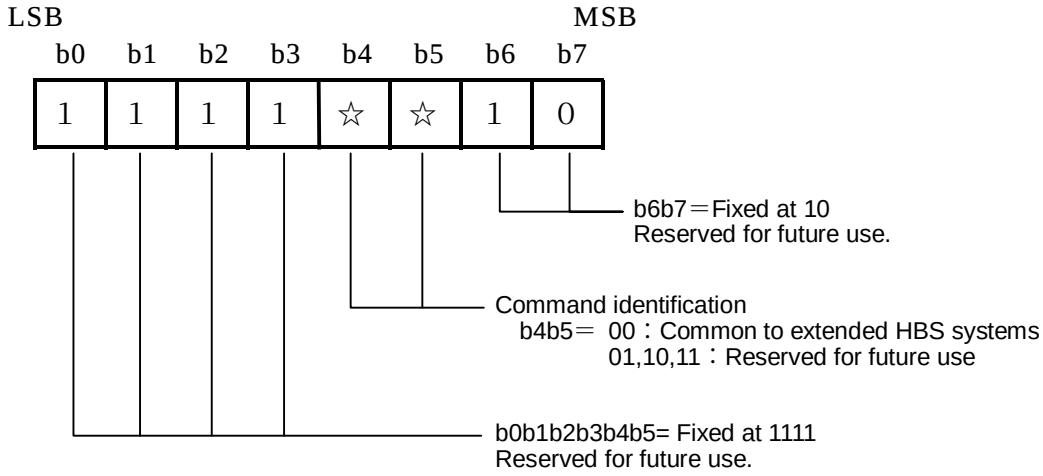


Fig. 4.8 Header Code Bit Allocation

4.6.2 System common commands

System common commands are defined as those commands used among devices connected to the extended HBS.

(1) Basic command form
- Structure with OPC (operation code) only
- Structure with OPC and OPR (operand)

(2) OPC (operation code)

The area specified as OPC shall be 128 codes whose 4 high-order bits are 8 to F. The OPC code is classified into 2 types depending on whether or not an OPR exists.
- 4 high-order bits = 8, 9: OPR does not exist.

- 4 high-order bits = A, B, C, D, E, F: OPR exists.

(3) OPR (operand code)

The area that can obtain an OPR code shall be one whose 4 high-order bits are 9 to 7. The size and meaning of an OPR differs with each OPC.

(4) Meaning of mandatory and free adoption

Transmitting source, mandatory:	Must always transmit.
Transmitting source, free adoption:	May or may not transmit.
Receiving destination, mandatory:	Must not ignore. (Must always process.)
Receiving destination, free adoption:	May ignore.

Table 4.3 shows an OPC code allocation table, and Appendix 4.2 of this section describes detailed specifications for each command, including OPR.

Table 4.3 OPC Code Allocation

	8	9	A	B	C	D	E	F
0	Reset		Startup start					
1			Startup check					
2		OK	Startup completion					
3		NG						
4		Dummy						
5								
6								
7								
8			Return request					
9			Return response					
A			Version request					
B			Version response					
C	Communication stop request		Maker name request					
D	Communication stop response		Maker name response					
E	Communication start request							
F	Communication start response							

Note: Shaded portions are reserved for future use.

4.6.3 Communication sequence

Two communication sequences are described:

1) Basic communication sequence

2) Startup communication sequence (physical address acquisition PnP sequence)

(1) Basic communication sequence

The following commands must always respond to requests, and the basic sequence for this is shown in Fig. 4.9. The value in parentheses denotes an OPC code value.

① Communication stop request (8C)/response (8D)
② Communication start request (8E)/response (8F)
③ Loopback request (A8)/response (A9)
④ Communication software version request (AA)/response (AB)

On reception of the request data of ① and ②, response processing shall be based on "free adoption". In the case of adoption, the communication sequence shown in Fig. 4.9 shall be observed.

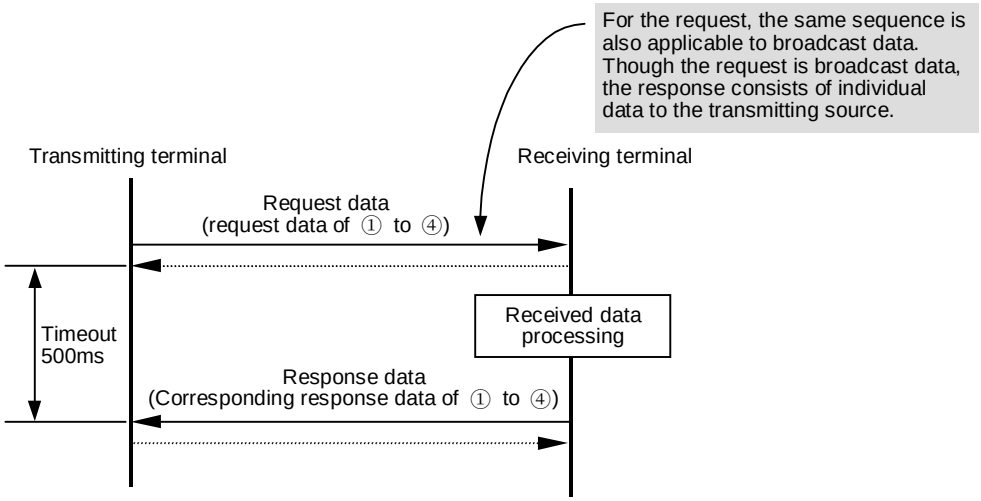


Fig. 4.9 Basic Communication Sequence

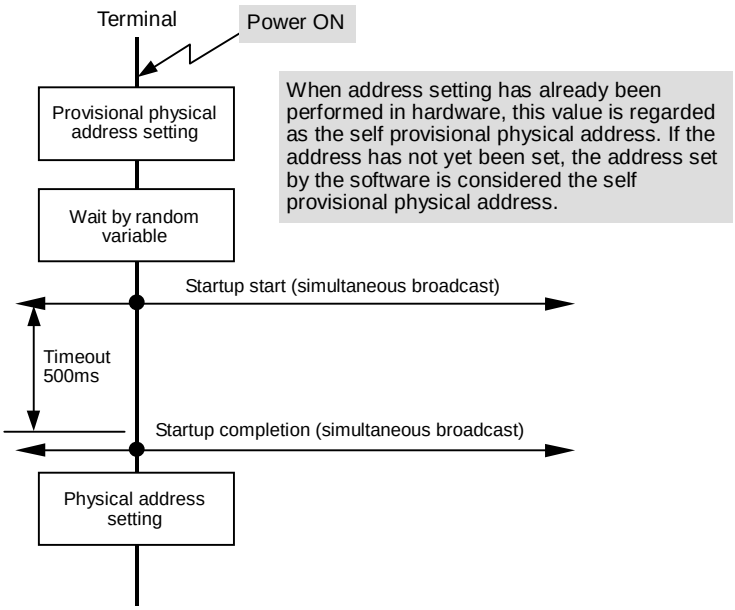
(2) Startup communication sequence (physical address acquisition PnP sequence)

The two commands shown below are used for physical address setting at startup. The value in parentheses denotes an OPC code value. In this sequence, the device conforming to the extended HBS must always hold the youngest physical address among the devices connected to the network.

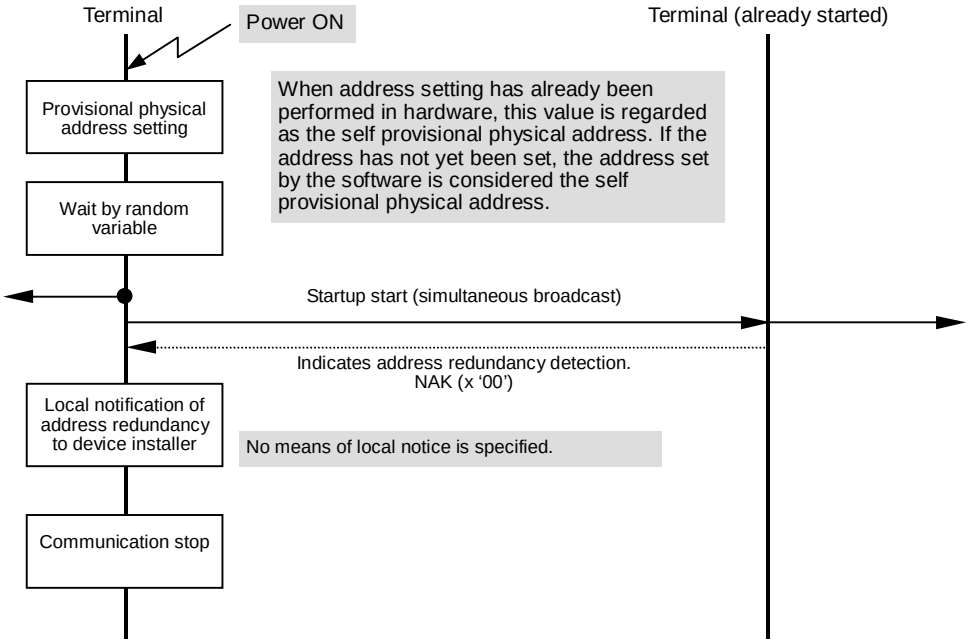
① Startup start (A0)
② Startup check (A1)
③ Startup completion (A2)

The communication procedures to be performed in various cases are described below:

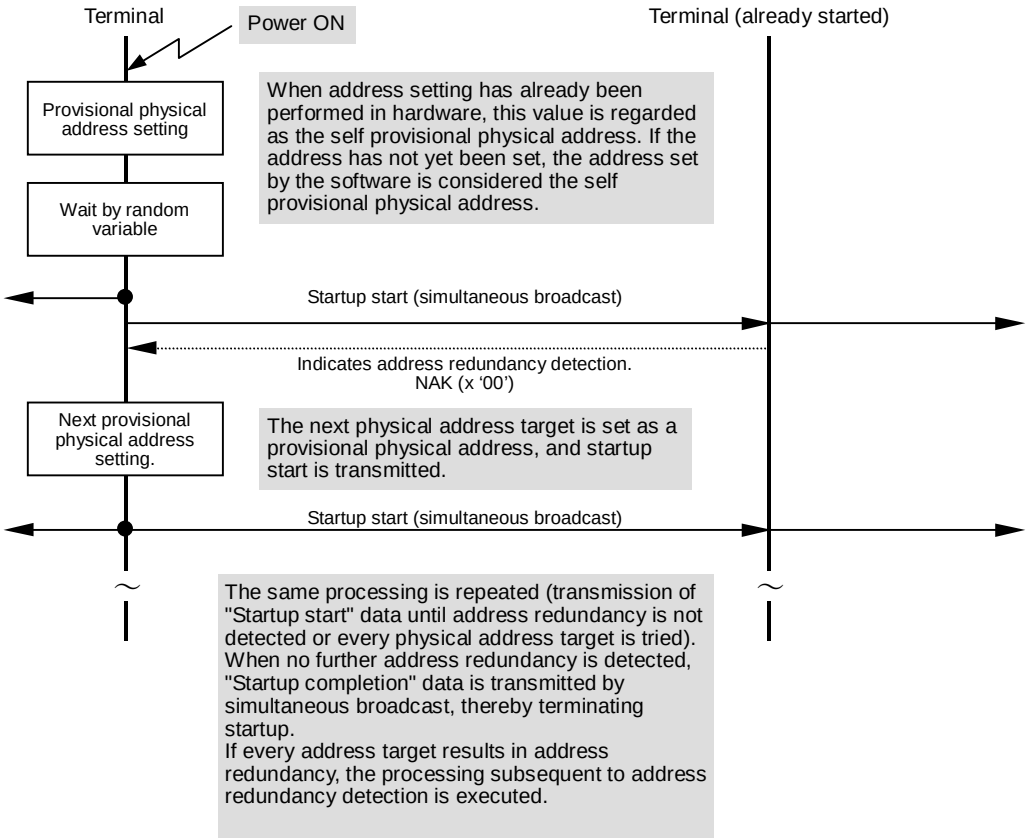
<CASE 1> ① Other terminals do not exist.
② Other terminals exist, but there is no address redundancy.



<CASE 2> ③ Other terminals exist, there is address redundancy, and the address setting is disabled except for the redundant address (fixed by the DIP switch).



<CASE 3> ④ Other terminals exist, there is address redundancy, and the address setting is enabled in software form except for the redundant address.



4.7 Basic Sequence (Software Internal State Transition Specifications)

This section describes the basic processing sequence of the lower-layer communication software for extended HBS communication.

The following is also included:

- State transition diagram
- Sequence description of various states indicated in the state transition diagram

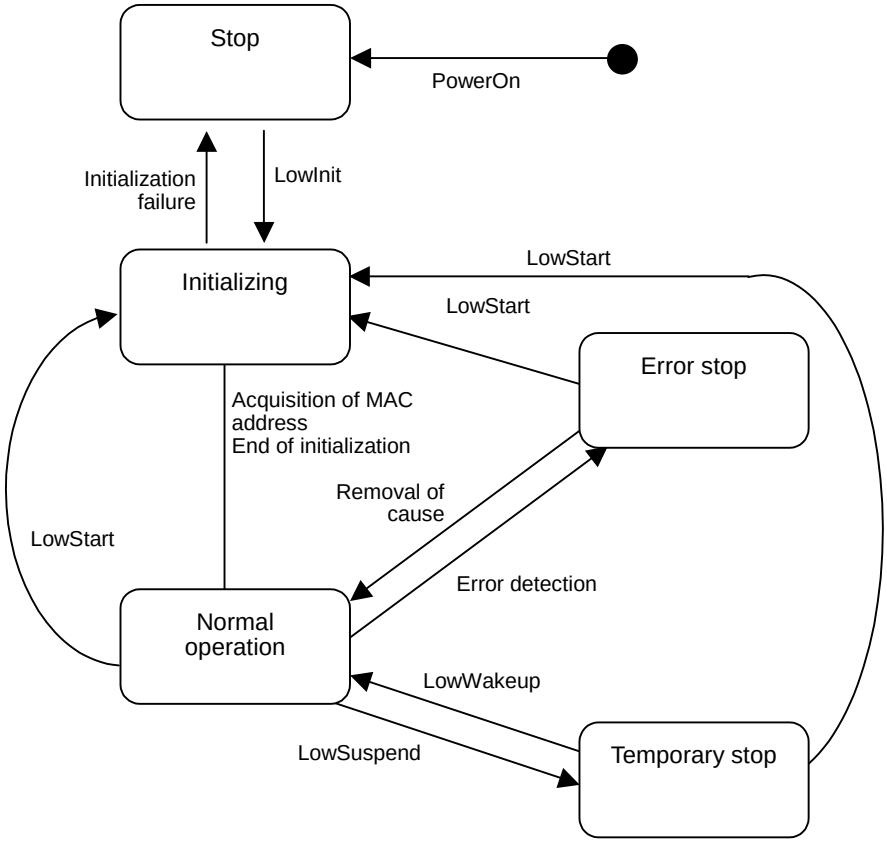
The function names used in Sections 4.7.1 to 4.7.6 correlate to those used in the state transition diagram shown below.

4.7.1 Basic concept

This subsection classifies the discrete lower-layer communication software status as shown below, and provides an outline of the sequence for each status.

① Stop status
② Initialize processing status
③ Normal operation status
④ Error stop status

The following figure shows the state transition for each state. In the figure, all English terms other than “PowerOn” are tentative designations for individual lower-layer communication interface services and are not official terms.



4.7.2 Stop status

Stop status signifies a state in which lower-layer communication software operations are not performed. This status is provided immediately after Power On. An outline of the processing immediately after state transition and an outline of the individual lower-layer communication interface services that the stop status receives, and related processing, are described below.

(1) Status acquisition service (LOWGetStatus) acquisition processing

When “Status acquisition service” is called via an individual lower-layer communication interface, “Stopping” is returned as the status.

(2) Initialization acquisition service (LowInit) acquisition processing

When “Initialization service” is called via an individual lower-layer communication interface, the transition is made to the initialized status. At this time, a response to “Initialization service” is returned immediately or after completion of the initialize processing. This is specified in the software mounting specifications but is not specified here.

(3) Lower-layer communication software type acquisition service (LowGetDevID) processing.

When the “lower-layer communication software type acquisition service” is called via an individual lower-layer communication interface, the lower-layer communication software type will be returned.

Triggers for state transition are as follows:

(1) Trigger for a shift to the “initialization processing in progress” state:

Initialization services (LowStart, LowInit)

4.7.3 Initialize processing status

Initialize processing status signifies that the lower-layer communication software is initialized.

An outline of the processing immediately after state transition and an outline of the individual lower-layer communication interface services that the initialize processing status receives, and related processing, are described below.

(1) Outline of initialize processing

A unique MAC address is obtained in the subnet. In particular, when the address is not fixed by the DIP switch as a product, the startup sequence processing specified in “4.6.3 Communication sequence” is executed to obtain a unique MAC address in the subnet.

(2) Status acquisition service (LOWSetStatus) acquisition processing

When “Status acquisition service” is called via an individual lower-layer communication interface, “Initialize processing” is returned as the status.

(3) Lower-layer communication software type acquisition service (LowGetDevID) processing

When the “lower-layer communication software type acquisition service” is called via an individual lower-layer communication interface, the lower-layer communication software type will be returned.

(4) Transition trigger to the initialization completion stop status

When initializing processing, including necessary buffer clearing after obtaining the MAC address, transition is set to “Normal operation status”.

(5) Transition trigger to stop status

This transition is caused by initialization failure.

4.7.4 Normal operation status

Normal operation status signifies the state in which data is transmitted to or received from a transmission medium as the primary function of the lower-layer communication software. An outline of the processing immediately after state transition and an outline of the individual lower-layer communication interface service that the normal operation status receives, and related processing, are described below.

(1) Outline of normal operation

Accepts a call for the individual lower-layer communication interface service from the Protocol Difference Absorption Processing Block and executes the specified processing, including data transmission. In addition, the data to be exchanged on the transmission line is received, and it is determined at the MAC address level whether or not this data is addressed to the self address. The received data is delivered to the Protocol Difference Absorption Processing Block through the individual lower-layer communication interface service.

(2) Status acquisition service (LOWGet Status) acquisition processing

Returns multiple operation status such as “Normally operating,” “Data transmitting,” and “Data receiving” when “Status acquisition service” is called via an individual lower-layer communication interface. At least these three status types shall be distinguished as a return value.

(3) Physical address acquisition service (LowGetAddress) acquisition processing

When the “physical address acquisition service” is called via an individual lower-layer communication interface, the MAC address will be returned.

(4) Profile data acquisition service (LowGetProData) acquisition processing

Performs processing to respond with the content of the property as the profile information specified in the “individual lower-layer communication software profile object.”

The lower-layer communication software implementation requirements shall apply to this service. There is no property-specific requirement for this particular service; however, it is recommended that the individual lower-layer communication interface specifications stated in Part 6 be satisfied.

(5) Message transmission service (LowSendData) acquisition processing

When the “message transmission service” is called via an individual lower-layer communication interface, processing will be performed to send the message provided, in accordance with the extended HBS communication protocol. Whether or not the response is to be sent in synchronization with the “message transmission service” shall be software implementation-dependent and is not specified here. However, it is recommended that the individual lower-layer communication interface specifications stated in Part 6 be satisfied.

(6) Message reception service (LowReceiveData) acquisition processing

When the “message reception service” is called via an individual lower-layer communication interface and there is a received message, the received message will be delivered as the response. When the “message reception service” is called via an individual lower-layer communication interface and there is no received message, a “No Received Message” message shall be returned as the response. However, a system in which the lower-layer communication software notifies the reception of a message via an individual lower-layer communication interface and delivers the message is also acceptable (The lower-layer communication software implementation specifications shall be taken into consideration).

(7) Suspension service (LowSuspend) acquisition processing

When the “suspension service” is called via an individual lower-layer communication interface while message transmission processing is in progress, a shift will be made to the “suspension” state after completion of the transmission processing. When the “suspension service” is called via an individual lower-layer communication interface while message reception processing is in progress, the processing will be terminated immediately and a shift will be made to the “suspension” state.

(8) Initialization service (LowStart, LowInit) acquisition processing

When the “initialization service” is called via an individual lower-layer communication interface while message transmission processing is in progress, a shift will be made to the “initialization processing” state (warm start processing state or cold start (2)) after completion of the transmission processing. When the “initialization service” is called via an individual lower-layer communication interface while message reception processing is in progress, the processing will be terminated immediately and a shift will be made to the “initialization processing” state (warm start processing state or cold start (2)).

(9) Stop service (LowHalt) acquisition processing

When the “stop service” is called via an individual lower-layer communication interface while message transmission processing is in progress, a shift will be made to the “stop” state after completion of the transmission processing. When the “stop service” is called via an individual lower-layer communication interface while message reception processing is in progress, the processing will be terminated immediately and a shift will be made to the “stop” state.

(10) Lower-layer communication software type acquisition service (LowGetDevID) processing

When the “lower-layer communication software type acquisition service” is called via an individual lower-layer communication interface, the type of lower-layer communication software will be returned.

(11) Transition trigger to error stop status

If received data remains without being read by the high-order software (Protocol Difference Absorption Processing Block software) or if an operation error of the high-order communication software is notified, transition is set to the error stop status.

4.7.5 Error stop status

Error stop status signifies a state in which a high-order software error is detected or an internal error is detected individually. An outline of the processing in the error stop status and an outline of the individual lower-layer communication interface services that the error stop status receives, and related processing, are described below.

(1) Outline of error stop status

In the error stop status, an error response is returned for services other than the services (listed below) of the individual lower-layer communication interface services from the Protocol Difference Absorption Processing Block. For data transmission/reception, receive processing is performed, but NAK (error existence code) shall be returned as the response for one data and the received data shall be abandoned.

(2) Status acquisition service (LowGetStatus) acquisition processing

When the “Status acquisition service” is called via an individual lower-layer communication interface, “Error stop” is returned as the status.

(3) Initialization service (LowStart, LowInit) acquisition processing

When the “initialization service” is called via an individual lower-layer communication interface and it is possible to perform the processing, a response will be returned and a shift will be made to the “initialization processing” state (warm start processing state or cold start (2)).

(4) Stop service (LowHalt) acquisition processing

When the “stop service” is called via an individual lower-layer communication interface while message transmission processing is in progress, a shift will be made to the “stop” state after completion of the transmission processing. When the “stop service” is called via an individual lower-layer communication interface while message reception processing is in progress, the processing will be terminated immediately and a shift will be made to the “stop” state.

(5) Lower-layer communication software type acquisition service (LowGetDevID) processing

When the “lower-layer communication software type acquisition service” is called

via an individual lower-layer communication interface, the type of lower-layer communication software will be returned.

(6)Transition trigger to normal operation status

When an internally recognized error is removed, a return is made to the normal status. Details of error recognition and details of error removal recognition shall be provided in the product specifications but are not specified here.

4.7.6 Suspension status

Suspension status signifies the state in which lower-layer software operation is suspended. In this state, no service processing is executed with the exception of some services (described below) of the individual lower-layer communication interface, and lower-layer communication processing is not performed at all. An outline of the processing in the suspension status and an outline of the individual lower-layer communication interface services that the suspension status receives, and related processing, are described below.

(1) Outline of suspension status

In the error stop status, an error response is returned for services other the services (listed below) of the individual lower-layer communication interface from the Protocol Difference Absorption Processing Block. For data transmission and reception, neither transmit processing nor receive processing is performed.

(2) Status acquisition service (LowGetStatus) acquisition processing

When the “Status acquisition service” is called via an individual lower-layer communication interface, “Suspended” will be returned as the status.

(3) Initialization service (LowStart, LowInit) acquisition processing

When the “initialization service” is called via an individual lower-layer communication interface and it is possible to perform the processing, a response will be returned and a shift will be made to the “initialization processing” state (warm start processing state or cold start (2)).

(4) Stop service (LowHalt) acquisition processing

When the “stop service” is called via an individual lower-layer communication interface while message transmission processing is in progress, a shift will be made to the “stop” state after completion of the transmission processing. When the “stop service” is called via an individual lower-layer communication interface while message reception processing is in progress, the processing will be terminated immediately and a shift will be made to the “stop” state.

(5) Operation restart instruction service (LowWakeUp) acquisition processing

When the “operation restart instruction service” is called via an individual lower-layer communication interface, a response will be returned and a shift will be made to the “normal operation” state.

Appendix 4.1 Documents Cited

(1) "EIAJ ET-2101 Home Bus System" published by JEITA

JEITA, General Affairs Department (Service Center) TEL: 03-3518-6422

(2) "EIAJ ET-2101 Home Bus System (Supplement)" published by JEITA

JEITA, General Affairs Department (Service Center) TEL: 03-3518-6422

(3) "EIAJ-RC-5202 Information Sockets for Home Bus System" published by JEITA

JEITA, General Affairs Department (Service Center) TEL: 03-3518-6422

Appendix 4.2 Details of Command Specifications

1. Reset command
   (1) OPC code: x ‘80’
   (2) OPR code: None

2. Communication stop request command
   (1) OPC code: x ‘8C’
   (2) OPR code: None

3. Communication stop response command
   (1) OPC code: x ‘8D’
   (2) OPR code: None
   (3) Other: Reception and response of communication stop request command

4. Communication start request command
   (1) OPC code: x ‘8E’
   (2) OPR code: None

5. Communication start response command
   (1) OPC code: x ‘8F’
   (2) OPR code: None
   (3) Other: Reception and response of communication stop request command

6. OK command
   (1) OPC code: x ‘92’
   (2) OPR code: None

7. MG command
   (1) OPC code: x ‘93’
   (2) OPR code: None

8. Startup start command
   (1) OPC code: x ‘A0’
   (2) OPR code: Self MAC address (1 byte)

9. Startup check command
   (1) OPC code: x ‘A1’
   (2) OPR code: MAC address of check destination (1 byte)
   (3) Other: When the controller exists, it receives and transmits the startup start command.

10. Startup completion command
   (1) OPC code: x ‘A2’
   (2) OPR code: MAC address of self (1 byte)
   (3) Other: Command is transmitted when startup is completed (completion of MAC address acquisition).

11. Loopback request command
   (1) OPC code: x ‘A8’
   (2) OPR code: Optional (254 bytes max)

12. Loopback response command
   (1) OPC code: x ‘A9’
   (2) OPR code: Contents set in OPR of loop request command
   (3) Other: Response command to loopback request command

13. Version request command
   (1) OPC code: x ‘AA’
   (2) OPR code: None
   (3) Other: Request for communication driver software version of lower-layer transmission medium

14. Version response command
   (1) OPC code: x ‘AB’
   (2) OPR code: Version information (3 bytes)
   (3) Other: Response command to version request command

15. Manufacturer name request command
   (1) OPC code: x ‘AC’
   (2) OPR code: None
   (3) Other: Request for manufacturer of communication driver software of lower-layer transmission medium

16. Manufacturer name response command
    (1) OPC code: x ‘AD’
    (2) OPR code: Version information (3 bytes)
    (3) Other: Response command to manufacturer name request command

Chapter 5 IrDA Control Communication Protocol Specifications

5.1 System Overview

5.1.1 Overview

This section specifies the communication protocol using IrDA Control in ECHONET. Unlike conventional systems used for infrared remote control, IrDA Control is quick, responsive and capable of two-way communication. Primarily, this specification provides for two-way communication between PCs specified as hosts and peripheral devices such as a mouse or keyboard. A host can communicate simultaneously with up to eight peripherals.

Regarding the use of IrDA Control in ECHONET, an IrDA device that performs host operations is arranged as an ECHONET router, and devices that communicate with this ECHONET router by using infrared are configured as peripherals.

Figure 5.1 shows an application example. In this figure, the ECHONET router acts as a router connecting the IrDA Control subnet and another subnet. The peripherals are assumed to be various sensors that transmit detected information to the centralized control unit through the router. Up to eight sensors can be installed.

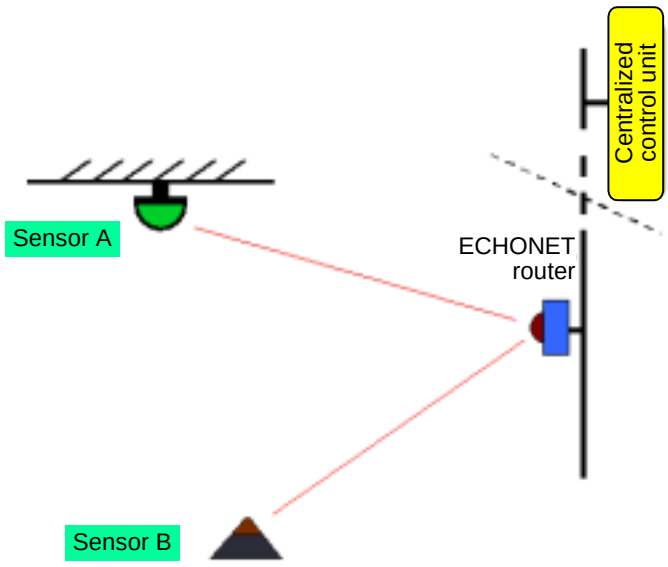


Fig. 5.1 IrDA Control Application Example

5.1.2 Scope of the specifications

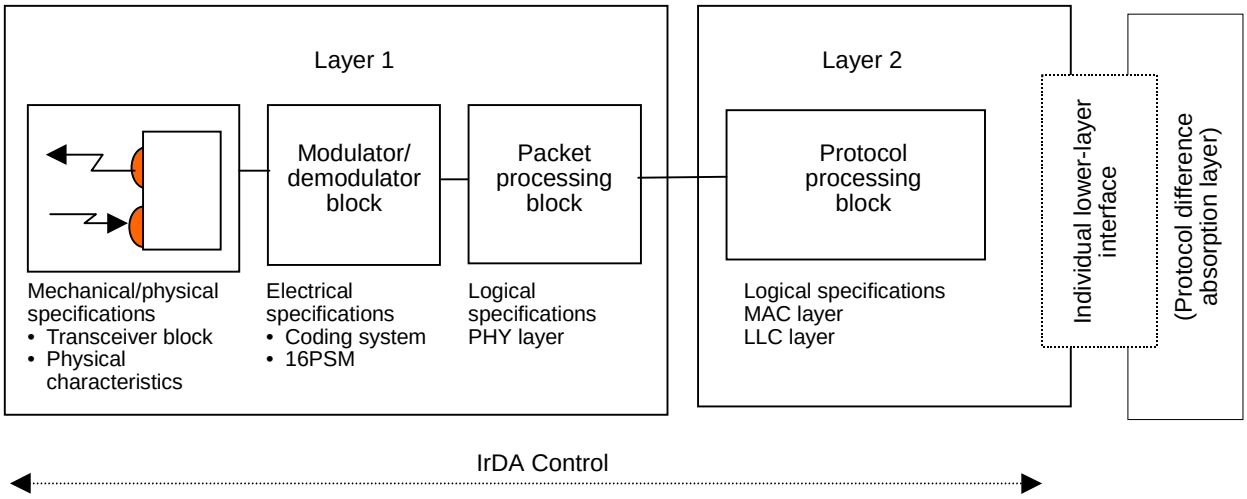


Fig. 5.2 IrDA Control Positioning in ECHONET

Figure 5.2 is a conceptual diagram that illustrates the positioning of IrDA Control in ECHONET. IrDA Control corresponds to Layers 1 and 2 of ECHONET. Layer 1 is a physical layer consisting of a transceiver block, modulator/demodulator block, and packet processing block (logical specifications). Layer 2 is a logical layer consisting of the MAC layer and LLC layer of IrDA Control. The MAC layer (Media Access Control layer) has functions to exchange property information (host address, host ID, and peripheral ID) between the host and peripherals, perform connections (binding: destination device numbering), perform scheduling for communication of 1:N (N = more than one), identify the destination device, and detect errors. The LLC layer (Logical Link Control layer) has functions of detecting missing packets and performing retransmission. Through data packet numbering and receipt check, the LLC layer provides a high-reliability communication line.

Layers 1 and 2 shall conform to the IrDA Control specification.

In addition, solutions to problems (address translation, broadcast processing, etc.) resulting from the accommodation of IrDA Control as an ECHONET transmission medium have been specified. These are described in “5.5 Basic Sequence”, “5.6 Accommodation Specification”, and “Part 7 ECHONET Communication Device Specifications, Chapter 6 IrDA Control Router”.

5.2 Mechanical/Physical Specifications

5.2.1 Characteristics

The following basic physical characteristics of IrDA Control are specified. These characteristics can be easily realized using a dedicated communication controller and a light receiving/transmitting element.

- Peak wavelength: 850 to 900 nm
- 16 PSM modulation system consisting of a 1.5 MHz sub-carrier
- Communication distance: 8 m standard
- Transmission rate: 75 kbps
- Response time: 138 ms standard

For details, see the IrDA Control Specification
(These specifications can be found at
http://www.irda.org/standards/specification.asp.)

5.2.2 Topology

The network shape (topology) of an IrDA Control subnet is illustrated below. The host functions as a router. Up to 8 Ir terminals operating as peripherals can simultaneously communicate with the router.

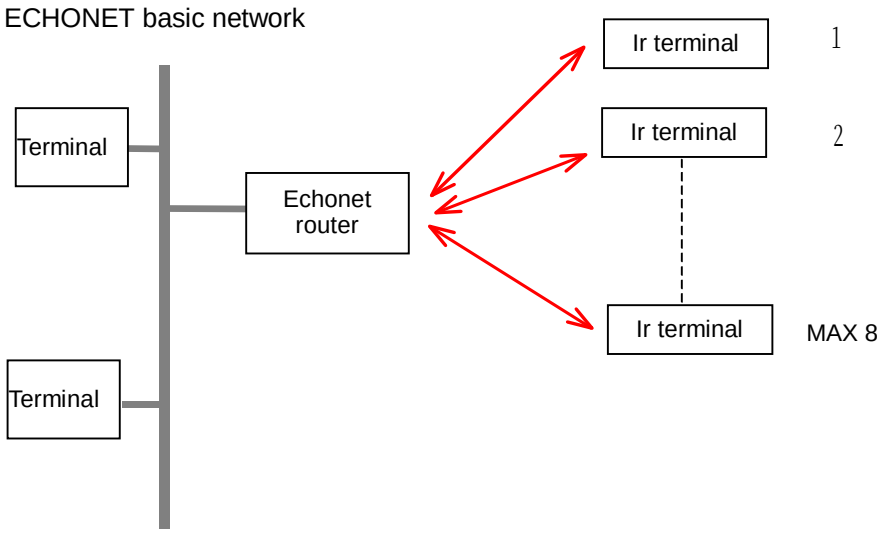


Fig. 5.3 Subnet Topology Using IrDA Control

5.3 Electrical Specifications

5.3.1 Coding system

The IrDA Control system uses the 16PSM system for data coding. Accordingly, there are 16 waveforms to be defined as 16PSM data symbols. There is a 4-bit set associated with each of the 16 symbol values. This is specified as a data bit set (DBS). The following table shows DBSs that are associated with the 16 symbols.

Table 5.1 16PSM Data Symbol Table

Data value (Hex)	Data bit set (DBS)	16PSM data symbol
0x0	0 0 0 0	1 0 1 0 0 0 0 0
0x1	0 0 0 1	0 1 0 1 0 0 0 0
0x2	0 0 1 0	0 0 1 0 1 0 0 0
0x3	0 0 1 1	0 0 0 1 0 1 0 0
0x4	0 1 0 0	0 0 0 0 1 0 1 0
0x5	0 1 0 1	0 0 0 0 0 1 0 1
0x6	0 1 1 0	1 0 0 0 0 0 1 0
0x7	0 1 1 1	0 1 0 0 0 0 0 1
0x8	1 0 0 0	1 1 1 1 0 0 0 0
0x9	1 0 0 1	0 1 1 1 1 0 0 0
0xA	1 0 1 0	0 0 1 1 1 1 0 0
0xB	1 0 1 1	0 0 0 1 1 1 1 0
0xC	1 1 0 0	0 0 0 0 1 1 1 1
0xD	1 1 0 1	1 0 0 0 0 1 1 1
0xE	1 1 1 0	1 0 1 0 0 1 0 1
0xF	1 1 1 1	1 1 1 0 0 0 0 1

A coding example is shown in Fig. 5.4.

Example

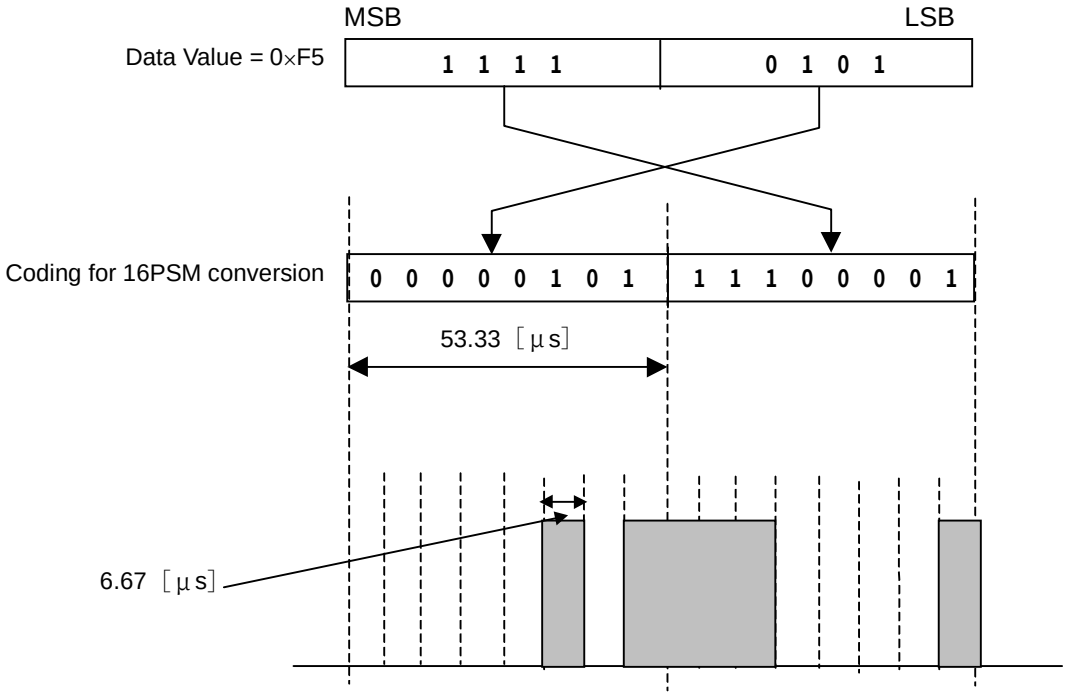


Fig. 5.4 Coding Example

For details, refer to the IrDA Control Specifications.

5.4 Logical Specifications

5.4.1 Overall data structure image

The relationship between the IrDA Control data structure and ECHONET data is described below. The ECHONET data is stored as an LLC frame payload of IrDA Control, the header of the MAC layer is added, and the date is then transmitted as a PHY layer packet. Details of each layer are described in the following sections.

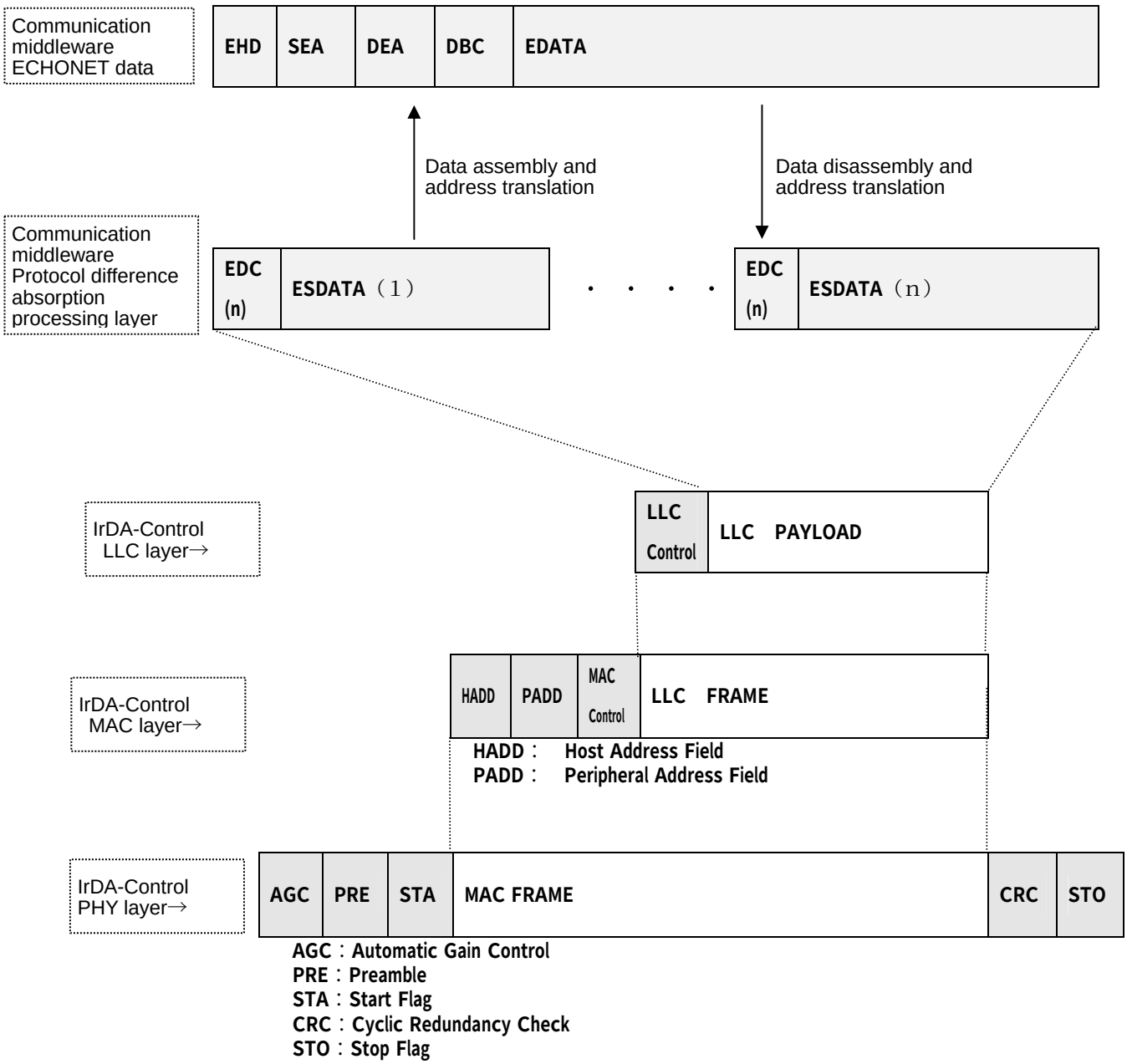
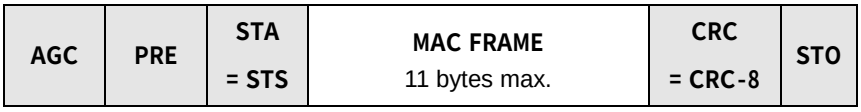


Fig. 5.5 Relationship Between Layers

5.4.2 Layer 1 (PHY layer)

IrDA Control is divided into two types of packet structure based on the MAC frame size.

1) Short packet



2) Long packet

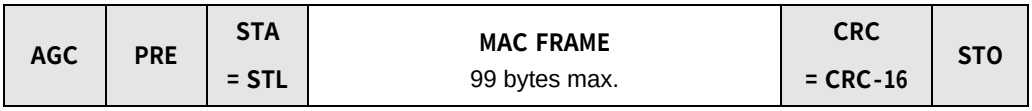


Fig. 5.6 Layer 1 Packet Structure

An outline of the structure is described below. For details, see the IrDA Control Specifications.

(1) AGC (Automatic Gain Control)
Signal for infrared receiver sensitivity adjustment.
The symbol is 1111.

(2) PRE: (Preamble)
Used for clock synchronization.
The symbol is 0101010101.

(3) STA (Start Flag)
Performs synchronization with the symbol.
For long packets: STL (0100101101) is used.
For short packets: STS (0100101100) is used.

(4) MAC FRAME
There are two types of frames: short and long.
Data is encoded by 16 PSM.

(5) CRC (Cyclic Redundancy Check)
Used for error detection.
For short packets: CRC-8 is used.
For long packets: CRC-16 is used.

(6) STO (Stop Flag)
Indicates the end of a packet.
The symbol is 01001011.

5.4.3 Layer 2 (MAC layer)

The contents of the MAC frame are as follows.

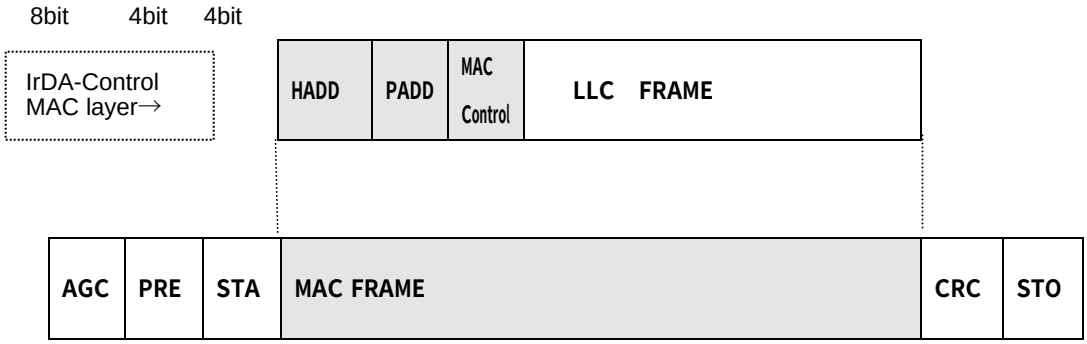


Fig. 5.7 MAC Frame Structure

(1) HADD (Host Address field)
MAC address of the IrDA Control host.
This address consists of 8 bits and is fixed at 0x01. The MAC address is associated with the ECHONET Node ID at 1:1.

(2) PADD (Peripheral Address field)
MAC address of the IrDA Control peripheral.
This address consists of 4 bits. The peripheral address is given by the host at each bind execution (as described later).

(3) MAC Control (MAC Control field)
Four bits are given.
Communication control is defined as described below.

Table 5.2 Details of MAC Control

		Meaning	1	0
Frame from host	D7	Packet direction	1	
	D6	Bind timer	Reset	
	D5	Long packet	Possible	Add
	D4	Haling	○	×
Frame from peripheral	D7	Packet direction	0	
	D6	Polling request	Yes	No
	D5	Reserved	—	—
	D4	Reserved	—	—

In the above table, "D7" indicates bit 7. This chapter uses "D*" to represent bits in accordance with the conventions of the IrDA standard.

5.4.4 Layer 2 (LLC layer)

The contents of the LLC layer are as follows:

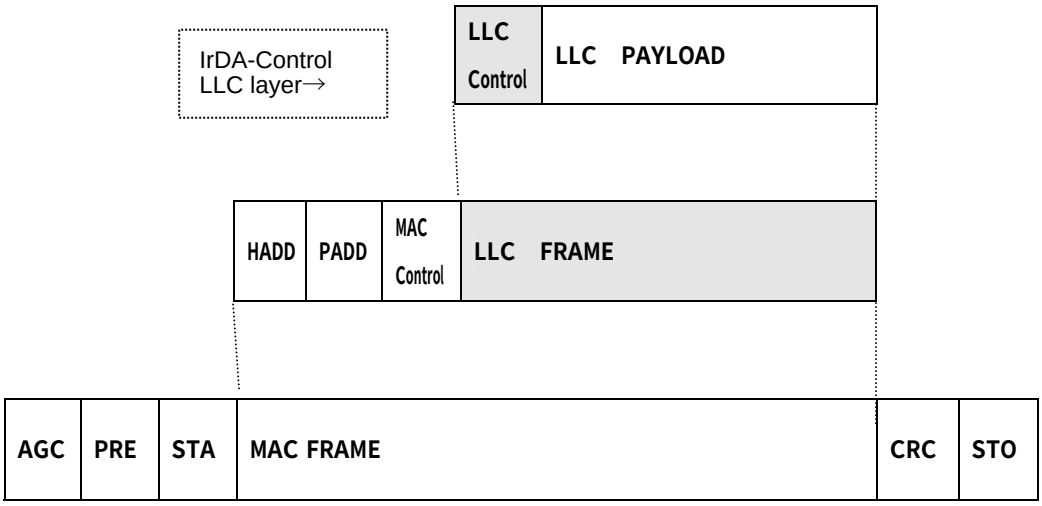


Fig. 5.8 LLC Frame Structure

Table 5.3 Details of LLC Control

LLC Control							
D7	D6	D5	D4	D3	D2	D1	D0
Reserve	Endpoint		Reserve	Packet Type Code			

(1) Packet Type Code

Implements the types (request to receive, data, ACK, NAK, etc.) of the LLC frame and frame sequence numbers.

For details, refer to the IrDA Control Specification.

(2) Endpoint

Represents the type of Pipe serving as a logical communication channel.

Table 5.4 shows Pipe types and Table 5.5 shows the relationship between Endpoint values and Pipe types.

Table 5.4 Pipe Types

Pipe type	Usage
Control Pipe	For transmission of host commands and device requests
IN Pipe	Used for data from devices to the host
OUT Pipe	Used for data from the host to devices

Table 5.5 Relationship Between Endpoints and Pipe Types

Endpoint	Pipe type
00	Control Pipe
01	IN Pipe
10	OUT Pipe
11	IN Pipe or OUT Pipe

An “Out Packet” is a packet moving from the host toward a peripheral. An “In Packet” is a packet moving from a peripheral toward the host.

5.4.5 Packet accommodation

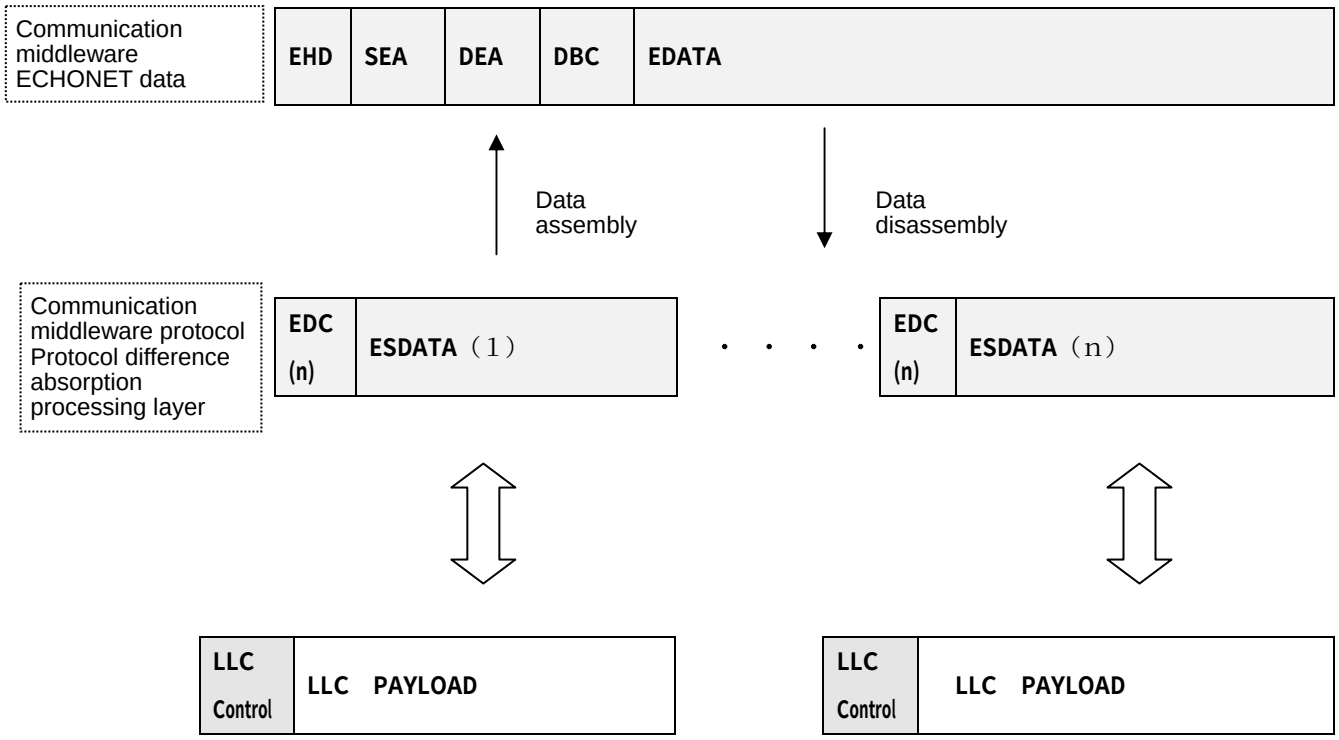


Fig. 5.9 Relationship Between LLC Payload and ECHONET Data

- The total length of ESDATA (1) to ESDATA (n) is 262 bytes.
- EDC is 1 byte.
- LLC PAYLOAD can accommodate up to 96 bytes.

5.5 Basic Sequence

5.5.1 Basic concept

In this section, the lower-layer communication software status for IrDA Control protocol is classified as follows, and an outline of the sequence for each status is provided:

① Stop
② Cold start
③ Warm start
④ Communication stop (enumeration completed)
⑤ Normal operation (bind completed)
⑥ Error stop
⑦ Suspension

The following diagram shows the state transition of items ① to ⑦ in the lower-layer communication software for IrDA Control.

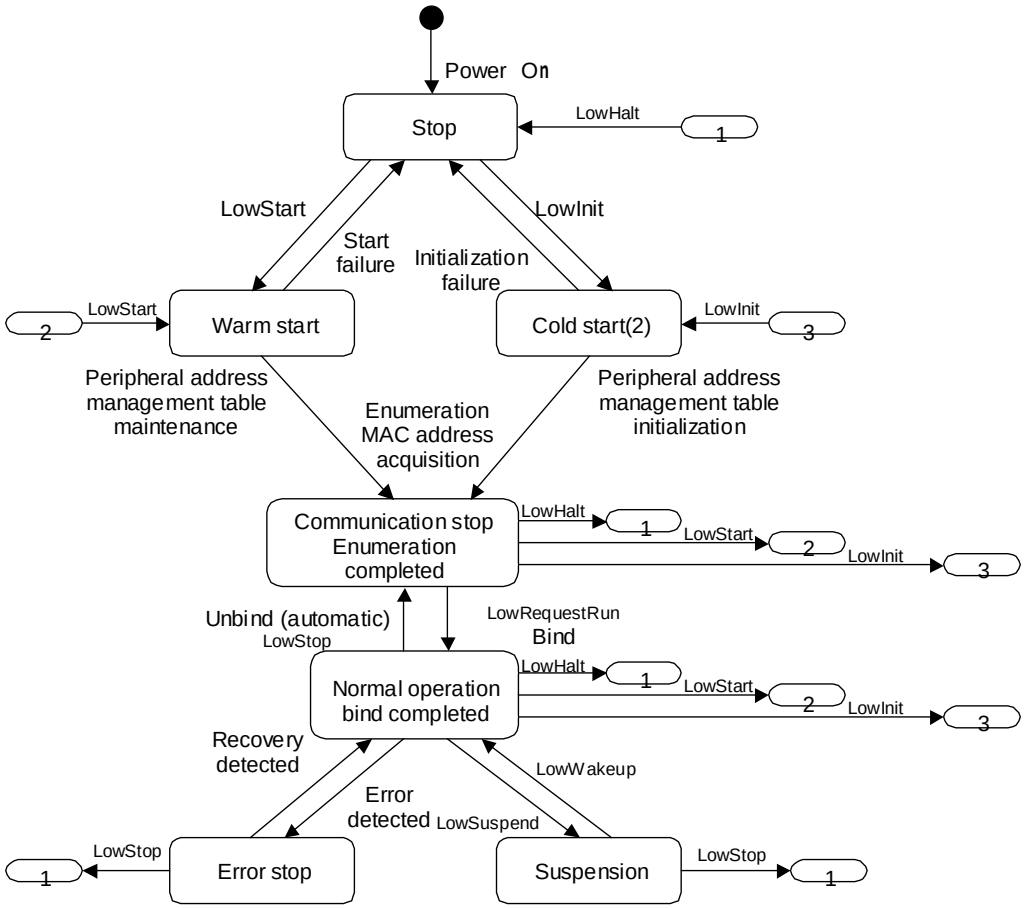


Fig. 5.10 Sequence Transition Diagram

5.5.2 Stop status

Stop status signifies a state in which no lower-layer communication software operations are performed. This status is provided immediately after Power On. An outline of the processing immediately after state transition and an outline of individual lower-layer communication interface services that the stop status receives, and related processing, are described below.

(1) Trigger and action
Waits for an individual lower-layer communication interface service.
(2) Status acquisition service (LowGetStatus)
Returns LOW_STS_STOP as status.

The triggers for state transition are as follows:

(1) Transition trigger to the cold start
This transition is caused by the initialization request service (LowInit).
(2) Transition trigger to the warm start
This transition is caused by the reset request service (LowStart).

5.5.3 Cold start

In the cold start state, the lower-layer communication software is initialized. When the IrDA Control protocol is used, the peripheral address management table is initialized to wait for the following individual lower-layer communication interface services:

(1) Trigger and action
   ① Initializes the software.
   ② Initializes the peripheral address management table.
   ③ Acquires a MAC address.
   The host's MAC address shall be the same as HADD. For the peripheral MAC address, after the host-to-peripheral information exchange procedure called "enumeration" is performed, the host uses the obtained information to assign a virtual MAC address (address correlating to Node ID on a 1:1 basis) and writes the virtual MAC address in the peripheral address management table it manages (details provided below).

(2) Status acquisition service (LowGetStatus)
   Returns LOW_STS_INIT as the status.

Triggers for state transitions are shown below:

(1) Transition trigger to communication stop status
   This transition is caused by the completion of enumeration.
(2) Transition trigger to stop status
   This transition is caused by an initialization failure.

An outline of enumeration is shown below.

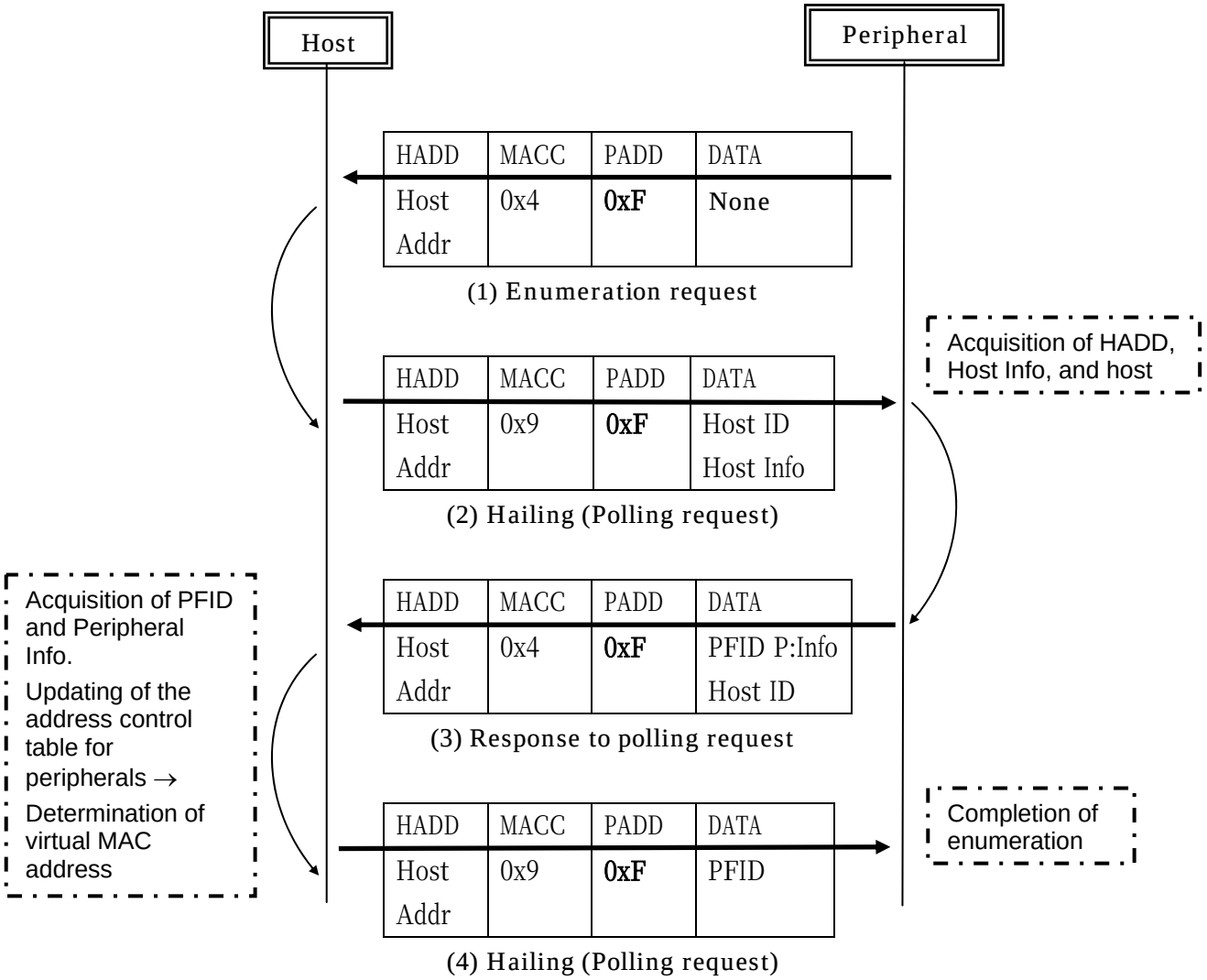


Fig. 5.11 Enumeration Procedure

(1) A peripheral makes an enumeration request.
MACC = 0x4 is a polling request command. Here, it is used as an enumeration request.

(2) The host transmits a polling request.
At this time, both Host ID and Host information are sent as data. PADD = 0xF is a special address for enumeration. MACC = 0x9 is a command for hailing (polling request).
* With this operation, the peripheral obtains both Host ID and Host information.

(3) The peripheral transmits a response of a polling request.
At this time, the peripheral ID, peripheral information, and host ID are transmitted to the host.
To the peripheral to which enumeration was performed, the host gives an 8-bit virtual MAC address and updates the "peripheral ID", "virtual MAC address", and the address control table that controlled the three parties of "PADD" to be determined by the binding operation.

(4) The host completes initialization by informing the peripheral of the "virtual MAC address" specified in (3).

Table 5.6 Address Control Table for Peripherals

	Peripheral ID (32bit) At delivery from the factory	Virtual MAC address (8bit) (= NodeID)	PADD (4bit) To be issued for bind execution
	A	B	C
Peripheral ①	xxxxxx-xxxx-xxxxxxxxxx	0x01	For example, (0x2)
Peripheral ②	xxxxxx-xxxx-xxxxxxxxxx	0x02	For example, (0x6)
Peripheral ③	xxxxxx-xxxx-xxxxxxxxxx	0x03	For example, (0x4)
・ ・ ・ ・	・ ・ ・ ・	・ ・ ・ ・	・ ・ ・ ・
Peripheral ⑧	xxxxxx-xxxx-xxxxxxxxxx	0x08	For example, (0xA)
(Peripheral ⑨) ・ ・ ・	xxxxxx-xxxx-xxxxxxxxxx	0x08	For example, (0xB)

A → B: Determine "virtual MAC address" by enumeration:
This virtual MAC address is associated with ECHONET Node ID at 1:1. The relationship between A and B is held. A certain relationship is held except for reset processing.
Determine PADD by bind execution:

B → C: PADD is cleared by unbind execution. Accordingly, PADD is given for each bind execution, so it does not always take the same value.

The host can simultaneously bind up to 8 units.

5.5.4 Warm start

In the warm start state, the lower-layer communication software is initialized without initializing the peripheral address management table managed by the host, and the following individual lower-layer communication interface services are awaited:

(1) Trigger and action

① Initializes the software.

② Acquires a MAC address.

The host's MAC address shall be the same as HADD. For the MAC address of a peripheral, after the enumeration procedure is performed, the peripheral address management table managed by the host is searched for the peripheral's peripheral ID that was derived from enumeration.

If no matching peripheral is found in the peripheral address management table and the peripheral address management table is not full, the host assigns a new virtual MAC address and updates the peripheral address management table.

If no matching peripheral is found in the peripheral address management table and the peripheral address management table is full, a start failure results and the status changes to stop status.

If any matching peripheral is found in the peripheral address management table, the associated virtual MAC address written in the management table is used as the peripheral's MAC address.

(2) Status acquisition service (LowGetStatus)

Returns LOW_STS_INIT as the status.

Triggers for state transitions are shown below:

(1) Transition trigger to communication stop status

This transition is caused by the completion of enumeration.

(2) Transition trigger to stop status

This transition is caused by an initialization failure or occurs when the maximum management count of the peripheral address management table is exceeded.

5.5.5 Communication stop status

In the communication stop status, an operation start request from the communication middleware is awaited after the completion of lower-layer communication software initialization. This section outlines the processing to be performed upon a state transition, describes the individual lower-layer communication interface services acceptable during communication stop status, and gives an overview of the associated processes.

(1) Trigger and action
Waits for an individual lower-layer communication interface service.
(2) Status acquisition service (LowGetStatus)
Returns LOW_STS_INIT as the status.
(3) Physical address acquisition service (LowGetMacAddress)
Returns HADD for the host or “virtual MAC address” for a peripheral.
(4) Lower-layer communication software type acquisition service (LowGetDevID)
Returns the lower-layer communication software type.
(5) Profile data acquisition service (LowGetProData)
Returns profile data.

Triggers for state transitions are shown below:

(1) Transition trigger to normal operation
This transition is caused by the operation start service (LowRequestRun).
The transition caused by LowRequestRun varies depending on whether it is initiated by the IrDA host or IrDA peripheral. If an IrDA peripheral operation is invoked by LowRequestRun, a bind is performed immediately so that the status changes to the normal operation state. If an IrDA host operation is invoked by LowRequestRun, on the other hand, the transition to the normal operation state occurs only when a means of communication start from the host is incorporated in compliance with Section 5.6 “Accommodation Specification”.
(2) Transition trigger to stop status
This transition is caused by the stop service (LowHalt).
(3) Transition trigger to the cold start
This transition is caused by the initialization request service (LowInit).
(4) Transition trigger to the warm start
This transition is caused by the reset request service (LowStart).

The bind procedure is shown below.

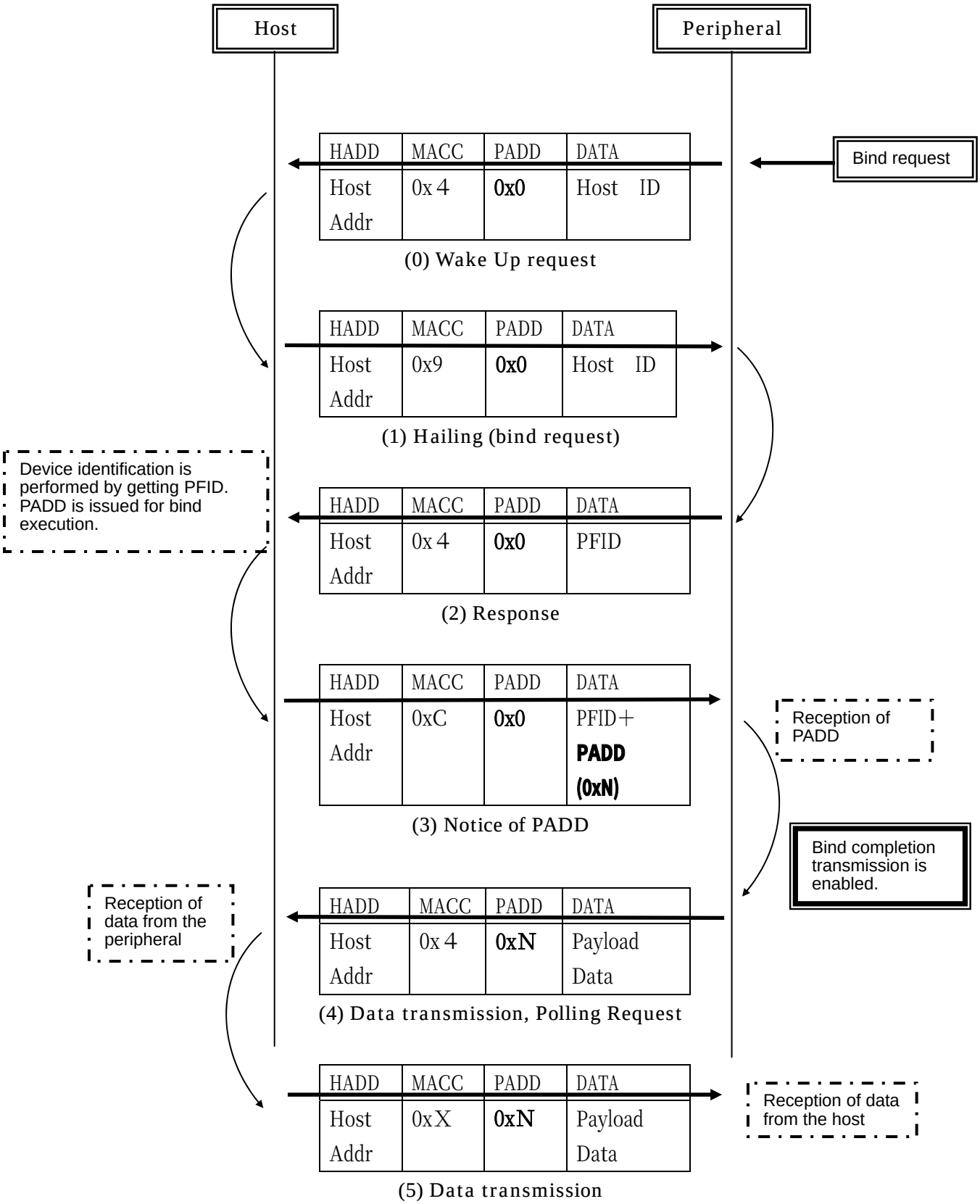


Fig. 5.12 Bind Procedure

However, procedure (0) is required only when the host is in the sleep status.

(0) The peripheral transmits a Polling Request to the host.
Here, HADD is the host address and PADD is 0x0. With this operation, the host in the sleep status starts the following operation (1).

(1) Host transmits Hailing.
At this time, it sends Host ID as data. PADD = 0x0 is a special address for bind execution. MACC = 0x9 is Hailing.

(2) Peripheral receives hailing command from host and transmits Polling Request in response.
At this time, Peripheral ID is transmitted to the host.

(3) The host gives a 4-bit HADD to the peripheral that received Peripheral ID.
This value is changed at each bind execution. The host generates the control table for peripheral ID and ECHONET NODE ID at enumeration. The PADD to be given by bind execution is added to the corresponding control table so that the peripheral can be identified uniquely.

(4) The host transmits the PADD set in (2) above to the peripheral and resets the bind timer.
With this, bind execution is completed.

5.5.6 Operation status

The operation status signifies a state in which data is transmitted to or received from a transmission medium as the primary function of the lower-layer communication software. An outline of the processing immediately after state transition and an outline of the individual lower-layer communication interface services that the operation status receives, and related processing, are described below.

(1) Outline of processing immediately after state transition
Waits for the individual lower-layer communication software service.

(2) Status acquisition service (LowGetStatus)
Returns LOW_STS_RUN as status.

(3) Physical address acquisition service (LowGetMacAddress)
Returns HADD (= MAC address) for the host.
Returns “Virtual MAC address” for a peripheral.

(4) Profile data acquisition service (LowGetProData)
Returns profile data.

(5) Data transmission service (LowSendData)
Translates the received protocol difference absorption processing block data into lower-layer communication software data and outputs it to the transmission medium.

(6) Data reception service (LowRecvData)
Translates the lower-layer communication software data received from the transmission medium into protocol difference processing block data and outputs it to the Protocol Difference Absorption Processing Block.

(7) Lower-layer communication software type acquisition service (LowGetDevID)
Returns the lower-layer communication software type.

Triggers for state transition are as follows:

(1) Transition trigger to stop status
This transition is caused by the end service (LowStop).
<Outline of end processing>
- Clears the bind status, abandons all parameters, and proceeds to the stop status.

(2) Transition trigger to communication stop status
This transition is caused by the end service (LowStop). Or, an unbind is performed (to undo a bind) after the elapse of a predetermined time in accordance with the IrDA Control protocol to invoke an automatic transition to the communication stop status.

(3) Transition trigger to warm start
This transition is caused by the reset request service (LowStart).

(4) Transition trigger to cold start
This transition is caused by the initialization request service (LowInit).

(5) Transition trigger to error stop status
This transition occurs when a lower-layer communication medium detects an error.

(6) Transition trigger to suspension status
This transition is caused by the lower-layer communication unit stop service (LowSuspend).

5.5.7 Error stop status

Error stop status signifies a state in which the operation is stopped by the occurrence of an error. An outline of the processing immediately after state transition and an outline of the individual lower-layer communication interface services that the initialization completion stop status receives, and related processing, are described below.

(1) Trigger and action
Performs error processing.

(2) Status acquisition service (LowGetStatus)
Returns LOW_STS_SUSPEND as the status.

(3) Lower-layer communication software type acquisition service (LowGetDevID)
Returns the lower-layer communication software type.

Triggers for state transition are as follows:

(1) Transition trigger to stop status
This transition is caused by the end service (LowHalt).

(2) Transition trigger to normal operation status
This transition is caused by removing the cause of the error.

5.5.8 Suspension status

Suspension status signifies a state in which the operation is paused by an instruction of the Communication Middleware. An outline of the processing immediately after state transition and an outline of the individual lower-layer communication interface services, and related processing, are described below.

(1) Trigger and action
Stops the operation of the lower-layer communication software.

(2) Status acquisition service (LowGetStatus)
Returns LOW_STS_SUSPEND as the status.

(3) Lower-layer communication software type acquisition service (LowGetDevID)
Returns the lower-layer communication software type.

Triggers for state transition are as follows:

(1) Transition trigger to normal operation status
This transition is caused by the operation restart service (LowWakeUp).

(2) Transition trigger to stop status
This transition is caused by the end service (LowHalt).

5.6 Accommodation Specifications

5.6.1 Relationship between host and peripherals

The only opposite party with which peripherals can communicate directly is the host. When one peripheral communicates with another peripheral or another subnet device, communication must be performed through the host.

5.6.2 Handling of individually specified messages within a subnet

When an IrDA Control communication medium is used, individually specified messages within a subnet are discarded by the “received message judgment process”, which is described in Part 2, Chapter 6 “ECHONET Communication Processing Block Processing Specifications”.

Therefore, when IrDA Control is used, the above problem shall be corrected by furnishing an appropriate bridge processing block between the “message reception/assembly process” and “message division/transmission process” of the preceding “Protocol Difference Absorption Processing Block”.

Before the release of Version 2.10, the above problem was avoided according to the stipulations set forth in Part 7 “ECHONET Communications Equipment Specifications”, Chapter 6 “IrDA Control Router”. After the release of Version 2.10, however, the use of a bridge process in the above “Protocol Difference Absorption Processing Block” is permitted.

5.6.3 Recommended conditions for host and peripherals

The IrDA Control standard does not permit the host to start communicating with peripherals that are completely initialized and stopped (not bound or unbound upon a bind timer timeout).

The ECHONET standard recommends that a means for bind maintenance be implemented by incorporating a means of starting communications from the host, a means of issuing a host’s request for the start of communications to a peripheral, or a function for transmitting a WakeUp request at regular intervals between the unbound host and peripherals.

However, this recommendation is void if the host does not need to start communications when, for instance, bidirectional remote control is used for peripherals.

5.6.4 Mandatory conditions for host and peripherals

It is imperative that a function for retaining a controller’s request in the host’s receive buffer for a predetermined period of time be implemented in situations where the host cannot communicate with a peripheral in response to a request message from another subnet controller, etc.

Detailed stipulations are provided in Part 7 “ECHONET Communications Equipment Specification”, Chapter 6, “IrDA Control Router”.

Chapter 6 LonTalk® Communication Protocol Specification

6.1 System Overview

The LonTalk® protocol conforms to the Open System Interconnection (OSI) reference model of the International Organization for Standardization (ISO) and supports Layer 1 to Layer 7. LonTalk itself can implement a perfect network protocol.

Figure 6.1 shows a typical node configuration formed with a Neuron® chip. Since protocol processing does not depend on transmission media, a wide variety of transmission media are supported, as indicated below.

- Twisted pair cable
- Radio frequency (radio wave)
- Infrared
- Coaxial cable
- Power line
- Optical cable

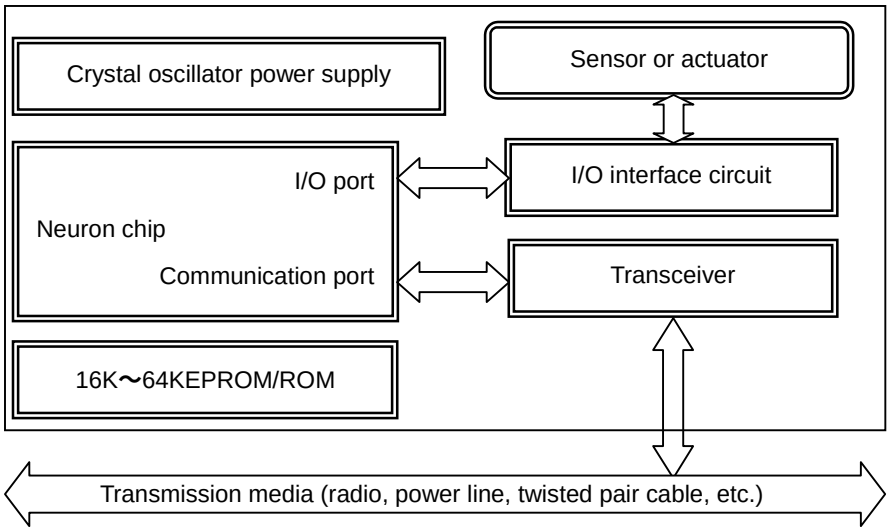


Fig. 6.1 Typical Node Configuration with Neuron Chip

Basically, use of the Neuron® chip permits free design of an individual transceiver, taking into consideration only the I/F with the MAC (Media Access Control) layer of the LonTalk® protocol. In other words, network processing in the transmission media controller subsequent to the transceiver need not be considered. Some transmission media may need special protocol processing because of legal requirements; the designer need only consider this point.

In ECHONET, the portion subsequent to the transceiver is regarded as Layer 1, which supervises transmit/receive and modulate/demodulate processing for the LonTalk® PPDU (Physical Protocol Data Unit). Layer 1 performs the processing equivalent to OSI reference model Layer 1 and Layer 2 individually using the specific protocol for each transmission medium. The LonTalk® protocol is positioned as lower-layer communication software to support the layers subsequent to the individual lower-layer communication interface and also as the equivalent to Layers 2 and 3, which perform basic communication processing. The ECHONET data (addresses, data) sent from the high-order layer is treated as LonTalk® protocol data in a lump. Explicit messages are used for communications. Versions 1.0, 2.0, 2.1 and 2.11 do not specify the use of network variables.

6.1.1 Organization of Chapter 6

Sections 6.1 to 6.5 summarize the specifications related to LonTalk® in ECHONET.

6.1 System Overview

6.2 Mechanical/Physical Characteristics

6.3 Electrical Characteristics

6.4 Logical Specifications

6.5 Basic Sequence

Sections 6.6 and after summarize the transceiver specifications. The item numbers in each section correspond to the section numbers in Section 6.1 to 6.5. ($X \geq 6$)

6.X.1 System overview

6.X.2 Mechanical/physical characteristics

6.X.3 Electrical characteristics

6.X.4 Logical specifications

6.X.5 Basic sequence

Additional specifications are provided in Sections 6.X.6 and after.

6.2 Mechanical/Physical Specifications

Nodes must use devices that can implement the LonTalk® protocol and must be based on the Neuron® chip. Legal requirements or other standards related to the mechanical/physical characteristics and specifications of cabinet, connector shape, cable, antenna, etc., are to be adhered to. The following specifications shall be provided as the ECHONET Standard for each transmission medium as required. Details are provided in 6.X.2 after Section 6.6.

- Connector shape
- Transmission media
- Topology

6.3 Electrical Characteristics

Regarding the electrical characteristics of peripheral devices, including the protocol circuit, the interface with devices implementing the LonTalk® protocol, including the Neuron® chip, shall be taken into consideration. Legal requirements or other standards related to electrical characteristics and specifications are to be adhered to. The following specifications shall be provided as the ECHONET Standard for each transmission medium, as required. Details are provided in 6.X.3 after Section 6.6.

- Electrical characteristics of transmission media
- Transmission rate
- Modulation system
- Transmitting/receiving sensitivity (level)

6.4 Logical Specifications

Layer 1 processing is performed by the transceiver, and Layer 2 and 3 processing is performed by the Neuron® chip. The Neuron® chip shall obtain the following transceiver operation status data, as required:

- READY: Transceiver is operating normally.
- BUSY: Transceiver performs transmit/receive processing or initialize processing.
- ERROR: Transceiver causes some errors.
- NO_ID: Node ID setting (updating) is required.

6.4.1 Layer 1

In the transceiver, the PPDU (Physical Protocol Data Unit) of LonTalk® is treated as the data portion of the communication format specified for each transmission medium. Basically, frames are converted into a data format such as that shown in Fig. 6.2 for communication. The header or footer is a general term for the preamble, address data, control code, etc. native to and specified for each transmission medium. Details are provided in 6.6.4 after Section 6.6. Processing native to the transmission media is described in and after 6.6.5.

Upon a request to send from the Neuron® chip or at the start of transmission, carrier sense and transmission timing adjustment (serving as transmission media protocol processing) are performed, and the transmission media modulate the transmission media communication basic data to perform communication. The receiving side deletes the header and footer (format substitution) from the transmission media communication basic data demodulated in the transmission media and transmits PPDU to the Neuron® chip. When the transceiver must stop new data transmission from the Neuron® chip during header/footer processing, it notifies the Neuron® chip of its BUSY status. When the transmission of new data is permitted, it clears BUSY and notifies READY.

Header	Data (LonTalk®PPDU)	Footer

Fig. 6.2 Transmission Media Communication Basic Data Format

The NODE_ID information received from the Protocol Difference Absorption Processing layer is reflected in AddEmt and Address. The following PDU is put into the Eccl.PDU area according to PDUFmt. Data content and format are performed automatically by LonTalk®. Accordingly, it is not necessary to consider the contents of the data structure, but they are described for reference. For details, see the LonTalk® Protocol Specification.

PPDU								
PPDU Header	NPDU							CRC
	Ver	PDUFmt	AddFmt	Len	Address	Domain	Encl.PDU	

Fig. 6.3 Layer 2 Data Format

- PPDU (Physical Protocol Data Unit)
  PPDU-Header: 9bit+ α (Bit synchronization signal + Byte synchronization one-bit length)
  CRC:16bit
- NPDU (Network Protocol Data Unit)
  Ver: 2bit Protocol Version
  PDUFmt: 2bit Encl.PDU is specified.
  00: Encl.PDU = TPDU

01: Encl.PDU = SPDU
10: Encl.PDU = AuthPDU (Not specified in Ver 1.0, 2.0, 2.1, 2.11)
11: Encl.PDU = APDU

AddFmt:	2bit	Specifies format of address data.
Len:	2bit	Specifies length of address data.
Address:		Neuron® chip address data 24 bits min. or 72 bits max.
Domain:	0bit	Neuron® chip domain: 0 (Not specified in Ver 1.0, 2.0, 2.1, 2.11)

Addressing method (in Neuron® chip)

- Broadcast
  - All nodes in the domain: 24 bits are used for address.
  - All nodes in the subnet: 24 bits are used for address.
- Multicast
  - All nodes in the group: 24 bits are used for address. (48 bits are used for ACK)
- Unicast
  - Specific logical node: 32 bits are used for address
  - Specific physical node: 72 bits are used for address (NeuronID)

Address length (in Neuron® chip)

- Domain: 0
- subnet: 8-bit
- Node: 8-bit (1 to 127: Effective set value is 1 to 126.)
- Group: 8-bit

The message code and address type are stipulated as follows:
Address length for subnet: 8 bits (ID: fixed at 0x01)

Outgoing message	Message code	Address type
ECHONET frame	0x04	BROADCAST
		SUBNET_NODE

6.4.2 Layer 3

This layer processes the Node ID and ECHONET data received from the Protocol Difference Absorption Processing layer. The address information is reflected in AddFmt and Address of Layer 2. The entire ECHONET data is placed in the DATA area of the APDU. When the Protocol Difference Absorption Processing layer divides the data,

EDC(n) + (ESDATA)(n) is placed in the DATA area. Here, data processing takes place n times. However, this is independent of the data content and format.

The data size that can be handled by Layer 3 varies depending on how the software is installed. Individual lower-layer communication interfaces notify the higher layer of the maximum buffer size in advance. The send buffer shall have a minimum value of 34 bytes, since the maximum value for an ECHONET message is 262 bytes and the maximum message division value is 8. Further, since the maximum data value that can be handled by a Neuron® chip is 228 bytes, the higher layer will be informed of a maximum processable data length of 228 bytes even when a 229-byte or larger send buffer is available. A 228-byte or larger receive buffer (255-byte) shall be furnished.

APDU					
APDU-Header	Data				
Destin&Type	EHD	SEA	DEA	EBC	EDATA

Fig. 6.4 Layer 3 Data Format (1)

APDU		
APDU-Header	Data	
Destin&Type	EDC(n)	ESDATA(n)

Fig. 6.5 Layer 3 Data Format (2)

6.5 Basic Sequence

This section describes the following items:

- State transition diagram
- Sequence explanation of each state in the state transition diagram

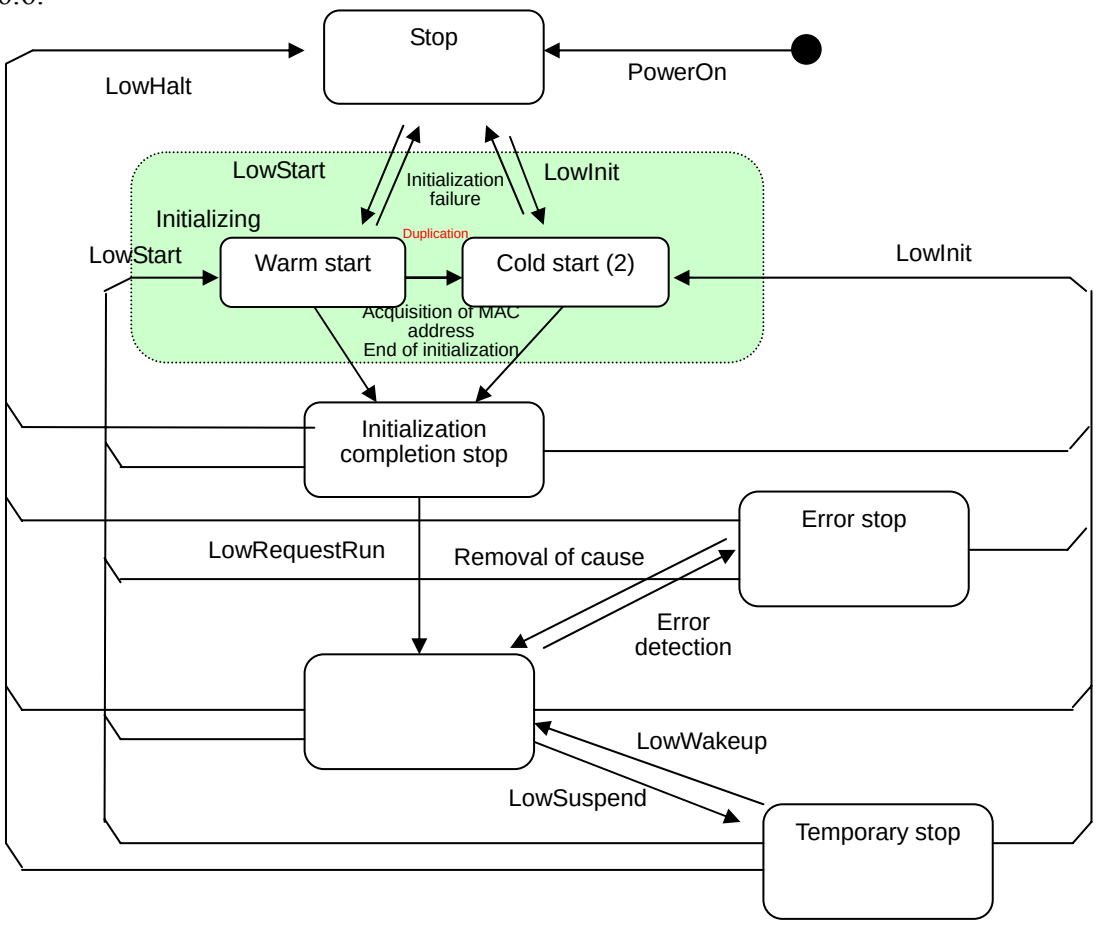
6.5.1 Basic concept

This subsection outlines the sequence in each state by classifying the LonTalk® individual lower-layer communication software status as follows:

① Stop status
② Initialization processing status
③ Initialization completion stop status
④ Normal operation status
⑤ Error stop status
⑥ Suspension status

The diagram below illustrates the state transition for each status.

The transceiver state transition includes a portion operating in non-synchronization with the Neuron® chip, and is therefore described for each transceiver in 6.X.5 after Section 6.6.



6.5.2 Stop status

Stop status signifies a state in which no lower-layer communication software operations are performed. This status is established immediately after Power On. An outline of the processing immediately after state transition and an outline of individual lower-layer communication interface services that the stop status receives, and related processing, are described below.

(1) Trigger and action
Waits for an individual lower-layer communication interface service.
(2) Status acquisition service (LowGetStatus)
Returns LOW_STS_HALT (stop status) as the status.
(3) Lower-layer communication software type acquisition service (LowGetDevID)
Returns the lower-layer communication software type.

Triggers for state transition are as follows:
(1) Transition trigger to initialize processing state
This transition is caused by the initialization service (LowStart, LowInit).

6.5.3 Initialize processing status

Initialize processing status signifies that the lower-layer communication software is initialized.
An outline of the processing immediately after state transition and an outline of individual lower-layer communication interface services that the initialize processing status receives, and related processing, are described below.

(1) Trigger and action
LowStart and LowInit are indicated by the Communication Middleware.
① Initializes the transceiver.
Transceiver initialization is performed in non-synchronization with the Neuron® chip.
The Neuron® chip is notified of a BUSY condition and initialization is effected.
When initialization is completed, READY is notified to the Neuron® chip.

② Obtains a unique MAC address in the subnet.
The Node ID of the Neuron® chip is converted into 8 bits, and the converted data with an MSB of "0" is notified as the MAC address.
In the warm start mode, acquisition starts with the retained MAC address used. In the cold start mode, the retained MAC address is discarded to obtain a new MAC address. No group ID operation is performed.

③ Obtains a house code.
Media that support house codes are not supported in Ver 1.0, 2.0, 2.1 or 2.11.

(2) Status acquisition service
In a cold start, returns LOW_STS_INIT as the status. In a warm start, returns LOW_STS_RST.

(3) Lower-layer communication software type acquisition service (LowGetDevID)
Returns the lower-layer communication software type.

Triggers for state transition are as follows:

(1) Transition trigger to initialization completion stop status
This transition is caused by initializing the transceiver, obtaining a MAC address, and obtaining a radio system identification code.

(2) Transition trigger to stop status
This transition is caused by a MAC address acquisition failure or other error.

6.5.4 "Communication Stop" status

The "communication stop" status is a state in which an operation start request from the communication middleware is being waited for after completion of the initialization of the lower-layer communication software. An overview of the processing immediately after the state transition is given below together with a list of the individual lower-layer communication interface services that can be accepted during the "communication stop" state with brief explanations of the processing to be performed in relation to the services.

(1) Trigger and the response behavior
Waits for an individual lower-layer communication interface service.

(2) Status acquisition service (LowGetStatus)
Returns LOW_STS_CSTOP ("communication stop" state) as the status.

(3) Physical address acquisition service (LowGetAddress)
Returns the MAC address.

(4) Profile data acquisition service (LowGetProData)
Returns the profile data.

(5) Lower-layer communication software type acquisition service (LowGetDevID)
Returns the type of lower-layer communication software.

State transition triggers:

(1) Trigger for a transition to the “normal operation” state
Operation start instruction service (LowRequestRun)
(2) Trigger for a transition to the “initialization processing in progress” state
Initialization processing services (LowStart, LowInit)
(3) Trigger for a transition to the “stop” state
Stop service (LowHalt)

6.5.5 “Normal Operation” status

The “normal operation” status is a state in which a message is being transmitted to or received from the transmission medium (i.e. a state in which the primary function of the lower-layer communication software is being performed). An overview of the processing immediately after the state transition is given below together with a list of the individual lower-layer communication interface services that can be accepted during the “normal operation” state with brief explanations of the processing to be performed in relation to the services.

(1) Trigger and the response behavior
Waits for an individual lower-layer communication interface service.
(2) Status acquisition service (LowGetStatus)
Returns LOW_STS_RUN (“operating” state) as the status.
(3) Physical address acquisition service (LowGetAddress)
Returns the MAC address.
(4) Profile data acquisition service (LowGetProData)
Returns the profile data.
(5) Message transmission service (LowSendData)
Converts the provided protocol difference absorption processing section message into a lower-layer communication software message and outputs it to the transmission medium.
<Overview of the Transmission Sequence>
* Starts outputting the signals after confirming that the transceiver is in the “READY” state.
* The processing specified in the LonTalk® Standard is performed.
The sequence of processing for exchanges between the MAC layer and transceiver varies depending on the transmission medium used.

* The transceiver confirms that outputting of the Neuron® chip's transmission request signal has been cancelled and completes the output. Outputting of the BUSY signal is cancelled.

(6) Message reception service (LowReceiveData)

Converts the lower-layer communication software message received from the transmission medium into a protocol difference absorption processing block message and outputs it to the Protocol Difference Absorption Processing Block.

<Overview of the Reception Sequence>

* After starting the reception, the transceiver starts outputting the collision detection signal and the BUSY signal to the Neuron® chip.
* Only the PPDU of the received message is transferred to the Neuron® chip. After completion of the reception on the transceiver, outputting of the collision detection and BUSY signals is cancelled.

  The processing specified in the LonTalk® Standard is performed.
* Converts the provided lower-layer communication software message into a protocol difference absorption processing block message and outputs it to the Protocol Difference Absorption Processing Block.

(7) Lower-layer communication software type acquisition service (LowGetDevID)

Returns the type of lower-layer communication software.

State transition triggers:

(1) Trigger for a transition to the "stop" state

Stop service (LowHalt)

(2) Trigger for a transition to the "suspension" state

Lower-layer communication section stop service (LowSuspend)

(3) Trigger for a transition to the "initialization processing in progress" state

Initialization processing services (LowStart, LowInit)

(4) Trigger for a transition to the "error stop" state

An error

6.5.6 "Error Stop" status

The "error stop" status is a state in which the operation of the software has been stopped as a result of an error. An overview of the processing to be performed immediately after the state transition is given below together with a list of the individual lower-layer communication interface services that can be accepted during the "error stop" state with brief explanations of the processing in relation to the services.

(1) Trigger and the response behavior

When an error is detected, a state transition will be made and the error processing

will be performed.

(2) Status acquisition service (LowGetStatus)
Returns LOW_STS_SUSPEND_ERROR (“error stop” state) as the status.

(3) Lower-layer communication software type acquisition service (LowGetDevID)
Returns the type of lower-layer communication software.

State transition triggers:

(1) Trigger for a transition to the “stop” state
Stop service (LowHalt)
Any message being received will be discarded and any new or outstanding message transmission request will be rejected and an error will be returned, before the transition to the “stop” state becomes effective.

(2) Trigger for a transition to the “initialization processing in progress” state
Initialization processing services (LowStart, LowInit)

(3) Trigger for a transition to the “normal operation” state
Removal of the cause of the error

6.5.7 “Suspension” status

The “suspension” status is a state in which the operation of the software has been temporarily stopped in response to an instruction from the communication middleware. An overview of the processing to be performed immediately after the state transition is given below together with a list of the individual lower-layer communication interface services that can be accepted during the “suspension” state with brief explanations of the processing in relation to the services.

(1) Trigger and the response behavior
A state transition is made in response to the lower-layer communication section stop service (LowSuspend).

(2) Status acquisition service (LowGetStatus)
Returns LOW_STS_SUSPEND (suspension state) as the status.

(3) Lower-layer communication software type acquisition service (LowGetDevID)
Returns the lower-layer communication software type.

Triggers for state transition are as follows:

(1) Transition trigger to normal operation status
This transition is caused by the operation restart service (LowWakeUp).

(2) Transition trigger to stop status

This transition is caused by the end service (LowHalt).

(3) Transition trigger to initialize processing status

This transition is caused by the initialization service (LowStart, LonInit).

6.5.8 (Neuron® Chip) Node ID setting sequence

Node ID in Section 6.5.8 denotes the Node ID in the Neuron® chip and corresponds to the MAC address specified in ECHONET. This Node ID is uniquely associated with the Node ID of ECHONET according to the conversion specifications described in Part 2, Section 7.4.5. In the subnet, one node always exists as the master and controls the (Neuron® chip) Node ID in the subnet. The Master (Neuron® chip) Node ID shall be 0x7E. (Neuron® chip) Node ID = 0x7F is specified as Node ID undefined status, and others are assigned uniquely to slaves in the subnet. Master/slave operations have the same relationship with portions higher than the Protocol Difference Absorption Processing layer. However, a distinction between the two is specified as required for communication processing after the transceiver.

Two methods for (Neuron® chip) Node ID setting are specified:

1. Setting by DIP-sw:
   - Set any unique optional value from 0x01 to 0x7D in the subnet using I/O ports.

2. Automatic setting by communication port
   1) The slave issues a service message when its own (Neuron® chip) Node ID is undefined.
   2) Upon receiving the service message, the master refers to its own domain table and sends free address data to the slave that issued the service pin message. Here, NEURON_ID is specified as the slave address (MAC address notification message).
   3) Upon receiving the address data, the slave rewrites its own domain table and sets its own (Neuron® chip) Node ID.

      If, in this instance, no notification is sent from the master within 10 seconds after process 1) or an illegal message format is encountered in 2), the subsequent operations are canceled.
   4) The slave sends a confirmation signal to the master with SUBNET_NODE specified (MAC address confirmation message).
   5) In response to the confirmation signal, the master transmits an ACK message. If no notification is sent from the slave within 10 seconds after process 2) or an illegal message format is encountered in 4), the subsequent operations are canceled.
   6) The slave receives ACK. The entire (Neuron® chip) Node ID acquisition sequence is now completed. If no notification is sent from the master within 10 seconds after process 4) or an illegal message format is encountered in 5), an abnormal end occurs.

- If NO_ID is received from a transceiver when ID acquisition is completed, it is concluded that the node has moved from one subnet to another. The current (Neuron® chip) Node ID is then invalidated to start a (Neuron® chip) Node ID acquisition process.

The range of addresses that can be registered with a DIP switch, etc. is differentiated from the range of automatically selectable addresses.

Specifications for message codes, confirmation signals, and ACK values are as follows:

Outgoing message	Message code	Address type	APDU (data)	
Service pin message	0x7F	BROADCAST	–	
MAC address notification message	0x01	NEURON_ID specified	0x01	MAC
MAC address confirmation message	0x02	SUBNET_NODE	0x02	MAC
ACK message	0x03	SUBNET_NODE	0x03	

6.6 RCR STD-16 Transceiver Specifications

The transceiver is specified for cases in which the specific low-power radio (hereafter referred to as ARIB STD-T67) to which ARIB Standard ARIB STD-T67 is applied is used as the transmission medium.

ARIB STD-T67: Radio equipment for telemetering and telecontrol in a specific low-power radio station.

6.6.1 System overview

In order to perform transmit/receive processing to satisfy the ARIB STD-T67, frame conversion is performed for the addition of special header data to the PPDU. In addition, protocol processing that cannot be controlled by LonTalk® as carrier sense, automatic communication channel switching, group ID registration, and internode communication check is performed in non-synchronization with LonTalk®, and the transceiver operation status (Status) is directly notified to Layer 3.

To perform such processing, the specific low-power radio transceiver consists of a Neuron® chip, RF module, RF microcomputer as an intermediary between the Neuron® chip and RF module, TxSW for transmission and RxSW for reception for switching messages with a header to be transmitted or received between the PPDU and RF module, auxiliary memory to store necessary control information, and identification codes as indicated in Fig. 6.5.

The RF microcomputer, TxSW, and RxSW described above need not always be microcomputers and switches; they may simply be the corresponding function blocks. ECHONET does not provide detailed configurations for these.

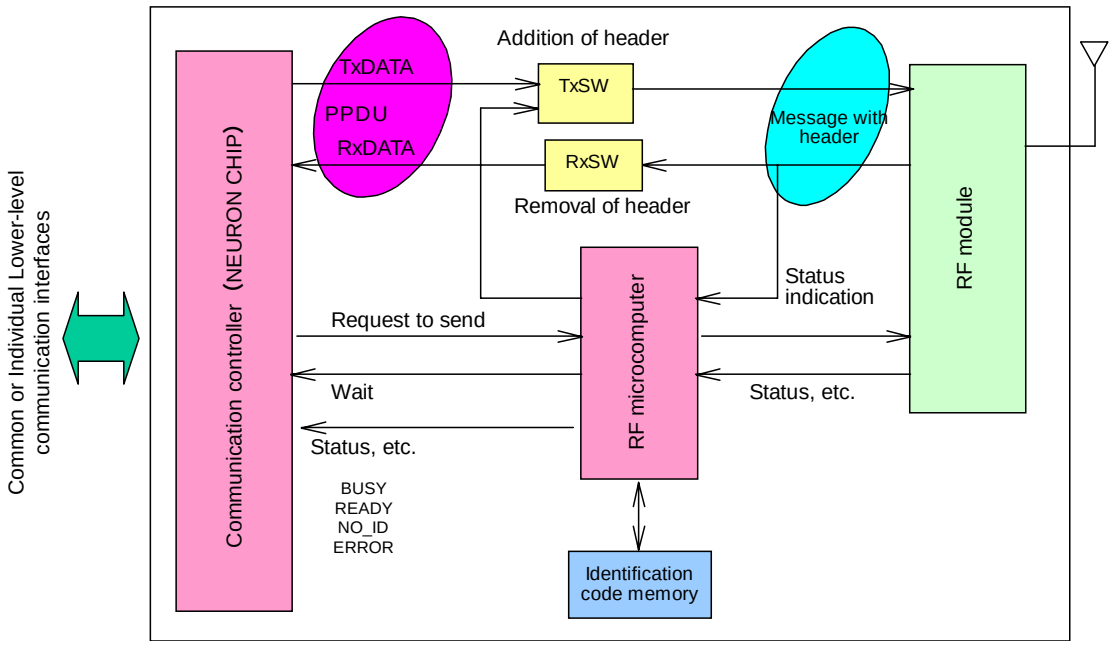


Fig. 6.5 Transceiver Configuration

6.6.2 Mechanical/physical specifications

- Connector shape: Antenna type and shape and connector shape are not specified. However, the requirements of ARIB STD-T67 shall be satisfied.
- Transmission media: 400 MHz band continuous communication channels, Channels 7 to 46, 429.2500 MHz to 429.7375 MHz.

  Channel groups such as those shown in Table 6.2 are configured. For channel use, the channel switching system described in Section 6.6.6 shall be operated.
- Topology: Free. A group ID shall be owned by each subnet.
- Others: Starting means for the group registration mode (mandatory) and the test signal transmission mode shall be provided (this is optional, but test signal receive processing is mandatory).

  A means that permits checking the progressive status of the group registration mode shall also be provided.

6.6.3 Electrical characteristics

- Electrical characteristics of media: ARIB STD-T67 shall be adhered to.
- Transmission rate: Differential Manchester 2400 bps
- Modulation system (transmission system): Binary FSK (F1D) NRZ
  Furthermore, the frequency shift shall be 2.1 kHz±0.4 kHz.
- Transmitting/receiving sensitivity (level): ARIB STD-T67 shall be adhered to.
  Furthermore, the code reference sensitivity shall be not higher than 1.4 μV.
- Channel pairs 0 and 1 (Channels 7, 26, 27, and 46) shall be used for group ID registration.
- Basically, the RF module shall always be waiting for reception, and the specific node (master) shall perform a transmit operation periodically as shown below. Accordingly, a stable, continuous supply of power is recommended.

Table 6.2 Channel Frequency Bands

Channel pair	ChA	ChB	
0	*8ch*	*28ch*	Pair group 1
1	*10ch*	*30ch*	
2	*12ch*	*32ch*	
3	*14ch*	*34ch*	
4	*16ch*	*36ch*	
5	*18ch*	*38ch*	
6	*20ch*	*40ch*	
7	*22ch*	*42ch*	
8	*24ch*	*44ch*	
0	*9ch*	*29ch*	Pair group 2
1	*11ch*	*31ch*	
2	*13ch*	*33ch*	
3	*15ch*	*35ch*	
4	*17ch*	*37ch*	
5	*19ch*	*39ch*	
6	*21ch*	*41ch*	
7	*23ch*	*43ch*	
8	*25ch*	*45ch*	
0	7ch	26ch	
1	27ch	46ch	

6.6.4 Logical specifications (Layer 1)

The Layer 1 data structure is as shown below; the footer is not specified. Data transmission is based on “MSB First”.

Header							LonTalk® PPDU
1st header			2nd header				
Bit synchroniza-tion	Frame synchroniza-tion 1	Identification code	Bit synchroniza-tion	Frame synchroniza-tion 2	Group ID	Command	

Fig. 6.6 RCR STD-16 Layer 1 Data Structure

- 1st header: Consists of two synchronization signals (preamble: Bit synchronization signal + Frame synchronization 1 signal) and the subsequent 48-bit identification code based on Radio Law regulations.
- 2nd header: Consists of two synchronization signals (preamble: Bit synchronization signal + Frame synchronization 2 signal), group ID, and a command. The 2nd header may be transmitted more then once. (1 to 16)

- Bit synchronization signal: Repetition of "1" with enough length for the system to perform carrier sense for one channel. (50 bits min., 120 bits max.)
- The frame synchronization signal shall be a 32-bit signal resulting from adding 0/1 to the 31-bit M series code.

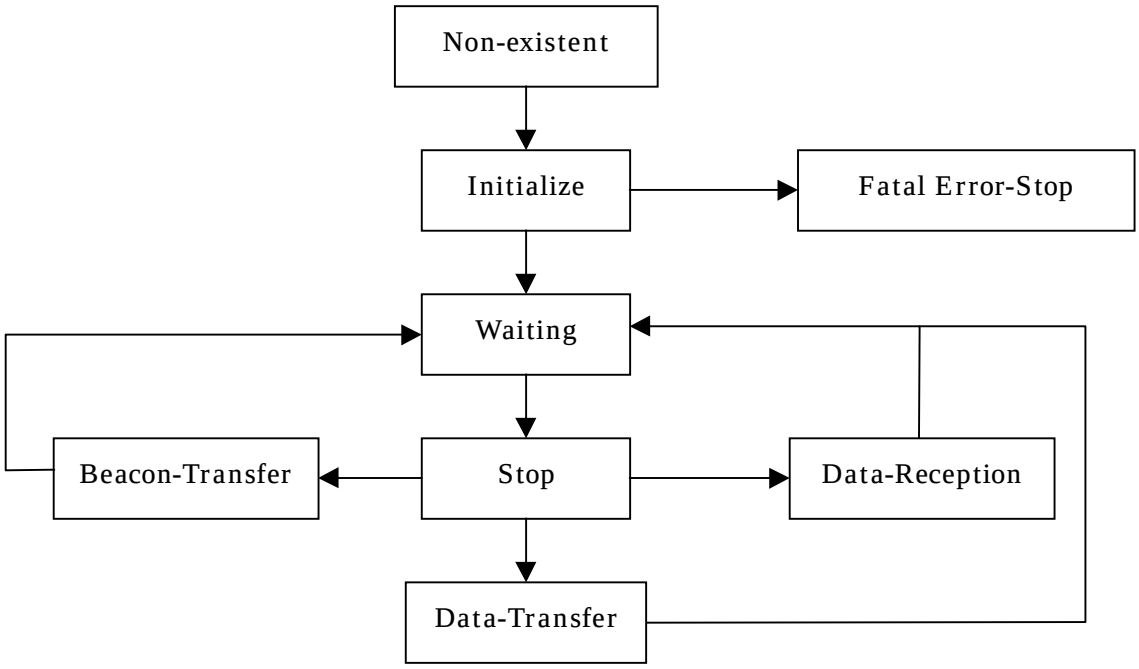
  Frame synchronization 1 signal: "00011011101010000100101100111110"

  Frame synchronization 2 signal: "00011011101010000100101100111111"
- The identification code is the Neuron_ID (48 bits) of the Neuron® chip of each node.
- Group ID: Uses an identification code having a node that becomes the master in the subnet. It is registered in all nodes in the subnet including the master (48 bits).
- Command: Discrete command to indicate the contents to be judged by the RF portion (32 bits). The following is specified for the high-order bit, b31, to the low-order bit, b0.

Group ID registration (GIDCMD = "0010000")	: Indicates a group ID transmission from the master.
Group ID registration response (SIDCMD = "0100000")	: Indicates a response concerning group ID registration.
Transmission test (TSTCMD = "0110000")	: Indicates an internode transmission test.
Reference signal (MCACMD = "1000000")	: Indicates that the outgoing data is a reference signal.
LonTalk (LONCMD = "1010000")	: Indicates that a LON message follows a command.
All other items shall be reserved for future use.	

  - b31: Assigned to the master/slave flag. "1" at the master.
  - b30 to b24: Command proper
  - b23 to b20: Residual count at 2nd header repetitive transmission (F to 0)
  - b19 to b16: Number of "1s" in b31 to b20.
  - b15 to b1: BCH code for b31 to b16 based on BCH (31, 16)
  - b0: Even parity
  - Neuron® Chip PPDU LonTalk® message.
  - If the command main body is other than LONCMD, the Layer 1 message consists of a header only.

6.6.5 Transceiver operation sequence



Non-existent:	Not operated.
Initialize:	Address setting, initial setting of RF circuit.
Waiting:	Wait for reception. Wait for timer, key input, or request to send.
Stopping:	Setting for a transmission stop request for LON and various requests.
Beacon-Transfer:	Periodic signal transmission to secure or check communication channels, including a communication test.
Data-Reception:	Data reception, including various processing of LONTALK® exchange and data.
Data-Transfer:	Data transmission (RF header and LONTALK® PPDU).
Fatal Error-Stopping:	Stop at address redundancy detection, stop upon error occurrence, etc.

6.6.6 Automatic channel switching system

- A specific node in the subnet is set as the master. The master node selects one pair from the pair groups. (Default: Channel pair 0.)
- Carrier sense is performed alternately by the two channels in the selected pair.
- After a lapse of one minute, if one of these channels is free, the reference signal is output. When either of them is free, the reference signal is output by the free channel. When both channels are free, the reference signal is output by chA.

If neither channel is free, the next channel pair is selected and carrier sense is restarted.

- For nodes other than the master node, carrier sense is performed alternately by the two channels of a certain channel pair. (Default: Channel pair 0.)
- When the reference signal or a signal from the same subnet is not detected within one minute, a channel change is made to the next pair.

- The beacon cycle shall not exceed 1 minute.
- The channel shall sequentially change from pair group 1 channel pairs 0 through 8 to pair group 2 channel pairs 0 through 8. This channel transition cycle shall be repeated.

6.6.7 Group ID registration

The ARIB STD-T67 nodes perform group ID registration as the initial setting to participate in the network. The registered group ID shall be nonvolatile, i.e., valid unless the network configuration is changed. The communication enable nodes shall be those belonging to the same subnet specified in ECHONET and have a unique group ID for each subnet. If the respective subnet is different, communication is performed through a router. An optional node (generally a router) in the subnet shall be specified as the master, and the master node identification code used as a group ID. The other nodes shall be slaves, with group ID registration performed according to the following procedure:

- To let a new node participate in the network as a slave, the slave starts the ID registration mode by the ID registration channel pair and waits for ID transmission from the master.
- After transmitting the reference signal in succession, the master moves to the ID registration channel pair and starts the ID registration mode, and then transmits its own group ID using the following format:

First header	Second header								
	Bit synchroni-zation	Frame synchroni-zation	Group ID (master identification code)	ID registration command					
				1	GIDCMD	b23~b20	b19~b16	BCH	P

- The slave stores the received group ID into its own memory and responds to the master according to the following format:
- If the transmission from the master cannot be confirmed within 12 seconds, the group registration mode is terminated and the operation is stopped.

First header	Second header								
	Bit synchroni-zation	Frame synchroni-zation	Group ID (master identification code)	ID registration response command (Slave)					
				1	SIDCMD	b23~b20	b19~b16	BCH	P

- After confirming the response from the slave, the master uses the following format to notify the slave of registration completion and exits the group ID registration mode. However, if the slave does not respond within 10 seconds after ID registration command transmission or if the slave's response contains a mismatched group ID, the master re-attempts the ID registration process. If the response cannot be confirmed after a maximum of four retries (i.e., five attempts including the first one), the master exits the group ID registration mode and returns to the normal operation state.

First header	Second header								
	Bit synchroni-zation	Frame synchroni-zation	Group ID (master identification code)	ID registration response command (Master)					
				1	SIDCMD	b23~b20	b19~b16	BCH	P

- After issuing a response, the slave confirms the ID registration response command from the master and then exits group ID registration mode. Subsequently, the slave notifies the Neuron® chip of group registration completion and issues a request for (Neuron® chip) Node ID setup. (Neuron® chip) Node ID setup is performed with the normal channel pair.

6.6.8 (Neuron® Chip) Node ID setting

The (Neuron® chip) Node ID corresponds to the Mac address specified in ECHONET. After group ID registration, (Neuron® chip) Node ID setting is performed. See Section 6.5.8.

6.6.9 Transmission system

- For a send request (transmission enable) or reference signal from the Neuron® chip, or upon occurrence of a test signal transmission request event, a collision detection signal and BUSY signal are output to the Neuron® chip.
- After completion of a procedure such as carrier sense, the RF circuit is switched over to transmission mode and the 1st and 2nd headers are output to the RF via a free channel. In cases other than LONCMD, the RF circuit is switched to reception mode, and the collision detection signal and BUSY signal are cleared to READY status.
- On the other hand, upon receiving a collision detection signal, the Neuron® chip completes the PPDU output or suspends it upon completion of preamble transmission (selectable by LON setting; in any case, the message output is not output to the RF before collision detection signal is cleared). After that, the Neuron® chip attempts retransmission in the randomized pause time.
- After completion of the RF output, the collision detection signal is cleared after confirming that the Neuron® chip is not transmitting the PPDU, and then the RF output (modulation input to the RF circuit) is switched over to the Neuron® chip side.
- After the end of the pause time, the Neuron® chip transmits the PPDU, and this message is sent to the RF.
- After clearing of the transmission request signal has been confirmed, RF output is terminated.
- The RF circuit is switched to reception mode, and the collision detection signal and BUSY signal are cleared to READY status.

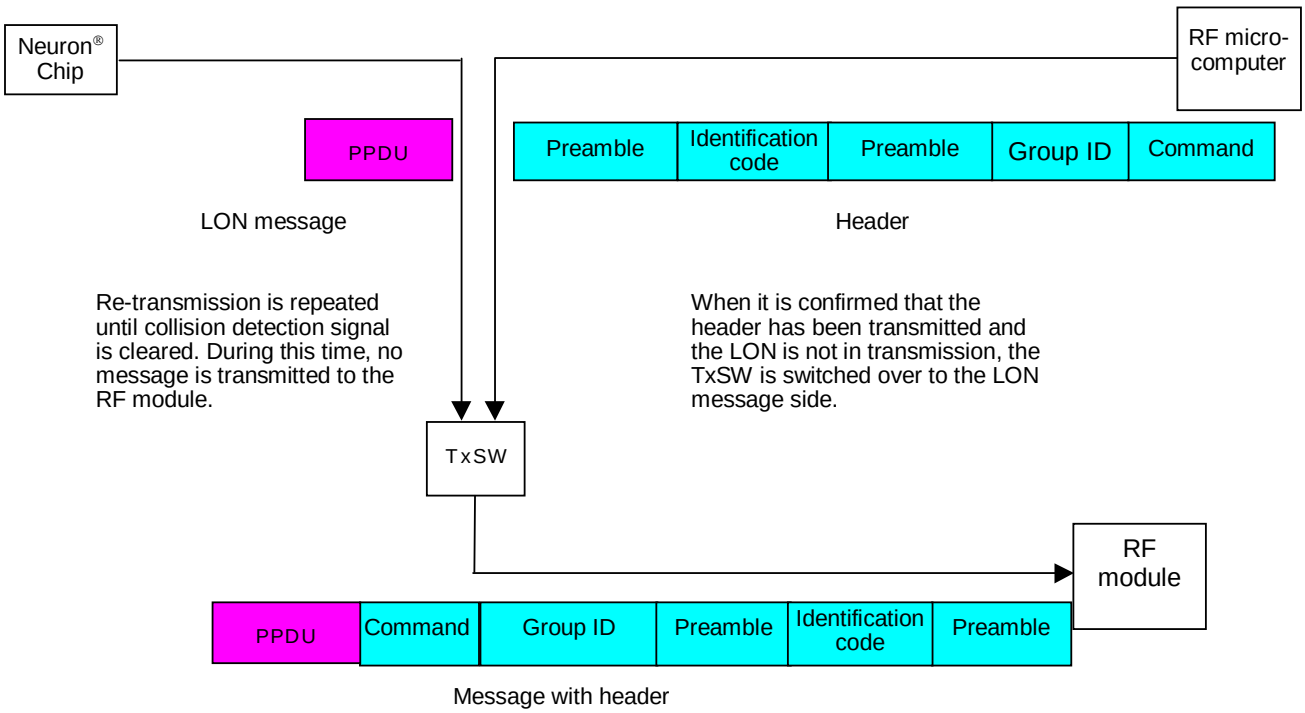


Fig. 6.7 ARIB STD-T67 Transmission During Neuron® Chip Use

- After confirming that a receiving carrier exists, the collision detection signal and the BUSY signal are output to the Neuron® chip, and reception is started.
- The group ID in the 2nd header is read. When it is found to match the self-group ID, the command data subsequent to it is received, and operations are performed according to the contents of the command data. If the group ID does not match (except for group ID registration), receive processing is suspended.
- When the command data is LONCMD, the RxSW is switched to cause the Neuron® chip to receive the PPDU.
- After confirming that a receiving carrier does not exist, reception is terminated.
- The collision detection signal and the BUSY signal are cleared to READY status.
- ACK/NAK/retransmission: No ACK/NAK request or response or data retransmission is performed by Layer 1. This depends on the setting or judgment of Layer 2 or higher.

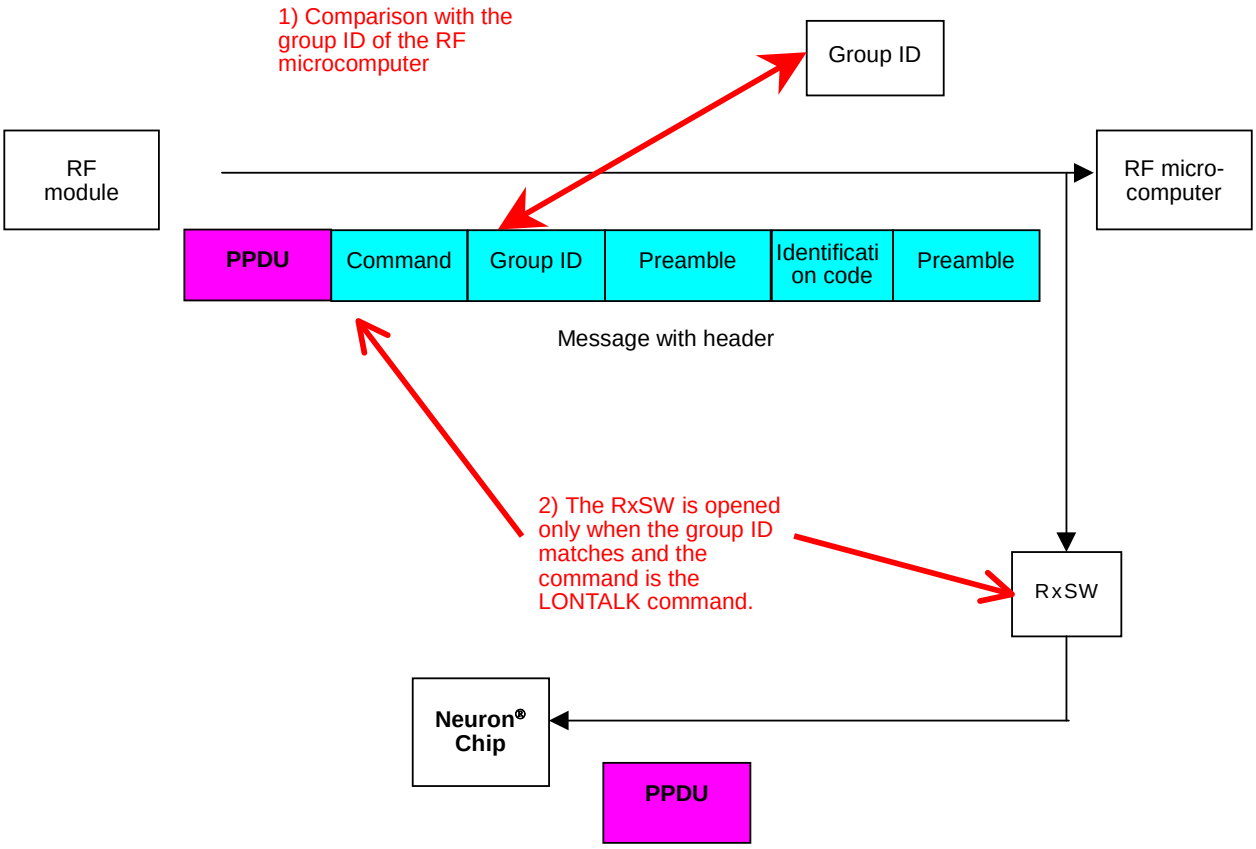


Fig. 6.8 Image of ARIB STD-T67 Reception during Neuron Chip Use

Appendix: Documents Cited

(1) Neuron Chip TMPN3150/3120 Data Book

(2) Neuron Chip Application Guide User's Manual

(3) Distributed Intelligent Control Network LON Works TM Overview
Toshiba Corporation Semiconductor Company, Domestic Sales Control Department
(Toshiba Bldg.) 1-1-1 Shibaura, Minato-ku, Tokyo 105-8001 (03) 3457-3405

(4) Neuron C Programmer's Guide

(5) Neuron C Reference Guide

(6) Technical Reference Materials for Lon Works® Custom Node Development

(7) Technical Reference Materials for Neuron 3150® Chip External Memory Interface

(8) Technical Reference Materials for LonTalk® Protocol

(9) Technical Reference Materials for Enhanced Media Access Control with LonTalk® Protocol
Echelon Corp. http://www.echelon.com ftp://lonworks.echelon.com

(10) ARIB Standard RCR STD-16 3.0
Association of Radio Industries and Businesses
(Tel: +81-3-5510-8590 Fax: +81-3-3592-1103)

Chapter 7 IP/Bluetooth Communication Protocol Specification

7.1 System Overview

This chapter specifies the system for accommodating the short-range wireless standard Bluetooth™ as an ECHONET transmission medium together with UDP/IP. Because most of the specifications in this Chapter are for defining the protocol as an application of UDP/IP, the advanced communication functions, low cost, and low power consumption of Bluetooth™ are preserved.

Particular emphasis was placed on 1), 2), 4) and 5) in defining the portion of the Specification that concerns the accommodation of Bluetooth functions, and on 3) and 5) in defining the portion that concerns the accommodation of UDP/IP.

1) It must be possible to obtain Bluetooth Qualification with an ordinary routine regardless of what equipment is installed, and the Qualification method must follow the specifications defined by the Bluetooth SIG (Special Interest Group).
2) The profile used for this system shall be based on the extent of the applicable profile defined in the Bluetooth Profile Specification. As the extent of the applicable profile defined in the Bluetooth Profile Specification will expand in the future, this Specification shall be updated accordingly. The same shall apply to any expansion other than expanding the extent of the applicable profile defined in the Bluetooth Core Specification (e.g., Scatternet, radio2 and others not considered initially).
3) No fundamental modification shall be made to the ECHONET Middleware layer and the layers above.
4) Efforts shall be made to minimize the implementation load concerning general equipment serving as slave equipment.
5) The ECHONET philosophy shall be adhered to for inter-subnet routing and connection with other networks.

The important point when using Bluetooth™ is selecting the right profile from the Bluetooth profiles contained in the Generic Access Profile that strictly define the implementation method according to use. For this Specification, it was decided to use the Personal Area Networking Profile (hereinafter referred to as "PAN Profile"), because it is most desirable from the viewpoint of ECHONET's network functions.
Figure 7.1 shows the layer structure. There are two cases defined by the PAN Profile, depending on whether or not the Network Bridge layer is present. The ECHONET layer is located above the UDP, IP and Bluetooth layers (Bluetooth Network Encapsulation Protocol, Logical Link Control and Adaptation Protocol, Link Manager Protocol and Baseband) and the Network Bridge. The ECHONET frames generated and processed in the ECHONET layer are encapsulated into UDP/IP and Bluetooth packets and transmitted between nodes.

The portion defined for ECHONET is treated as an application layer by Bluetooth™ and UDP/IP. The portion defined in this Chapter and the portion below it are seen as corresponding to Layers 1 and 2 by the ECHONET Communication Middleware.

For the portion above the individual lower-layer communication interface, there is no distinction between master and slave as defined by Bluetooth. However, for the portion below the individual lower-layer communication interface, this distinction shall be made for the purposes of this Specification. This version of the Specification will define only Internet Protocol Version 4 (hereinafter referred to as "IPv4") and will not define Internet Protocol Version 6 (hereinafter referred to as "IPv6").

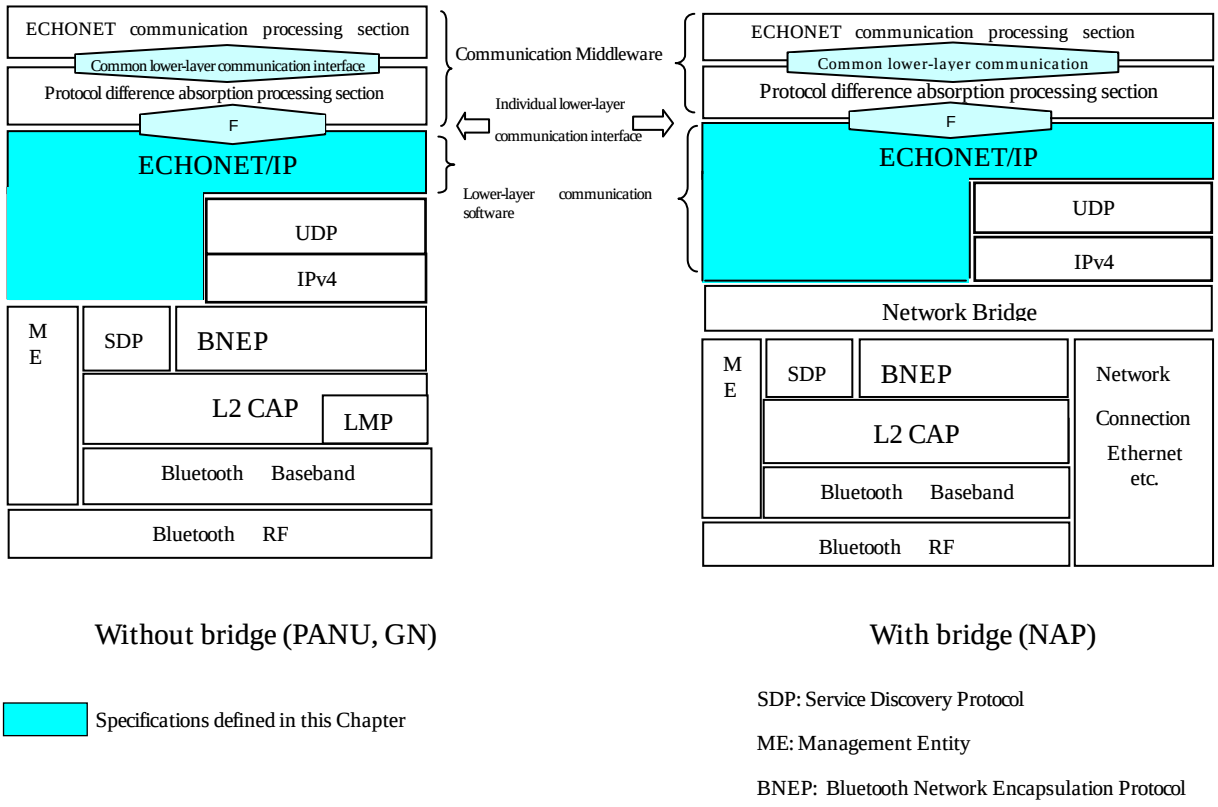


Fig. 7.1 Layer Structure

7.1.1 Communication model

(1) Topology

The topology used in a Piconet defined by Bluetooth is the star type topology. Communication between slave nodes, i.e., PAN User (defined by the Bluetooth PAN Profile; hereinafter referred to as "PANU") nodes, in a Piconet, is made via one or more Network Access Points (defined by the Bluetooth PAN Profile; hereinafter referred to as "NAPs") or one or more Group Ad-hoc Networks (defined by the Bluetooth PAN Profile; hereinafter referred to as "GNs") that satisfy the requirements presented in this Specification.

Requirements for the accommodation of Bluetooth™ are as follows:

1) The smallest subnet unit in ECHONET shall be Piconet. Each Piconet comprises a number of Bluetooth nodes that are defined in this Specification. Connection between subnets shall be made by means of one or more ECHONET routers, as with other media. Of course, a Piconet may contain a general Bluetooth node. Figures 7.2 and 7.3 show sample configurations.

Figure 7.4 provides an example of an unacceptable connection, in which Scatternet is used instead of ECHONET routers for connection.
Connection with a non-ECHONET network shall be made using an ECHONET gateway that uses NAP/GN as shown in Fig. 7.5.

The common accommodation requirements for all IP medium types are as follows:

1) A single IP subnet that contains nodes using different media (including Bluetooth and IP layers to be specified in the future in conjunction with ECHONET), with groups of nodes formed in such a way that no inter-group medium type difference is allowed but inter-group medium type differences are allowed, and connected by means of Layer 2 bridges defined by ANSI/IEEE Std. 802.1D etc. (including NAP bridges), shall be treated as a ECHONET subnet. That is, node sets using different ECHONET media but connected using Layer 2 bridges are defined as a single ECHONET subnet. This allows node sets using different media to be connected without using an ECHONET router and eliminates NetID changes during node transport within the subnet.
When a Layer 2 bridge is to be used, the ECHONET packet timeout requirement specified in this Specification must be satisfied.
In a Piconet, use of only one Layer 2 bridge (NAP) and incorporation of the bridge function in a PANU node are not allowed. Figure 7.6 shows an example of a bridge connection.

2) In a network comprising two or more IP subnets connected by means of an IP router or IP routers, the ECHONET subnets shall be contained in the respective IP subnet. That is, it is not allowed to use an IP network comprising two or more IP subnets connected by means of an IP router or IP routers as one ECHONET subnet, and connection between ECHONET subnets shall be made by means of an ECHONET router or ECHONET routers. ECHONET communication via an IP router is not allowed. Figure 7.7 shows an example of this type of connection.

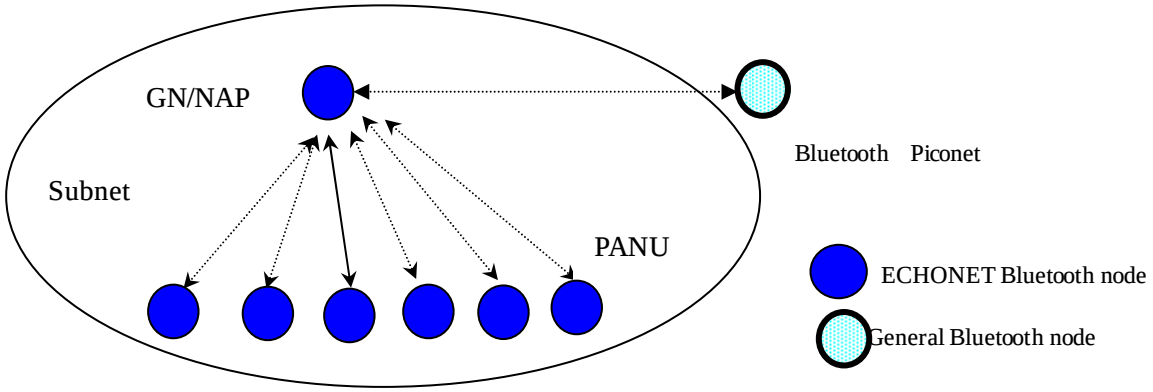


Fig. 7.2 Basic Form of Subnet

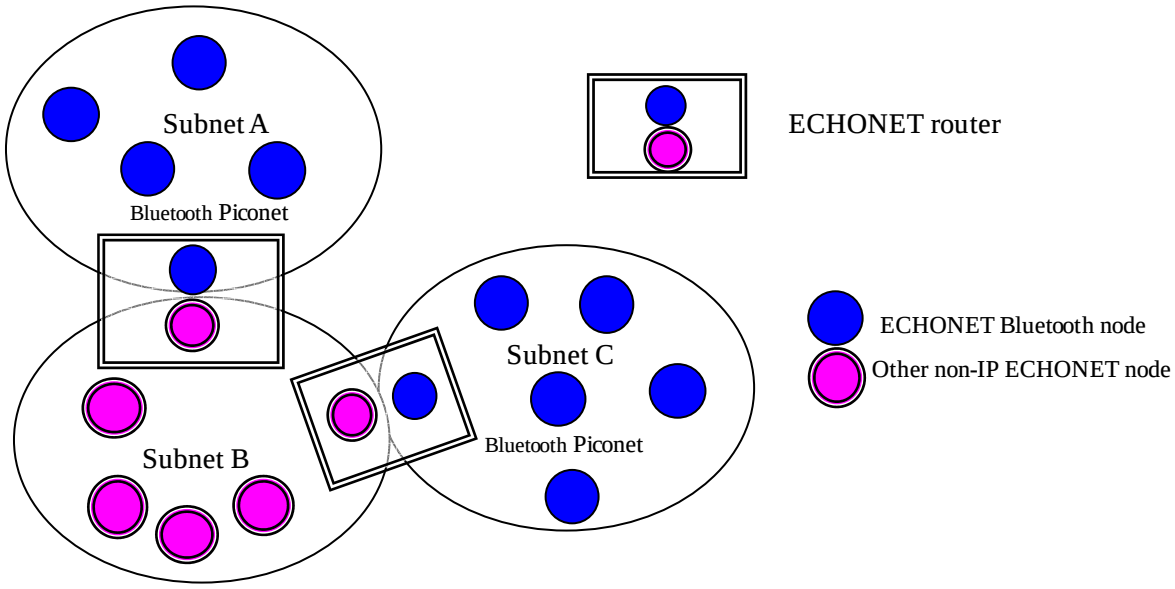


Fig. 7.3 Example of Subnet Connection Using ECHONET Routers

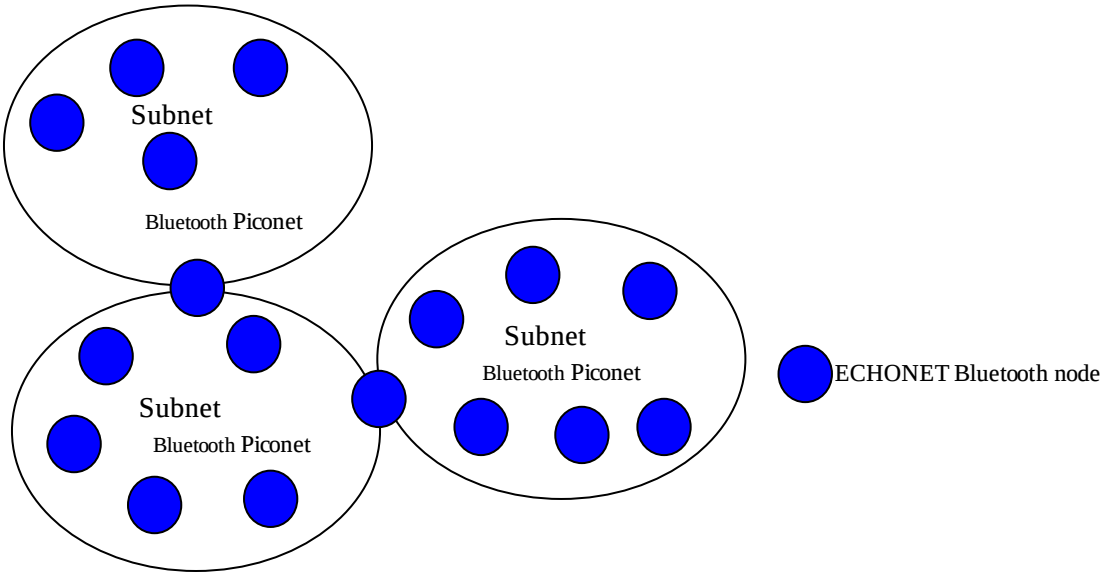


Fig. 7.4 Example of Unacceptable Subnet Connection (Scatternet)

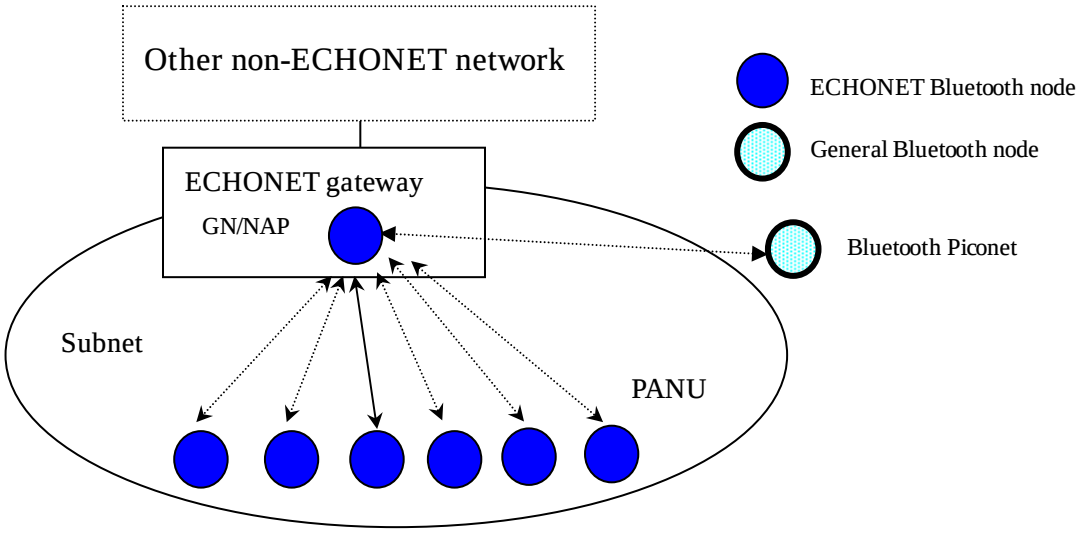


Fig. 7.5 Example of Connection Using an ECHONET Gateway

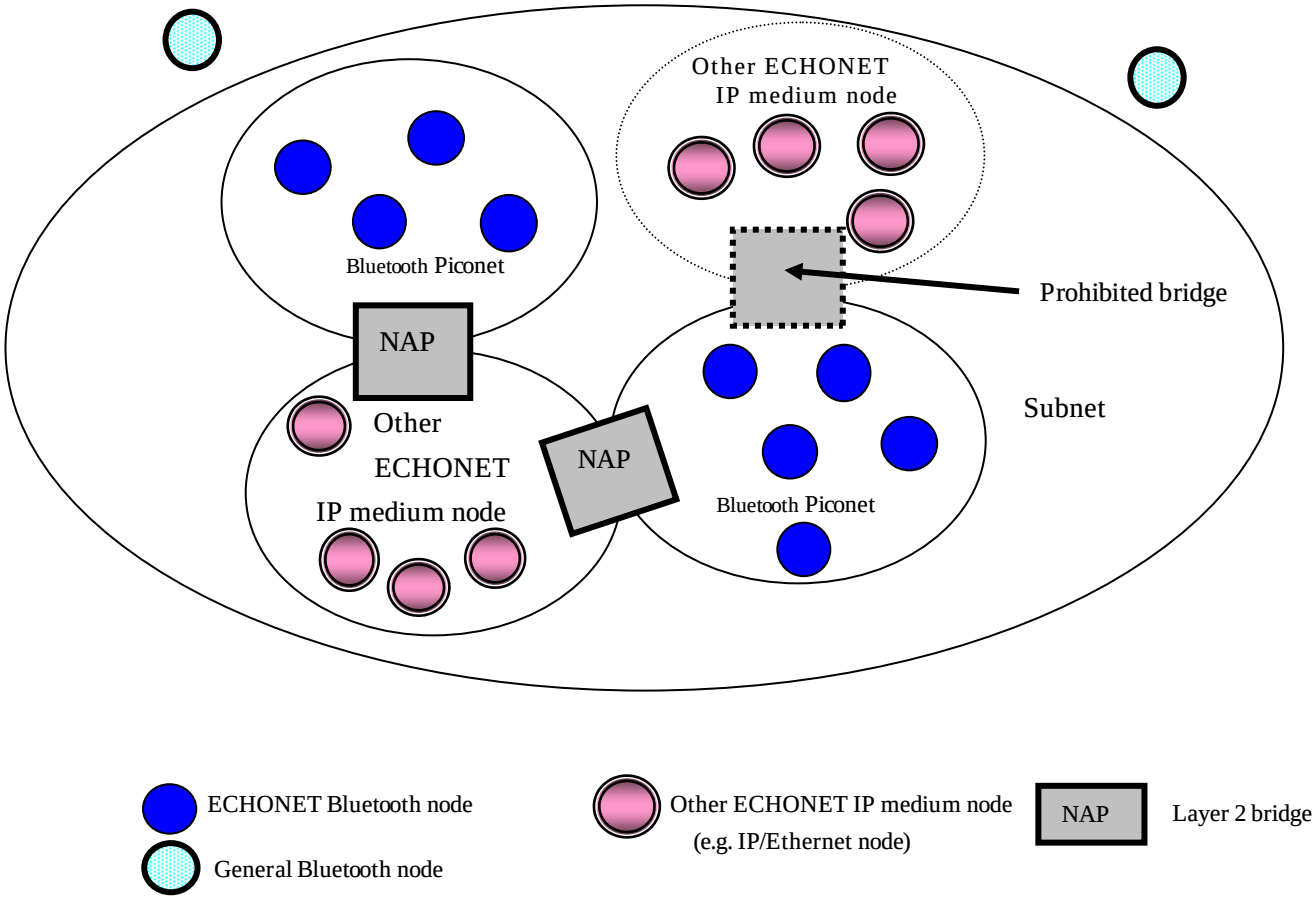


Fig. 7.6 Example of Subnet Using Layer 2 Bridges

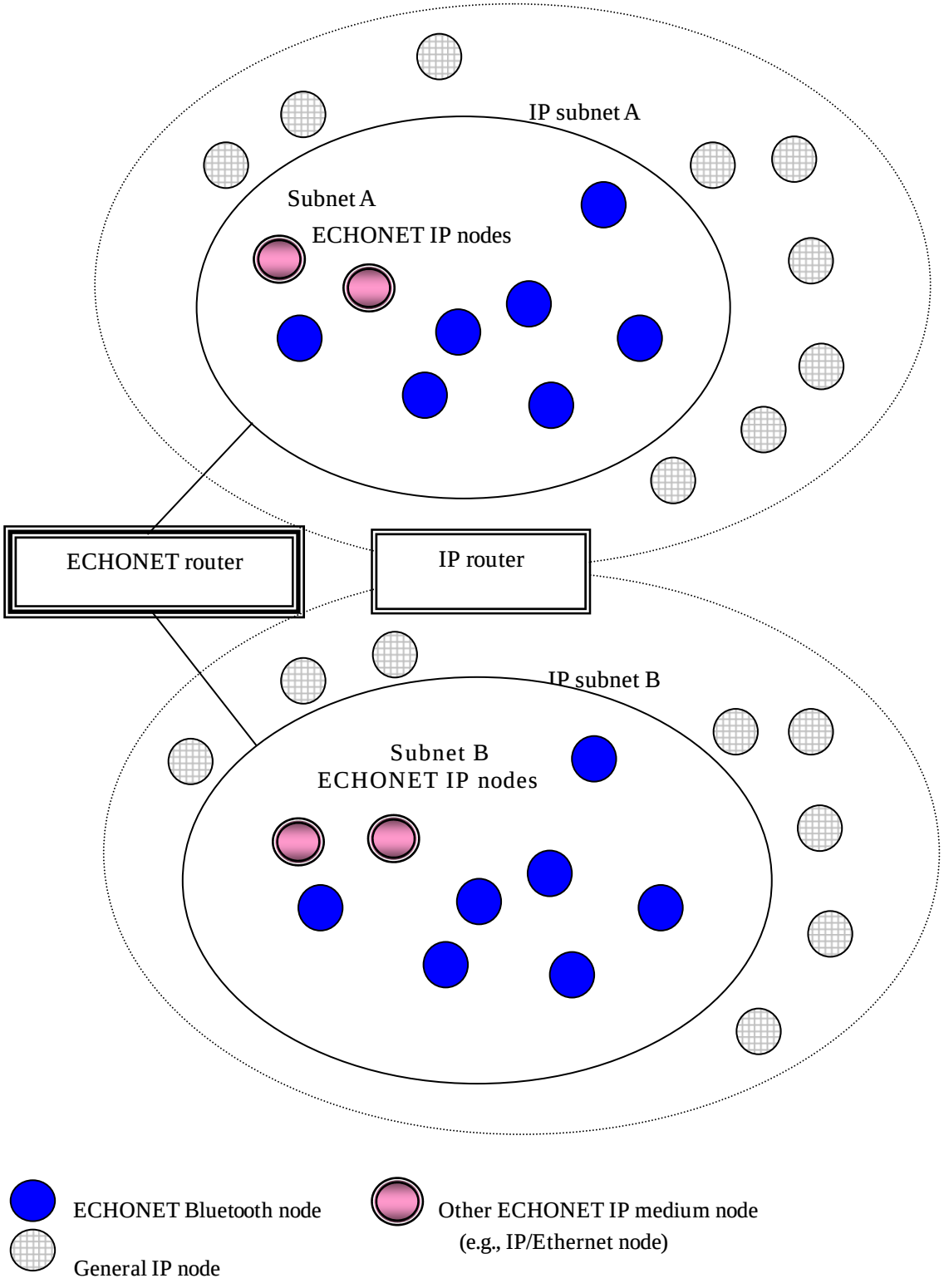


Fig. 7.7 Relationship Between IP and ECHONET Subnets

(2) Limit on Number of Terminals

With the use of the Park mode, a Piconet can be configured with Bluetooth nodes with a theoretical maximum of 256 ECHONET addresses. System designers, system operators, and so on must determine the topology and the maximum number of nodes in the Piconet (the recommended maximum number of nodes in a Piconet is 32) in full consideration of the necessary response time, duration of operation, etc. The maximum number of nodes in a Piconet that can perform communication simultaneously (i.e., the maximum number of active PANU nodes) is 7.

(3) Packet Length

ECHONET frames are accommodated in Bluetooth L2CAP packet payloads, with a BNEP header (max. 35 bytes), an IP header (IPv4, max. 24 bytes), and a UDP header (max. 8 bytes) attached to each frame. An L2CAP packet payload can easily accommodate an ECHONET frame with a packet length of 262 bytes (the maximum allowable packet length of an ECHONET frame), because the size of an L2CAP packet payload can be made as large as 65,535 bytes (default = 672). Therefore, there is no need to use split ECHONET frames. L2CAP packets are segmented into PDUs (Protocol Data Units) and exchanged using ACL packets defined by Bluetooth.

(4) Timeout Period

The length of time during which the response packet from the destination node for a PANU, GN, or NAP packet is to be transmitted varies between systems and states depending on such factors as the NAP/GN packet transfer processing time, the number of PANUs, whether or not the PARK mode is used, and whether or not the link is occupied by other applications. It also varies depending on whether or not a bridge is used, on the performance of the bridge, the processing speeds of individual nodes in the subnet, including the bridge, and the total number of nodes. This version of the Specification will specify standard fixed timeout period values taking into account these conditions and interconnectivity considerations. Methods for dynamically determining timeout periods and related matters will be specified in subsequent versions of this standard, as necessary.

7.1.2 Applicable standards

The requirements specified in the following Bluetooth Specifications shall be satisfied:

1) Bluetooth Specification Version 1.1 (Core Specification)
2) Bluetooth Specification Version 1.1 (Profile Specification)
3) Bluetooth Specification (Personal Area Networking Profile Version 1.0)
4) Bluetooth Specification (Bluetooth Network Encapsulation Protocol Version 1.0)

With regard to use in Japan, the requirements specified in the ARIB STD.-T66 “Second Generation Low-Power Data Communication Systems / Wireless LAN Systems Standard Specification” shall be satisfied, although these requirements are also included in the Bluetooth Specification Version 1.1 (Core Specification) mentioned above.

UDP/IP-related requirements are specified in Section 7.6.

7.1.3 Scope of this Specification

This Specification defines the interface specifications for communication between the ECHONET Communication Middleware and the Bluetooth and UDP/IP layers based on the ECHONET layer structure shown in Fig. 7.1. For detailed mechanical, physical, electrical, and logical specifications for the Bluetooth and UDP/IP layers, please refer to the relevant specification documentation.

Bluetooth communication functions are provided by the Bluetooth layer. The ECHONET/IP layers perform processing, referencing the status of the Bluetooth layer as necessary, and exchange control commands with the Bluetooth layer as necessary, but no interfaces are stipulated in this Specification.

As shown in Fig. 7.8, this Specification can be applied to Case1 and Case 2 when there is a coexisting application other than an ECHONET application using the PAN Profile, and to Case 3 when there is a coexisting profile other than the PAN Profile. However, this Specification does not specify a method for achieving coexistence with another application on the Bluetooth layer.

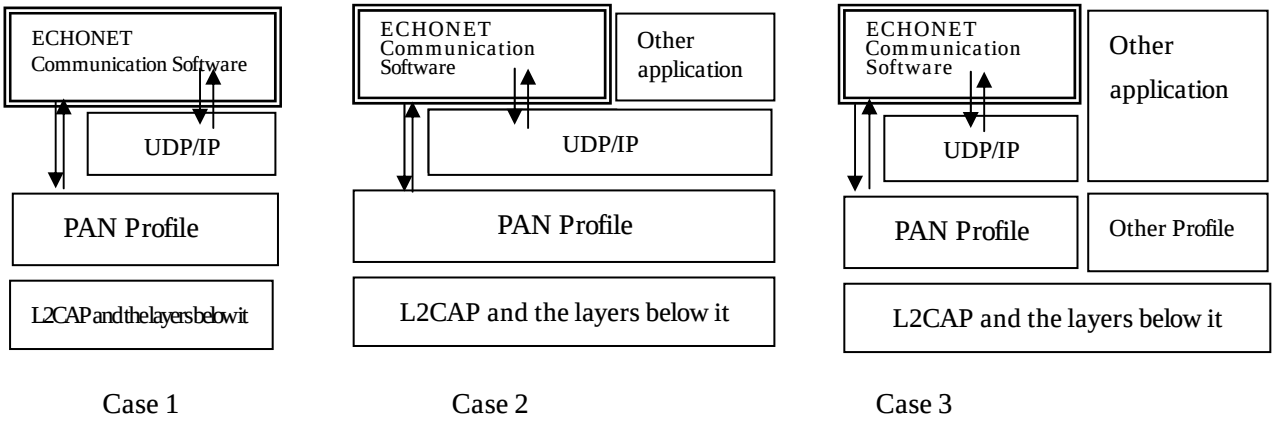


Fig. 7.8 Examples of ECHONET Communication Software Implementation

This Specification defines the requirements for PANUs, NAPs, and GNs as defined by the PAN Profile. For Layer 2 bridges, refer to the relevant sections of the PAN Profile Specification containing requirements relating to NAP and other standards such as the ANSI/IEEE Std. 802.1D.

7.2 Mechanical and Physical Characteristics

The mechanical and physical characteristics requirements of this Specification are drawn from ARIB Std.-T66 "Standard Specifications for Second Generation Low-Power Data Communication Systems / Wireless LAN Systems."

7.3 Electrical Requirements

The electrical requirements of this Specification are drawn from ARIB Std.-T66 "Standard Specifications for Second Generation Low-Power Data Communication Systems / Wireless LAN Systems" and the Bluetooth Specification Version 1.1 (Core Specification).

7.3.1 Transmission system and transmission signals

(1) Radio Wave Type

F1D

F: Modulation method for the main carrier = frequency modulation

1: Type of the signal for modulating the main carrier

= Digital signal (single channel) without sub carrier

D: Types of data transmitted = data transmission, telemetry, remote directions

(2) Output Power

Table 7.1 Radio Wave Output Power

Power class	Maximum output power (Pmax)	Minimum output power (Pmin)	Power control
1	100 mW (+20dBm)	1 mW (0dBm)	(Less than +4 dBm) to (+20 dBm) control mandatory Optional: Popt -Pmax
2	2.5 mW (+4dBm)	0.25 mW (– 6dBm)	Optional: Popt - Pmax
3	1 mW (0dBm)	–	Optional: Popt - Pmax

For radio wave output power, there are 3 power classes, namely, Class 1 (100 mW), Class 2 (2.5 mW) and Class 3 (1 mW). For Class 1 equipment, the power control function is mandatory and it must be possible to control the radio wave output to +4dBm or less. For the lower limit value (Popt) in the optional power control, use of a value less than –30dBm is recommended. The power control notch width shall be between 2dB (min.) and 8dB (max.).

(3) Communication Method (spread spectrum modulation)

requency hopping spread spectrum type simplex communication (See Fig. 7.9)

Hopping speed: 1600 hops/sec. (1 time slot = 625 μsec.)

79-channel hopping (1 MHz intervals)

(4) Modulation Method

GFSK (Gaussian Frequency Shift Keying)
BT (normalization bandwidth of Gaussian base band filter) = 0.5
Modulation index = 0.28 to 0.35

(5) Modulation Speed

1 M symbols/sec.

(6) Receiver Sensitivity

– 70 dBm (BER: 0.1% or less)

The receiver sensitivity is specified using a bit error rate (BER). It is necessary to achieve a receiver sensitivity of – 70 dBm with a BER of 0.1% or less.

(7) Duplex Communication Method

TDD (Time Division Duplex)

The length of one slot shall be 625 microseconds. Two-way communication shall be performed using the TDD (Time Division Duplex) method, in which transmission and reception are performed simultaneously. The number of slots that can be used for one packet is 1, 3, or 5, with the data transfer rate varying depending on the number of slots used.

(8) Connection Method

Asynchronous Connectionless (ACL) link

Maximum asymmetric rate (DH5/DH1 packet)	723/57.6 kbps
Maximum symmetric rate (DH5 packet)	433.9 kbps

ECHONET uses an ACL link to perform data communication (asynchronous packet exchange). The communication speed varies depending on the type of packets used. In asymmetric communication (communication mode in which uplink and downlink speeds are different), data can be transferred at a maximum rate of 723 kbps for the downlink and 57.6 kbps for the uplink using DH5 and DH1 packets. In symmetric communication, data can be transferred in both ways at a maximum rate of 433.9 kbps using DH5 packets.
The data transferred are error-corrected using FEC (Forward Error Correction), ARQ (Automatic Retransmission Query), etc., and are protected.

7.3.2 Frequency

(1) Operating frequency band

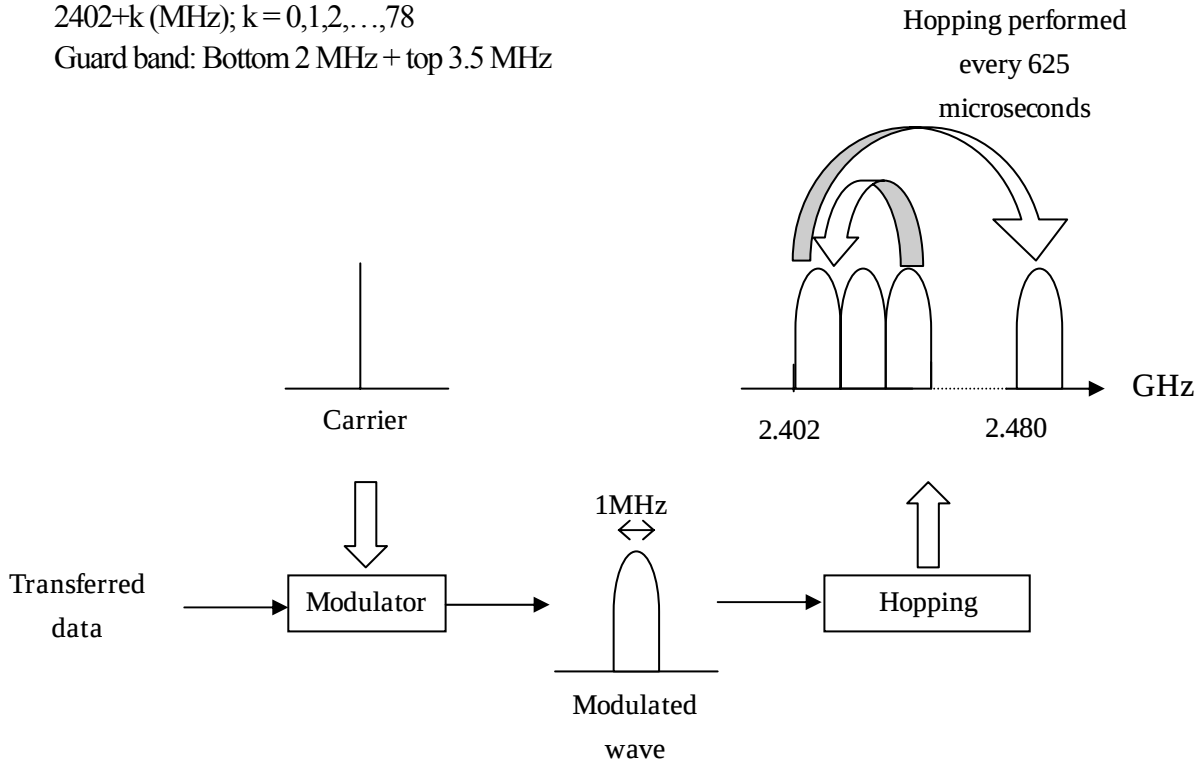
2.4000 - 2.4835GHz

Of the 2.4 GHz ISM (Industrial Scientific Medical) Bands that can be used without a license, the 2.4000–2.4835 GHz band shall be used (in which the equipment can be used as a "radio station" for a second-generation low-power data communication system). As this bandwidth is also used by radio stations other than Bluetooth systems, the occurrence of radio interference harmful to such a station would require the operator of the responsible equipment to stop transmission immediately and take corrective measures.

(2) RF Channel

2402+k (MHz); k = 0,1,2,…,78

Guard band: Bottom 2 MHz + top 3.5 MHz



Frequency hopping spread spectrum method: A technique that changes the carrier frequency of the main signal that has been modulated by the information signal in a random and discrete manner within the given frequency bandwidth according to the spread signal to achieve sweeping.

Fig. 7.9 Frequency Hopping Spread Spectrum Method

7.4 Overview of the Logical Specifications

When ECHONET operates on an Internet Protocol (IP) network, one ECHONET subnet is mapped onto one IP subnet. ECHONET frames are encapsulated into IP packets and transferred over the IP network. Of the two types of IP networks (IPv4 and IPv6), IPv4 shall be used under this Specification. The mapping specifications are explained in Section 7.7.

Each ECHONET node shall have one IP address. In the case of IPv4, the IP address shall be a global unique IP address or a private IP address. The method for obtaining this IP address is not specified in this Specification, but no ECHONET node can operate as an ECHONET node until it obtains an IP address.

A connectionless type UDP (User Datagram Protocol) is to be used for the IP network transfer protocol for transferring ECHONET frames. This is because communication in ECHONET is also connectionless. *** shall be used as a permanently fixed number for the UDP port number for transferring ECHONET frames or related control packets (the application for the use of *** is currently being processed). One ECHONET frame shall be encapsulated into one UDP packet.

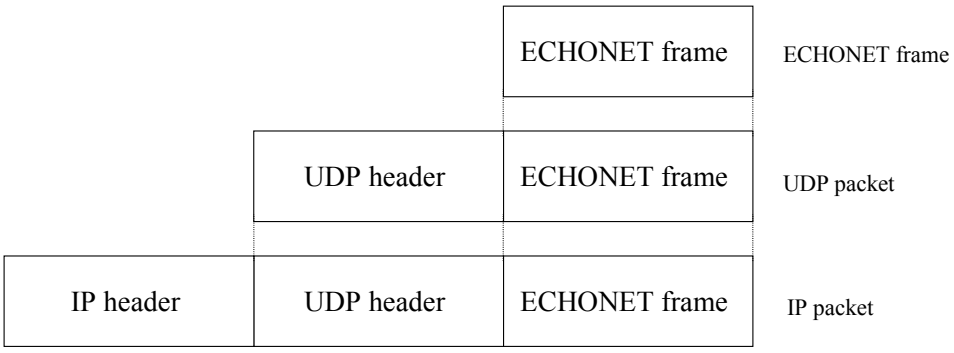


Fig. 7.10 ECHONET Frame Encapsulation

The destination address (source address) of an ECHONET frame shall be an ECHONET address. The destination address (source address) of an IP header shall be an IP address. All ECHONET nodes operating on an IP subnet (hereinafter referred to as IP/ECHONET nodes) shall subscribe to an IP multicast address assigned for ECHONET. The IP multicast address for ECHONET shall be *** (in the case of IPv4; the application for the use of *** is currently being processed).

ECHONET frame broadcasts and group broadcasts are mapped onto IP multicast packets addressed to this IP multicast address and transferred.

When Bluetooth is to be used as the ECHONET transmission medium, the Bluetooth standard PAN Profile, specified as the method for performing IP packet transfers over Bluetooth, is to be used.

For transfers of IP packets to Bluetooth, a protocol called BNEP for transferring Layer 3 packets is specified. This protocol is used to transfer Ethernet frames over Bluetooth. Figure 7.11 shows the protocol stack.

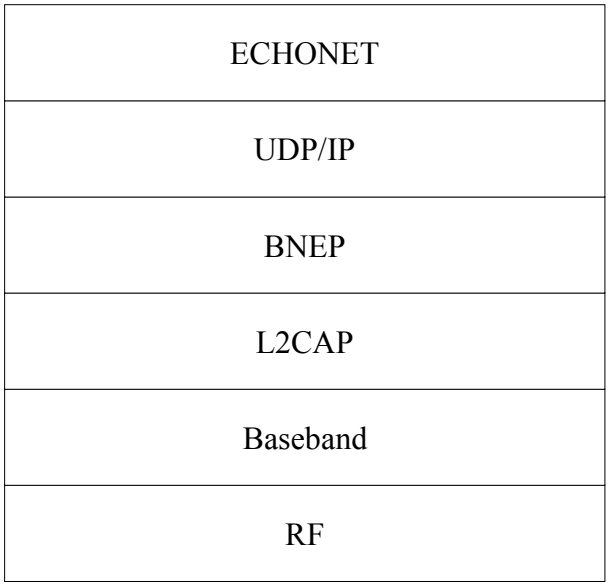


Fig. 7.11 Protocol Stack

IP packets to be transferred over Bluetooth are encapsulated into BNEP frames, which in turn are stored in L2CAP packets, segmented into Bluetooth frames and stored, and transferred over Bluetooth. Figure 7.12 shows the structure of packets transferred over Bluetooth.

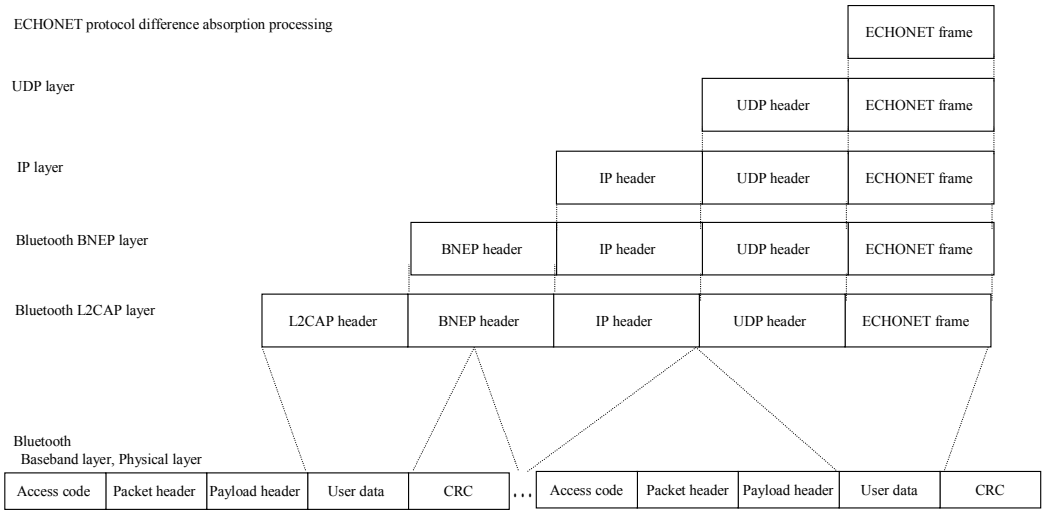


Fig. 7.12 Packet Structure

There are three methods that can be used for ECHONET address initialization (the process of determining the ECHONET MAC address to be used). These are 1) a method that uses MAC address servers; 2) a method called the Distributed Determination Method that does not use a MAC address server; and 3) a method in which the ECHONET MAC address is set manually.

To provide for cases in which a MAC address server must be used to set the ECHONET MAC address, three address setting modes have been defined: 1) an address setting mode called "Server Required Mode" (SR-MODE); 2) an "Automatic Mode" (A-MODE) that activates the Distributed Determination Method-based ECHONET MAC address determination mechanism when there is no MAC address server; and 3) a "Manual Setting Mode" (M-MODE) that allows manual setting of ECHONET MAC addresses (for details, see Table 7.23).

In the method that uses MAC address servers, one MAC address server is used in each subnet and all ECHONET MAC addresses in a subnet are managed by the MAC address server for that subnet. When there is a MAC address server in a subnet, the ECHONET nodes obtain ECHONET MAC addresses from the MAC address server at boot time. A sample initialization sequence for a subnet with a MAC address server follows.

The ECHONET node multicasts an "ECHONET address initialization packet" in the subnet to determine its ECHONET MAC address at boot time. When there is a MAC address server in the subnet, the MAC address server sends to the ECHONET node a MAC address server initialization response packet containing the ECHONET MAC address to be used by the node. In addition, address information is collected from all nodes in the subnet (by receiving ECHONET address initialization responses). After this, the ECHONET communication processing section formally establishes the ECHONET address.

The term "ECHONET MAC address" as used here means the address used to realize communication over an ECHONET transmission medium (IP/Bluetooth in this Chapter) as defined in the existing ECHONET Specifications, and must be distinguished from the Bluetooth address. The Bluetooth address shall be referred to as the "hardware address" in this chapter.

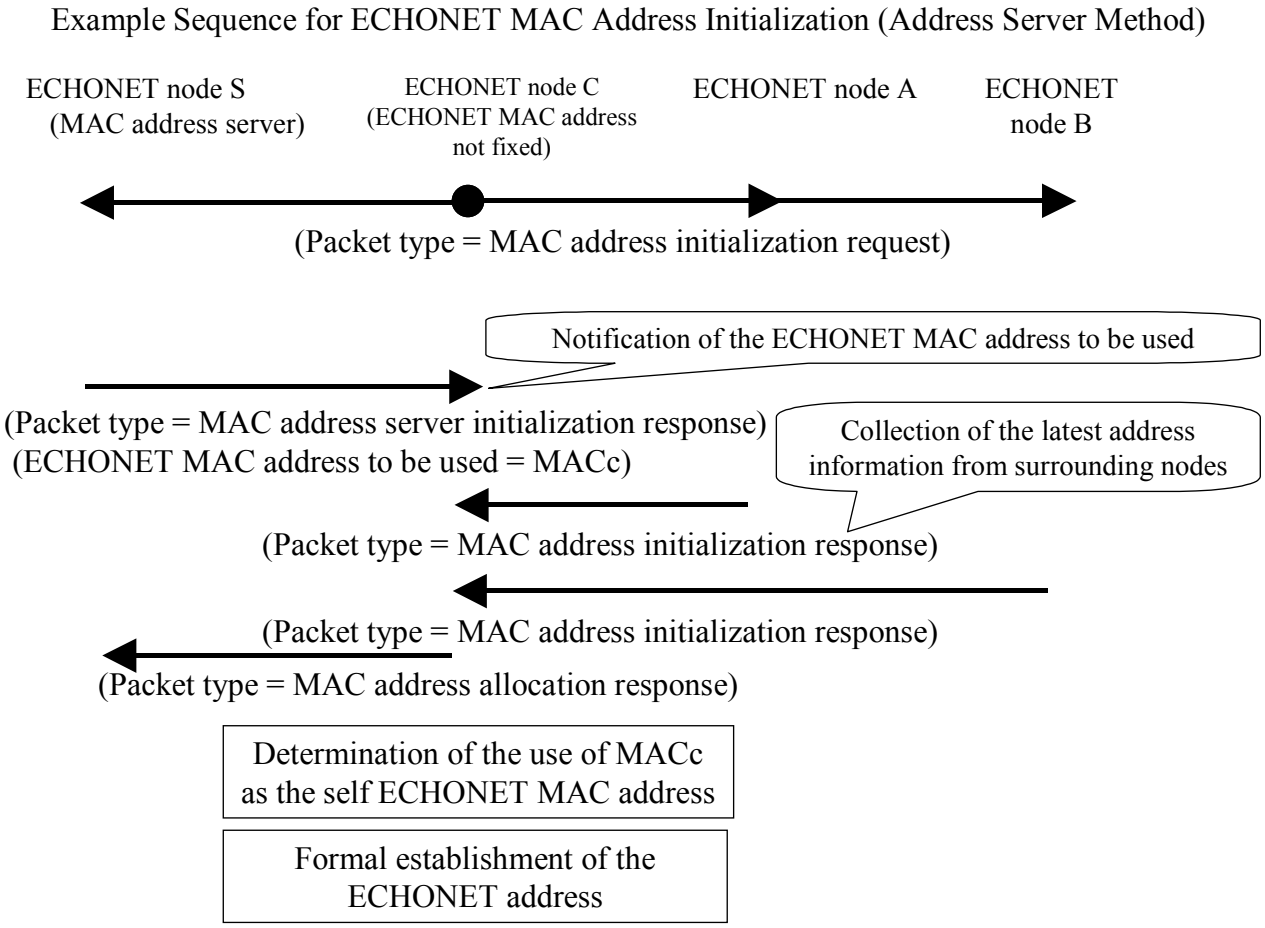


Fig. 7.13 Sample Sequence for ECHONET MAC Address Initialization (Address Server Method)

The Distributed Determination Method allows the ECHONET MAC address to be determined in an autonomous and distributed manner without using a MAC address server. In this method, the subnets do not have a MAC address server, and each node determines the ECHONET MAC address in an autonomous and distributed manner. This method is also used when a MAC address server has failed for some reason. In such a case, automatic switching is performed to enable the use of this method. A sample initialization sequence for a subnet without a MAC address server is shown in Fig. 14.

As with the MAC Address Server Method, the ECHONET node multicasts a "MAC address initialization request packet" in the subnet at boot time. Because there is no MAC address server in this case, all other ECHONET nodes in the subnet send their latest address information to the node. The node checks this information and selects an ECHONET MAC address that is not being used in the subnet for use as its own ECHONET MAC address. The ECHONET communication processing section then formally establishes the ECHONET address.

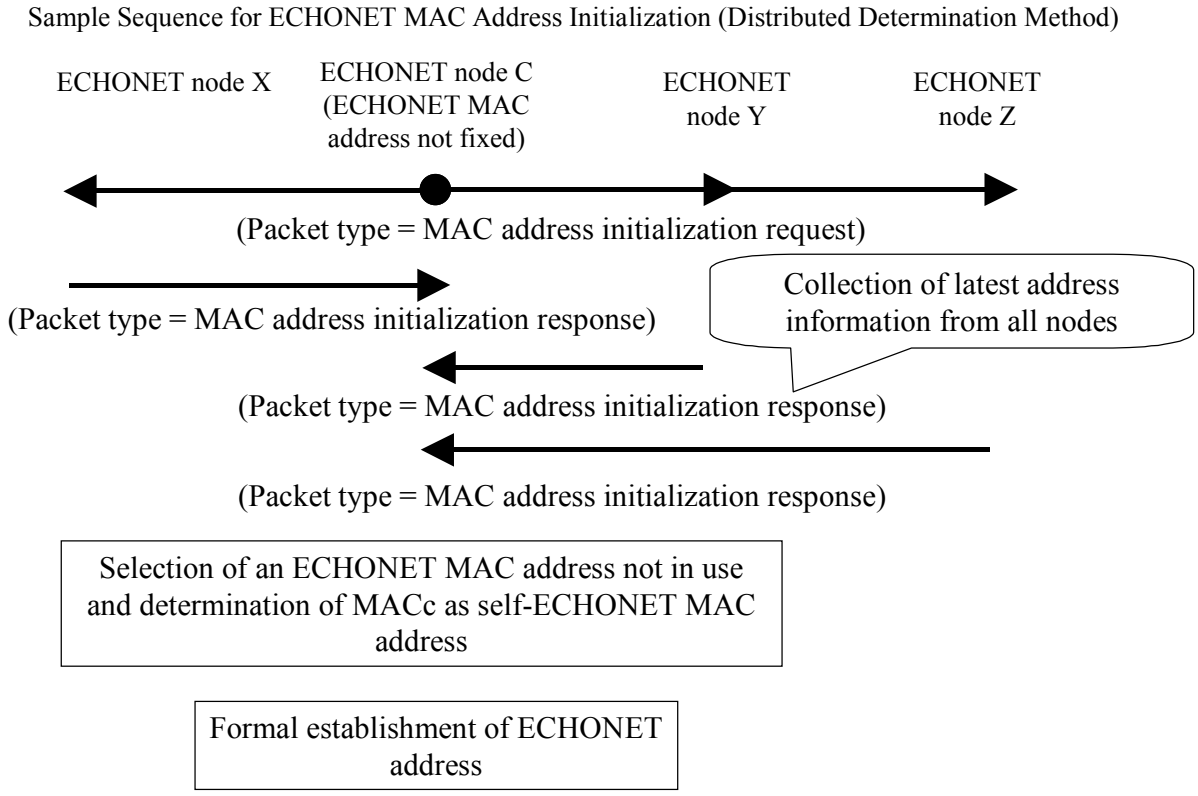


Fig. 7.14 Example Sequence for ECHONET MAC Address Initialization (Distributed Determination Method)

Upon completion of this address initialization process, the ECHONET node has an established ECHONET address and is ready to perform ECHONET communication. ECHONET communication is performed using UDP packets.

When an ECHONET node knows the ECHONET MAC address of the node with which it wishes to communicate but does not know the node's IP address, it can use "address resolution from MAC address into IP address." For example, consider the case shown in Fig. 7.15 where ECHONET Node C seeks to send ECHONET packets to ECHONET Node A (ECHONET MAC address = MACa) but does not know the IP address of ECHONET Node A. ECHONET Node C first multicasts a MAC/IP address resolution request in the subnet (destination IP address = IP multicast address for ECHONET). The MAC/IP address resolution request contains the ECHONET MAC address to be resolved (MACa). Upon receipt of the request, ECHONET Node A notifies ECHONET Node C of its IP address and hardware address (Bluetooth address) using a MAC/IP address resolution response. ECHONET Node C stores the relationship between the addresses in the internal address table.

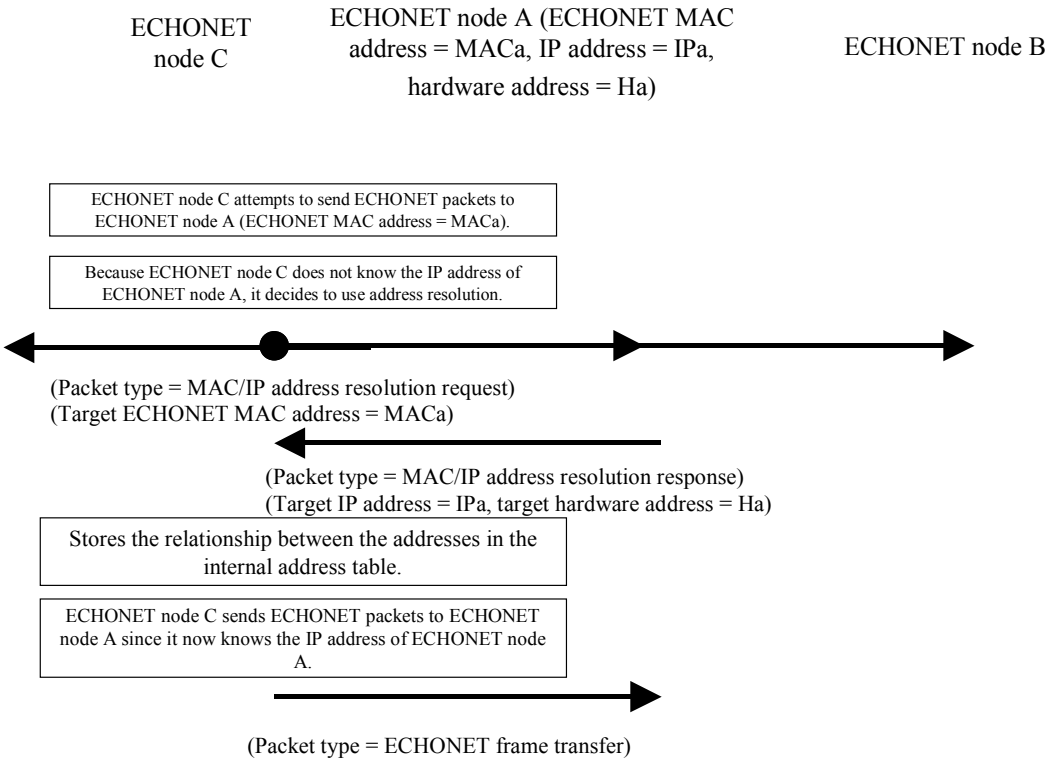


Fig. 7.15 MAC/IP Address Resolution

7.5 Logical Specifications (Bluetooth® Layer and Layers Below)

7.5.1 Bluetooth®

Bluetooth® has a Synchronous Connection Oriented (SCO) link, an Asynchronous Connectionless (ACL) link, and two kinds of communication links. The SCO link is a line-switching communication link that is used for (mainly voice) applications requiring real-time communication. The ACL link is a packet communication link that is used mainly for best-effort-type data communication applications. The IP/Bluetooth communication protocol specifications-based ECHONET performs communication using the ACL link.

Bluetooth has the following packet structure:

Access code	Packet header	Payload

Fig. 7.16 Bluetooth Packet Structure

The access code is used for DC offset compensation, Piconet identification and synchronization.

The packet header is the header necessary to control the baseband layer communication link.

The packet header has a 3-bit identifier called AM_ADDR, which is used by Bluetooth to identify the nodes in the Bluetooth Piconet.

The payload comprises a payload header, a payload body, and CRC. For ACL packets, the payload header has the functions necessary to control data in layers above the baseband layer, such as the length field.

Table 7.2 shows the relationship between Bluetooth packet types and ECHONET frames. Six Bluetooth ACL packet types have been defined, but DM1 is to be used by default for ECHONET frames. Other ACL packet types may be used optionally.

Table 7.2 Relationship Between Bluetooth Packet Type and Echonet Frames

ACL packet type	Payload header byte	User data byte BT definition	Error correction code FEC	Error detection code CRC	Maximum data transfer rate Symmetric type, kbps	Maximum data transfer rate Asymmetric type, kbps		Use for ECHONET
						Forward	Backward	
DM1	1	0-17	2/3 rate	Used	108.8	108.8	108.8	Mandatory
DH1	1	0-27	Not used	Used	172.8	172.8	172.8	Optional
DM3	2	0-121	2/3 rate	Used	258.1	387.2	54.4	Optional
DH3	2	0-183	Not used	Used	390.4	585.6	86.4	Optional
DM5	2	0-224	2/3 rate	Used	286.7	477.8	36.3	Optional
DH5	2	0-339	Not used	Used	433.9	723.2	57.6	Optional

Bluetooth® has defined the error correction code to be attached to packets (FEC) and the error detection code for retransmission control (CRC) by ACL packet type.

Bluetooth® supports transmission packet encryption and internode authentication in the form of Bluetooth Security. For ECHONET nodes, the use of Bluetooth Security is strongly recommended to ensure radio security.

7.5.2 PAN profile

The PAN profile is specified for IP communication over Bluetooth®. The PAN Profile introduces two different scenarios for IP communication over Bluetooth: the Network Access Point (NAP) scenario and the Group Ad-hoc Network (GN) scenario.

In the Network Access Point scenario, Bluetooth nodes use the Internet and exchange IP packets via Bluetooth access points connected to the Internet.

The Group Ad-hoc Network scenario is primarily concerned with ad-hoc connections by mobile nodes. In this scenario, Bluetooth nodes connect to the PiconetMaster (Group Ad-hoc network), but communication is performed within the boundaries of the Bluetooth Piconet, although entirely through IP packets.

Figure 7.17 illustrates the Network Access Point and Group Ad-hoc Network scenarios. The node that serves as the access point in the Network Access Point scenario shall be referred to as the Network Access Point (NAP), the node that serves as master in the Group Ad-hoc Network scenario shall be referred to as the Group Ad-hoc Network (GN), and the nodes that establish Bluetooth connections to the masters in both scenarios (slaves) shall be referred to as PAN Users (PANUs).

The PAN Profile describes the methods for broadcasting and multicasting to NAPs and GNs. NAPs and GNs must transfer to the appropriate receiving nodes all packets they receive with a broadcast or multicast destination address.

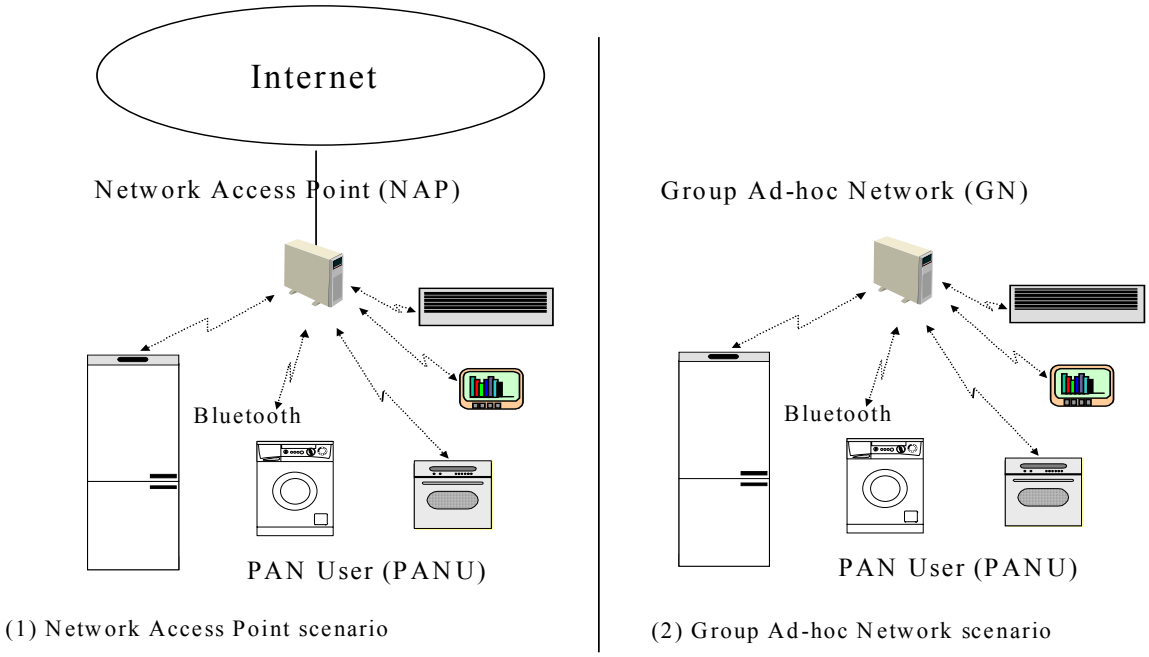


Fig. 7.17 Network Access Point and Group Ad-hoc Network Scenarios

The protocol specified for IP packet transfers over Bluetooth® is the Bluetooth Network Encapsulation Protocol (BNEP). As shown in Fig. 7.18, BNEP replaces the Ethernet header of each Ethernet frame with a BNEP header to form a packet and transfers it over the Bluetooth L2CAP link. The incorporation of an IP packet into this Ethernet payload enables the sending of IP packets over Bluetooth®.

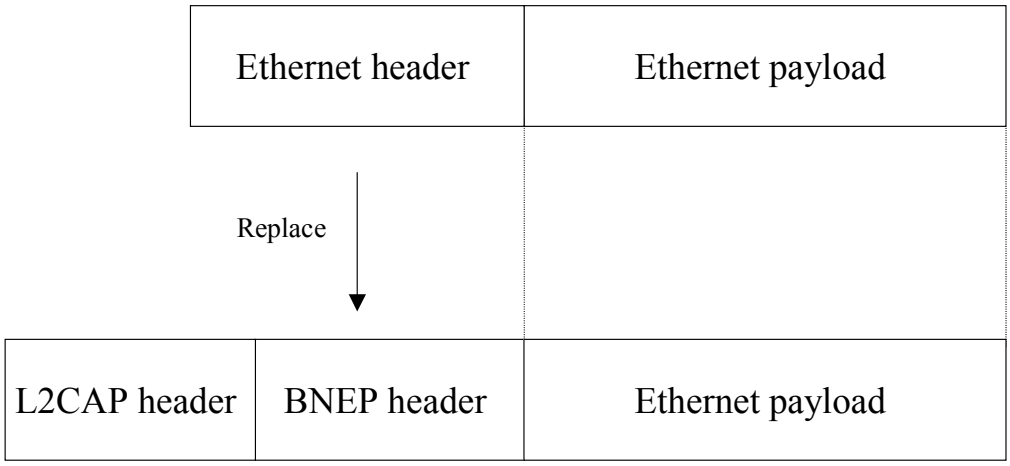
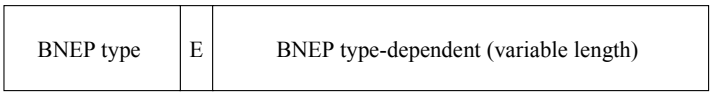
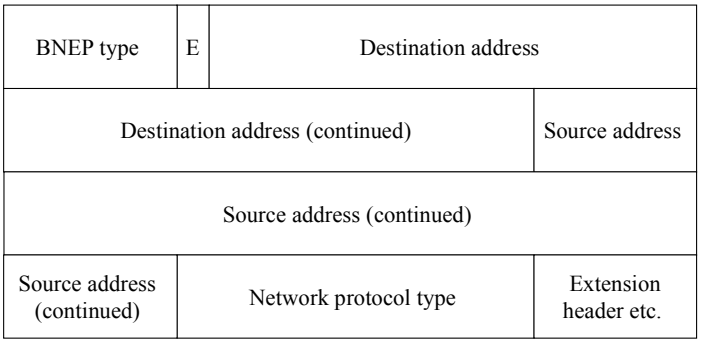


Fig. 7.18 BNEP Packet

Figure 7.19 (1) shows the basic format for the BNEP header. The BNEP type value determines the header formats that will be used subsequently. Figure 7.19 (2) shows an example of a basic BNEP header. When the node specified by the source address or destination address exists on the Bluetooth Piconet, the address may not be shown. For details, see the Bluetooth Specification "Bluetooth Network Encapsulation Protocol Version 1.0."



(1) BNEP Header Format



(2) Sample BNEP Header

Fig. 7.19 BNEP Header Format

With BNEP it is possible to make protocol type and multicast packet filtering settings. Protocol type filtering is a mechanism that allows a node to request the node on the other end to filter (reject) packets transmitted using a protocol other than the specified one(s). Multicast packet filtering is a mechanism that allows a node to make a request to the node on the other end concerning subscription to or unsubscription from a multicast address or multicast addresses. This is achieved by the sending of a filter control packet by a Bluetooth slave to the Bluetooth master.

Nodes that support ECHONET shall be regarded as supporting the bonding procedure.

One-to-one ECHONET communication between individual nodes shall be achieved using the unicast mechanism defined by the Bluetooth® BNEP. ECHONET broadcasting and group broadcasting shall be achieved using the multicast/broadcast and IP multicast mechanisms defined by the Bluetooth® BNEP (see Fig. 7.20).

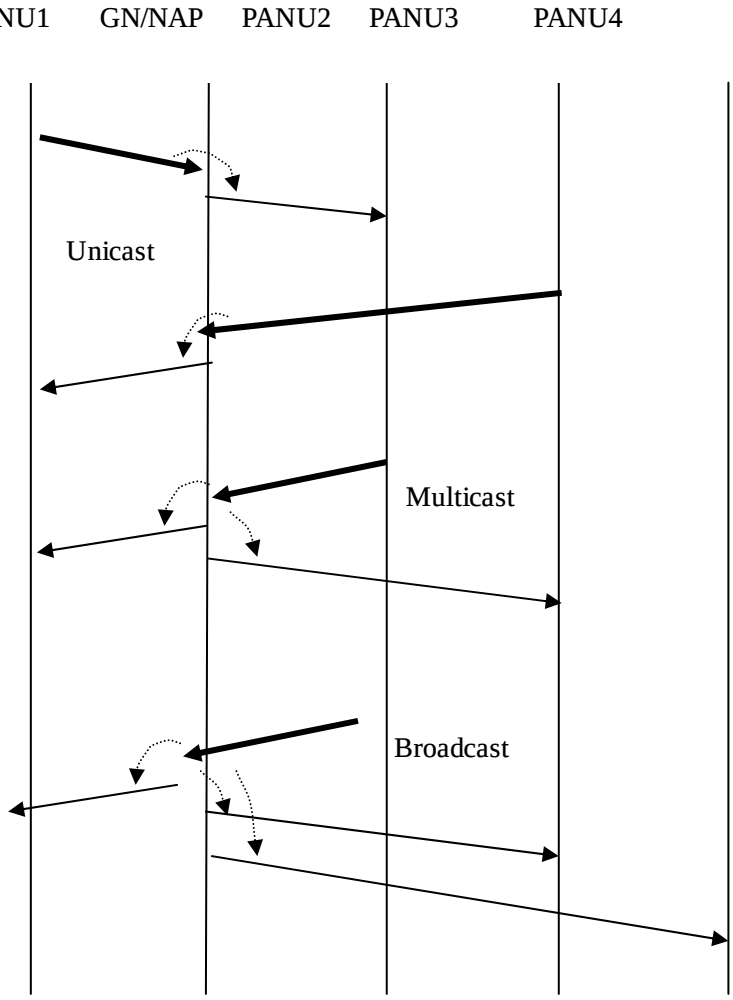


Fig. 7.20 Bluetooth® Unicast, Multicast, and Broadcast

7.6 Logical Specifications (IP Layer)

UDP/IP shall be used for the IP layer.

(1) IPv4

(A) Protocols to Be Used and Pertinent RFCs

For the ECHONET/IP layer, the following mandatory protocols shall be implemented:

IP v4	RFC 791 Internet Protocol
ARP	RFC 826 Address Resolution
ICMP	RFC 792, RFC 950
UDP	RFC 768 User Datagram Protocol
DHCP	RFC1541 Dynamic Host Configuration Protocol
	RFC1122 Requirements for Internet Hosts

The following RFCs are to be used for reference:

IGMP RFC1112 Internet Group Multicast Protocol

RFC1597 Address Allocation for Private Internets

(B) IP Address

Each node supporting this layer shall have an IP address. This Specification does not specify the range of IP addresses that can be used by individual nodes. Either a private or a global IP address may be used.

(C) Multicast Address

The multicast address ##.##.##.## (IPme) shall be used for this layer. Each node supporting this layer must be capable of sending packets to this address and receiving the packets sent to this address.

(D) DHCP

Each node supporting this layer shall have a function to obtain address setting information using a DHCP server. For operation, it is recommended that a DHCP server be deployed in the IP network.

(E) Method for Obtaining IP Address by Manual Setting, etc.

This Specification does not specify any IP address setting method other than DHCP.

(F) Routing

Operations in which ECHONET/IP packets are transferred beyond an IP router are not allowed. This layer must not send a packet addressed to a node located in another IP subnetwork. All packets received from a node or nodes located in another IP subnetwork must be discarded.

(2) UDP

(A) RFC for UDP

ECHONET/IP layer communication between nodes shall use UDP.

For more information on UDP, see the following RFC:

RFC768 User Datagram Protocol

(B) Port Number

The destination port number for UDP packets is to be ##.

This Specification does not specify an origination port number.

7.7 Logical Specifications (IP/Bluetooth Interface Layer)

7.1.1 DP interface

For access to UDP/IP by a UDP/IP application, a socket interface or equivalent interface is normally used. These interfaces depend on OS and development environment. For details, see the UDP/IP interface specifications for the specific development platform.

7.7.2 Packet format

ECHONET frames and their control packets (e.g., address resolution packets) are encapsulated into UDP packets when transferred over the Internet. These packets are transferred with the UDP port number *** attached (the application for the use of *** is currently being processed). This number is for the receiving port. The port number is the same regardless of packet type (unicast, multicast, or broadcast).

ECHONET frame types and control packets transferred using this UDP port are as follows:

- ECHONET frame transfer
- MAC/IP address resolution request/response (resolution of IP address from ECHONET MAC address)
- IP/MAC address resolution request/response (resolution of ECHONET/MAC address from IP address)
- Hardware/MAC address resolution request/response (resolution of ECHONET/MAC address from hardware address)
- MAC address initialization request/response, MAC address server initialization response
- MAC address allocation response
- MAC address confirmation request/response
- MAC Address request to all nodes /response
- MAC address server detection request/response, MAC address server notification

Because all these packet types are multiplexed onto the same UDP port, it is necessary to use packet type numbers for multiplexing. Therefore, these packets are multiplexed onto UDP packets using the format shown in Fig. 7.21 below. The entry in the "version number" section must always be 0x01.

Version number (0x01)	Packet type number	Packet type number-dependent

Fig. 7.21 Packet Format

Packet type numbers are shown in Table 7.3. "Compulsory for all nodes" in the "Support" column means that all ECHONET nodes must support packets having the packet type number shown. "Compulsory only for address servers" means that only those nodes that may become a MAC address server are required to support packets having the packet type number shown.

Table 7.3 Packet Type Numbers

Packet type number	Packet type	Support
0	ECHONET frame transfer	Compulsory for all nodes
1	MAC/IP address resolution request	Compulsory for all nodes
2	MAC/IP address resolution response	Compulsory for all nodes
3	IP/MAC address resolution request	Optional for all nodes
4	IP/MAC address resolution response	Compulsory for all nodes
5	Hardware/MAC address resolution request	Optional for all nodes
6	Hardware/MAC address resolution response	Compulsory for all nodes
7	MAC address initialization request	Compulsory for all nodes (not required for manual setting mode-only nodes)
8	MAC address initialization response	Compulsory for all nodes
9	MAC address server initialization response	Compulsory only for address servers
10	MAC address allocation response	Compulsory for all nodes (not required for manual setting mode-only nodes)
11	MAC address confirmation request	Compulsory for all nodes (not required for manual setting mode-only nodes)
12	MAC address confirmation response	Compulsory for all nodes
13	MAC Address request to all nodes	Compulsory only for address servers
14	MAC Address response to all nodes	Compulsory for all nodes
15	MAC address server detection request	Compulsory only for address servers
16	MAC address server notification	Compulsory only for address servers
17	MAC address server detection response	Compulsory only for address servers
Other than above	Reserved for future use	

The packet types are defined and their formats explained below.

In this chapter, the term "multicast" means "multicast to all ECHONET nodes in the ECHONET subnet" unless otherwise specified. Multicast packets are encapsulated into an IP packet addressed to the assigned IP multicast address (also referred to as "IPme") and sent to the ECHONET nodes.

The maximum hardware address length is 8 bytes.

Since this Chapter is based on the premise that the data link layer for the ECHONET nodes is Bluetooth, the value for "hardware type" in the explanation below will be "0x00" (the value assigned to Ethernet/Bluetooth), and the values for "hardware address length" and "hardware address" will be

"0x06" and the Bluetooth address value, respectively, throughout the remainder of this Chapter.

For fields with the comment "Enter null" or "Padding," "0x00" will be used throughout the remainder of this Chapter.

In the explanation of packet formats in this Chapter, the term "MSB" will refer to the leftmost bit and "LSM" to the rightmost bit. Each byte shall be transmitted with the most significant bit (MSB) first.

In the Flag field, "bit 7" will refer to the leftmost bit (MSB) and "bit 0" to the rightmost bit (LSB).

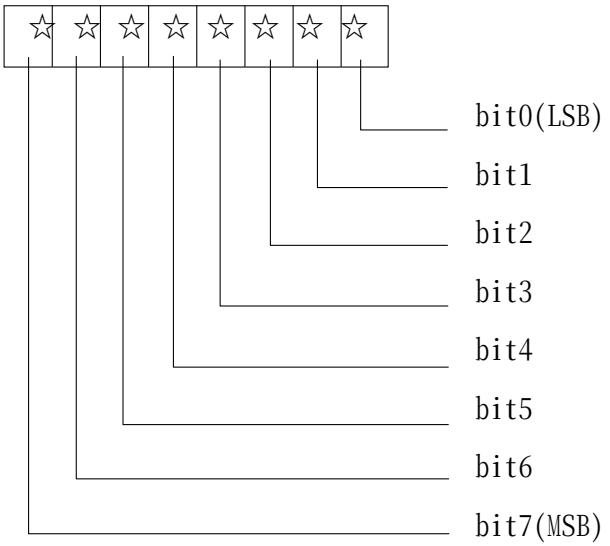


Fig. 7.22 Notation for Bits in the Flag Field

The unit of size used in this chapter is byte (octet).

An example is shown in Fig. 7.23 below.

When the Bluetooth address is used as the hardware address, "0x acde48000080" is entered into the packet (for the sample address data shown).

MSB LSB

company_id		company_assigned
NAP	UAP	LAP
1010110011011110	01001000	000000000000000010000000

Fig. 7.23 Bluetooth Address

IP addresses represented in dotted decimal notation are treated similarly. For example, "192.168.10.5" is entered into the packet as "0xc0a80a05".

(1) ECHONET frame transfer

(a) Ordinary ECHONET frame.

(b) The node that sends the ECHONET frame will be referred to as the "transmission node."

(c) The ECHONET frame is transferred with the transmission node's hardware type, hardware address length, and hardware address attached.

Table 7.4 Format for "ECHONET frame transfer" Packets

Item	Size	Explanation
Version	1	Enter "0x01" (Version 1)
Type	1	Enter "0x00" (ECHONET frame transfer)
Htype	1	Hardware type of transmission node
Hlen	1	Hardware address length of transmission node
Haddr	HLen	Hardware address of transmission node
Msg		Direct storage of ECHONET frame

(2) MAC/IP address resolution request/response

(a) Used to obtain the IP address of an ECHONET node that has an ECHONET MAC address.

(b) The node that wishes to resolve the ECHONET MAC address (i.e., that wishes to know the IP address) will be referred to as the "requesting node," and the node that notifies the relationship between the addresses in response to the request will be referred to as the "target node."

(c) The requesting node multicasts to the ECHONET subnet an IP address resolution request packet containing the ECHONET MAC address to be resolved. Upon receipt of the request, the target node sends an IP address resolution response packet containing the relevant addresses.

Table 7.5 below shows the format for MAC/IP address resolution request packets.

Table 7.5 Format for "MAC/IP address resolution request" Packets

Item	Size	Explanation
Version	1	Enter "0x01" (Version 1)
Type	1	Packet type (IP address resolution request; enter "0x01")
Padding	1	Padding
RMAC	1	ECHONET MAC address of requesting node
RIPAddr	4	IP address of requesting node
RHType	1	Hardware type of requesting node
RHLen	1	Hardware address length of requesting node
RHAddr	RHLen	Hardware address of requesting node
Padding	1	Padding
TMAC	1	ECHONET MAC address of target node
TIPAddr	4	IP address of target node (Enter "null")
THType	1	Hardware type of target node (Enter "null")
THLen	1	Hardware address length of target node (Enter "RHLen")
THAddr	THLen	Hardware address of target node (Enter "null")

Table 7.6 below shows the format for MAC/IP address resolution response packets.

Table 7.6 Format for "MAC/IP address resolution response" Packets

Item	Size	Explanation
Version	1	Enter "0x01" (Version 1)
Type	1	Packet type (IP address resolution response; enter "0x02")
Padding	1	Padding
RMAC	1	ECHONET MAC address of requesting node
RIPAddr	4	IP address of requesting node
RHType	1	Hardware type of requesting node
RHLen	1	Hardware address length of requesting node
RHAddr	RHLen	Hardware address of requesting node
Padding	1	Padding
TMAC	1	ECHONET MAC address of target node
TIPAddr	4	IP address of target node
THType	1	Hardware type of target node
THLen	1	Hardware address length of target node
THAddr	THLen	Hardware address of target node

(3) IP/MAC address resolution request/response

(a) Used to obtain the ECHONET MAC address of an ECHONET node that has an IP address.

(b) The node that wishes to resolve the IP address (i.e., that wishes to know the ECHONET MAC address) will be referred to as the "requesting node," and the node that notifies the relationship between the addresses in response to the request will be referred to as the "target node."

(c) The requesting node sends to the destination IP address an IP/ECHONET address resolution request packet containing the target IP address to be resolved. Upon receipt of the request, the target node sends an IP/ECHONET address resolution response packet containing the relevant addresses.

Table 7.7 below shows the format for IP/MAC address resolution request packets.

Table 7.7 Format for "IP/MAC address resolution request" Packets

Item	Size	Explanation
Version	1	Enter "0x01" (Version 1)
Type	1	Packet type (IP/ECHONET address resolution request; enter "0x03")
Padding	1	Padding
RMAC	1	ECHONET MAC address of requesting node
RIPAddr	4	IP address of requesting node
RHType	1	Hardware type of requesting node
RHLen	1	Hardware address length of requesting node
RHAddr	RHLen	Hardware address of requesting node
Padding	1	Padding
TMAC	1	ECHONET MAC address of target node (Enter "null")

TIPAddr	4	IP address of target node
THType	1	Hardware type of target node (Enter "null")
THLen	1	Hardware address length of target node (Enter "RHLen")
THAddr	THLen	Hardware address of target node (Enter "null")

Table 7.8 below shows the format for IP/MAC address resolution response packets.

Table 7.8 Format for "IP/MAC address resolution response" Packets

Item	Size	Explanation
Version	1	Enter "0x01" (Version 1)
Type	1	Packet type (IP/ECHONET address resolution response; enter "0x04")
Padding	1	Padding
RMAC	1	ECHONET MAC address of requesting node
RIPAddr	4	IP address of requesting node
RHType	1	Hardware type of requesting node
RHLen	1	Hardware address length of requesting node
RHAddr	RHLen	Hardware address of requesting node
Padding	1	Padding
TMAC	1	ECHONET MAC address of target node
TIPAddr	4	IP address of target node
THType	1	Hardware type of target node
THLen	1	Hardware address length of target node
THAddr	THLen	Hardware address of target node

(4) Hardware/MAC address resolution request/response

(a) Used to obtain the ECHONET MAC address of an ECHONET node that has a hardware address.

(b) The node that wishes to resolve the hardware address (Bluetooth address, etc.) (i.e., that wishes to know the ECHONET MAC address) will be referred to as the "requesting node," and the node that notifies the relationship between the addresses in response to the request will be referred to as the "target node."

(c) The requesting node multicasts to the ECHONET subnet a hardware/ECHONET address resolution request packet containing the target hardware type, hardware address length, and hardware address to be resolved. Upon receipt of the request, the target node sends a hardware/ECHONET address resolution response packet containing the relevant addresses.

Table 7.9 below shows the format for hardware/MAC address resolution request packets.

Table 7.9 Format for "hardware/MAC address resolution request" Packets

Item	Size	Explanation
Version	1	Enter "0x01" (Version 1)
Type	1	Packet type (hardware/ECHONET address resolution request; enter "0x05")

Padding	1	Padding
RMAC	1	ECHONET MAC address of requesting node
RIPAddr	4	IP address of requesting node
RHType	1	Hardware type of requesting node
RHLen	1	Hardware address length of requesting node
RHAddr	RHLen	Hardware address of requesting node
Padding	1	Padding
TMAC	1	ECHONET MAC address of target node (Enter "null")
TIPAddr	4	IP address of target node (Enter "null")
THType	1	Hardware type of target node
THLen	1	Hardware address length of target node
THAddr	THLen	Hardware address of target node

Table 7.10 below shows the format for hardware/MAC address resolution response packets.

Table 7.10 Format for "hardware/MAC address resolution response" Packets

Item	Size	Explanation
Version	1	Enter "0x01" (Version 1)
Type	1	Packet type (hardware/ECHONET address resolution response; enter "0x06")
Padding	1	Padding
RMAC	1	ECHONET MAC address of requesting node
RIPAddr	4	IP address of requesting node
RHType	1	Hardware type of requesting node
RHLen	1	Hardware address length of requesting node
RHAddr	RHLen	Hardware address of requesting node
Padding	1	Padding
TMAC	1	ECHONET MAC address of target node
TIPAddr	4	IP address of target node
THType	1	Hardware type of target node
THLen	1	Hardware address length of target node
THAddr	THLen	Hardware address of target node

(5) MAC address initialization request/response

(a) Used by ECHONET nodes to initialize their MAC addresses at boot time.

(b) The booting node (i.e., the node requesting a MAC address initialization) will be referred to as the "requesting node."

(c) When an ECHONET node boots (whether warm or cold), it multicasts to the ECHONET subnet a MAC address initialization request packet to initiate the MAC address initialization process. Any ECHONET mode that boots up in Automatic Mode (A-MODE) or Server Required Mode (SR-MODE) must send this packet at boot time.

(d) A MAC address initialization request packet functions to:

① confirm whether or not a MAC address server exists in the ECHONET subnet;
② ask all other nodes located in the ECHONET subnet to send their MAC address initialization response packets (a MAC address initialization response packet contains information on the relationship between the MAC address, IP address, and other relevant addresses of the node in question); and
③ confirm whether or not the subnet has an ECHONET node already using a MAC address that is the same as the MAC address the requesting node intends to use (i.e., the provisional MAC address set by the requesting node).

(e) Upon receipt of the MAC address initialization request packet, the other ECHONET nodes (excluding the MAC address server) send to the requesting node their MAC address initialization response packets.

Table 7.11 below shows the format for MAC address initialization request packets.

Table 7.11 Format for "MAC address initialization request" Packets

Item	Size	Explanation
Version	1	Enter "0x01" (Version 1)
Type	1	Packet type (MAC address initialization request; enter "0x07")
Flag	1	When inServer Required Mode, the value in the bit 7 field is "1", and when in other thanServer Required Mode, the value is "0". Bit 0 to bit 6 are reserved and "0" will be entered in the fields for these bits.
RMAC	1	Provisional ECHONET MAC address of requesting node
RIPAddr	4	IP address of requesting node
RHType	1	Hardware type of requesting node
RHALen	1	Hardware address length of requesting node
RHAddr	RHALen	Hardware address of requesting node

Table 7.12 below shows the format for MAC address initialization response packets.

Table 7.12 Format for "MAC address initialization response" Packets

Item	Size	Explanation
Version	1	Enter "0x01" (Version 1)
Type	1	Packet type (MAC address initialization response; enter "0x08")
Flag	1	For a node that is a master router (see Part 2 of the Standard), "1" will be entered in the field for bit 7. Otherwise use "0". Bit 0 to bit 6 are reserved and "0" will be entered in the fields for these bits.
TMAC	1	ECHONET MAC address of responding node
TIPAddr	4	IP address of responding node
THType	1	Hardware type of responding node
THALen	1	Hardware address length of responding node
THAddr	THALen	Hardware address of responding node
UsedMAC	32	"MAC address in use" flag (if MAC address n is currently being used, bit n is "1"). (LSB corresponds to MAC address "0" and MSB corresponds to MAC address "255".)

(6) MAC address server initialization response / MAC address allocation response

(a) Upon receipt of a MAC address initialization request packet from a booting node, the MAC address server sends a MAC address server initialization response packet to the booting node. The node receiving this packet sends a MAC address allocation response packet to the MAC address server to confirm receipt of the packet.

(b) The booting node (i.e., the node requesting a MAC address initialization) will be referred to as the "requesting node."

(c) The MAC address server initialization response packet sent by the MAC address server to the requesting node in response to the MAC address initialization request packet contains the ECHONET MAC address to be used by the requesting node. The requesting node receiving this packet must use the ECHONET MAC address specified.

(d) Upon receipt of the MAC address server initialization response packet, the ECHONET node sends a MAC address allocation response packet to the MAC address server.

Table 7.13 below shows the format for MAC address server initialization response packets.

Table 7.13 Format for "MAC address server initialization response" Packets

Item	Size	Explanation
Version	1	Enter "0x01" (Version 1)
Type	1	Packet type (MAC address server initialization response; enter "0x09")
Flag	1	For a master router node, "1" will be entered in the field for bit 7. Otherwise use "0". Bit 0 to bit 6 are reserved and "0" will be entered in the fields for these bits.
Padding	1	Padding
RMAC	1	MAC address to be used by requesting node
SMAC	1	ECHONET MAC address of address server node
SIPAddr	4	IP address of address server node
SHType	1	Hardware type of address server node
SHALen	1	Hardware address length of address server node
SHAddr	SHALen	Hardware address of address server node

Table 7.14 below shows the format for MAC address allocation response packets.

Table 7.14 Format for "MAC address allocation response" Packets

Item	Size	Explanation
Version	1	Enter "0x01" (Version 1)
Type	1	Packet type (MAC address allocation response; enter "0x0a")
SMAC	1	ECHONET MAC address of MAC address server
RMAC	1	ECHONET MAC address to be used by requesting node
RIPAddr	4	IP address of requesting node
RHType	1	Hardware type of requesting node

RHALen	1	Hardware address length of requesting node
RHAddr	RHALen	Hardware address of requesting node

(7) MAC address confirmation request/response

(a) When the requesting node learns through the MAC address initialization request/response process described above that the MAC address it intended to use (i.e., the provisional MAC address set by the requesting node) is already being used by another node, the requesting node uses a MAC address confirmation request packet to set a different provisional MAC address and confirm whether or not there is a node in the ECHONET subnet that is already using the new provisional MAC address.

(b) ECHONET nodes receiving the MAC address confirmation request packet check whether or not their MAC addresses coincide with the provisional MAC address contained in the packet. An ECHONET node with a MAC address identical to the provisional MAC address sends a MAC address confirmation response packet to the requesting node.If there is no response to the MAC address confirmation request packet within a certain time, the requesting node determines that no node in the ECHONET subnet is using the provisional ECHONET MAC address and assumes the provisional ECHONET MAC address as its formal ECHONET MAC address.

Table 7.15 below shows the format for MAC address confirmation request packets.

Table 7.15 Format for "MAC address confirmation request" Packets

Item	Size	Explanation
Version	1	Enter "0x01" (Version 1)
Type	1	Packet type (MAC address confirmation request; enter "0x0b")
Padding	1	Padding
RMAC	1	Provisional ECHONET MAC address of requesting node
RIPAddr	4	IP address of requesting node
RHType	1	Hardware type of requesting node
RHALen	1	Hardware address length of requesting node
RHAddr	HALen	Hardware address of requesting node

Table 7.16 below shows the format for MAC address confirmation response packets.

Table 7.16 Format for "MAC address confirmation response " Packets

Item	Size	Explanation
Version	1	Enter "0x01" (Version 1)
Type	1	Packet type (MAC address confirmation response; enter "0x0c")
Flag	1	For a master router, "1" will be entered in the field for bit 7. Otherwise use "0". Bit 0 to bit 6 are reserved and "0" will be entered in the fields for these bits.
TMAC	1	ECHONET MAC address of responding node
TIPAddr	4	IP address of responding node
THType	1	Hardware type of responding node
THALen	1	Hardware address length of responding node
THAddr	THALen	Hardware address of responding node

(8) MAC Address request to all nodes /response

(a) A node in an ECHONET subnet can ask other nodes in the subnet to send their MAC Address response to all node packets (a MAC Address response to all node packet contains information on the relationship between the MAC address, IP address, and other relevant addresses of the node in question) by multicasting a MAC Address request to all nodes packet to the subnet.

(b) Upon receipt of the MAC Address request to all nodes packet, the ECHONET nodes send their MAC Address response to all node packets containing the relevant addresses to the requesting node.

Table 7.17 below shows the format for MAC Address request to all nodes packets.

Table 7.17 Format for " MAC Address request to all nodes" Packets

Item	Size	Explanation
Version	1	Enter "0x01" (Version 1)
Type	1	Packet type (MAC Address request to all nodes; enter "0x0d")
RMAC	1	ECHONET MAC address of requesting node
RIPAddr	4	IP address of requesting node
RHType	1	Hardware type of requesting node
RHALen	1	Hardware address length of requesting node
RHAddr	HALen	Hardware address of requesting node

Table 7.18 below shows the format for MAC Address response to all node packets.

Table 7.18 Format for " MAC Address response to all node" Packets

Item	Size	Explanation
Version	1	Enter "0x01" (Version 1)
Type	1	Packet type (MAC Address response to all node; enter "0x0e")
Padding	1	Padding
TMAC	1	ECHONET MAC address of responding node
TIPAddr	4	IP address of responding node
THType	1	Hardware type of responding node
THALen	1	Hardware address length of responding node
THAddr	THALen	Hardware address of responding node

(9) MAC address server detection request/response, MAC address server notification

(a) A node in a subnet without a MAC address server that wishes to become the MAC address server can use MAC address server detection request/response and MAC address server notification packets.

(b) The node seeking to become the MAC address server will be referred to as the "requesting node."

(c) The requesting node multicasts a MAC address server detection request packet to the ECHONET subnet. If there is no response within a certain time, the requesting node determines that there is no MAC address server in the subnet and multicasts a MAC address server notification packet to the ECHONET subnet to declare that it will serve as the MAC address server.

Table 7.19 below shows the format for MAC address server detection request packets.

Table 7.19 Format for "MAC address server detection request" Packets

Item	Size	Explanation
Version	1	Enter "0x01" (Version 1)
Type	1	Packet type (MAC address server detection request; enter "0x0f")
RMAC	1	ECHONET MAC address of requesting node

Table 7.20 below shows the format for MAC address server notification packets.

Table 7.20 Format for "MAC address server notification" Packets

Item	Size	Explanation
Version	1	Enter "0x01" (Version 1)
Type	1	Packet type (MAC address server notification; enter "0x10")
Padding	1	Padding
SMAC	1	ECHONET MAC address of MAC address server node
SIPAddr	4	IP address of MAC address server node
SHType	1	Hardware type of MAC address server node
SHALen	1	Hardware address length of MAC address server node
SHAddr	SHALen	Hardware address of MAC address server node

(d) If a MAC address server already exists in the subnet, a MAC address server detection response packet is sent in response to the MAC address server detection request packet.

Table 7.21 below shows the format for MAC address server detection response packets.

Table 7.21 Format for "MAC address server detection response" Packets

Item	Size	Explanation
Version	1	Enter "0x01" (Version 1)
Type	1	Packet type (MAC address server detection response; enter "0x11")
Padding	1	Padding
SMAC	1	ECHONET MAC address of MAC address server node
SIPAddr	4	IP address of MAC address server node
SHType	1	Hardware type of MAC address server node
SHALen	1	Hardware address length of MAC address server node
SHAddr	SHALen	Hardware address of MAC address server node

(10) Network management message (destination address unidentified)

(a) When DMAC (destination ECHONET MAC address) value of receivred ECHONET frame packet (Table 7.4) is different from the ECHONET MAC address of own node, the packet is to nofify destination address unidentified to the node sending the ECHONET frame tranfer packet.

(b) The node sending the network management message (destination address unidentified) is called

"sending node", and The node receiving the massage is called "receiving node".

(c) The node receiving the message (receiving node) shall do address resolution (MAC/IP address resolution) again.

Table 7.22 Format for network management message (destination address unidentified) packets

Item	Size	Explanation
Version	1	Enter "0x01" (Version 1)
Type	1	Packet type (enter "0x12" destination address unidentified)
Padding	1	Padding
SMAC	1	ECHONET MAC address of sending node
SIPAddr	4	IP address of sending node
SHType	1	Hardware type of sending node
SHALen	1	Hardware address length of sending node
SHAddr	SHALen	Hardware address of sending node
DMAC	1	DMAC value written in received ECHONET frame transfer packet

(11) Network management message (ECHONET MAC address duplicated)

(a) The packet is for a node discovering ECHONET MAC address duplicated (2 or more nodes have the same ECHONET MAC address) to notify to relevant node

(b) The node sending the message is called "sending node".

(c) The massage is broadcasted in the subnet.

(d) The node (duplicate ECHONET MAC address) receiving the massage shall do address determination adain after confirmation of duplication.

Table 7.23 Format for network management message (ECHONET MAC address duplicated) packets

Item	Size	Explanation
Version	1	Enter "0x01" (Version 1)
Type	1	Packet type (enter "0x13" ECHONET MAC address duplicated)
Padding	1	Padding
SMAC	1	ECHONET MAC address of sending node
SIPAddr	4	IP address of sending node
SHType	1	Hardware type of sending node
SHALen	1	Hardware address length of sending node
SHAddr	SHALen	Hardware address of sending node
DMAC	1	Duplicated ECHONET MAC address

7.7.3 Basic communication sequences

An ECHONET node first performs the ECHONET MAC address initialization procedure (for details on this procedure, see 7.7.4 and 7.7.5). Upon completion of this procedure, the ECHONET node's ECHONET MAC address is established and the ECHONET node is now ready to be handled by the ECHONET Middleware.

In an IP network, the IP address of a node may change (for several reasons, including the fact that the IP address allocated by the DHCP server may differ from time to time). Similarly, the ECHONET MAC address of a node

may change for such reasons as a change in the location of the node and an ECHONET MAC address overlap. Therefore, the ECHONET nodes must always have the latest address information (i.e., information on the relationship between the hardware, IP and ECHONET MAC addresses of each of the nodes). Table 7.22 below shows a sample address relation table.

Table 7.24 Sample Address Relation Table

Hardware type	Hardware address	IP address	ECHONET MAC address
1	Ha	IPa	MACa
1	Hb	IPb	MACb
1	Hc	IPc	MACc
1	…	…	…

For this reason, all ECHONET nodes must collect address information on surrounding nodes (by sending an ECHONET MAC address initialization packet and receiving the response) at boot time and notify the ECHONET nodes located in the domain of their established address to keep their address relation tables up to date. The timeout periods contained in the tables are implementation-dependent.

An ECHONET node with an established ECHONET address can perform ECHONET communication. Such a node may encounter any of the following cases:

(1) It knows the ECHONET MAC address of the node with which it wishes to communicate but does not know the node's IP address;
(2) It knows the IP address of the node with which it wishes to communicate but does not know the node's ECHONET MAC address; or
(3) It knows the hardware address (e.g., Bluetooth address) of the node with which it wishes to communicate but does not know the node's ECHONET MAC address.

To solve the problem the ECHONET node can use 1) MAC/IP address resolution request/response packets, 2) IP/MAC address resolution request/response packets, or 3) hardware/MAC address resolution request/response packets. A detailed explanation of each method follows.

(1) MAC/IP address resolution request/response packets (resolution of the ECHONET MAC address into the IP address)

(a) The node seeking to resolve the ECHONET MAC address will be referred to as the "requesting node," and the node on which resolution is to be performed will be referred to as the "target node."

(b) An ECHONET node uses a MAC/IP address resolution request packet when it wishes to know the IP address of another ECHONET node with an ECHONET MAC address.

(c) The requesting node multicasts to the ECHONET subnet an address resolution request packet containing the target ECHONET MAC address to be resolved. Upon receipt of the request, the target node sends a MAC/IP address resolution response packet that contains its ECHONET MAC address, IP address, transmission medium hardware type, hardware address length, and hardware address.

(d) The timeout period between the transmission of the IP address resolution request and the receipt of the

IP address resolution response packet shall be T3. If no IP address resolution response packet is received within the timeout period, an error will occur.

(e) It is recommended that the content of the IP address resolution response packet be reflected in the address relation table.

(f) To prevent an ARP Flood, the frequency of MAC/IP address resolution request packet transmissions shall be one packet per second or less. The maximum number of MAC/IP address resolution request packets that can be sent in succession shall be 5 (the interval between a MAC/IP address resolution request packet and the succeeding one must be at least 1 second). If no MAC/IP address resolution response packet is received after the 5th attempt, an error will occur.

The formats for MAC/IP address resolution request and response packets are shown in Tables 7.5 and 7.6, respectively.

The basic sequence is as shown in Fig. 7.24 below.

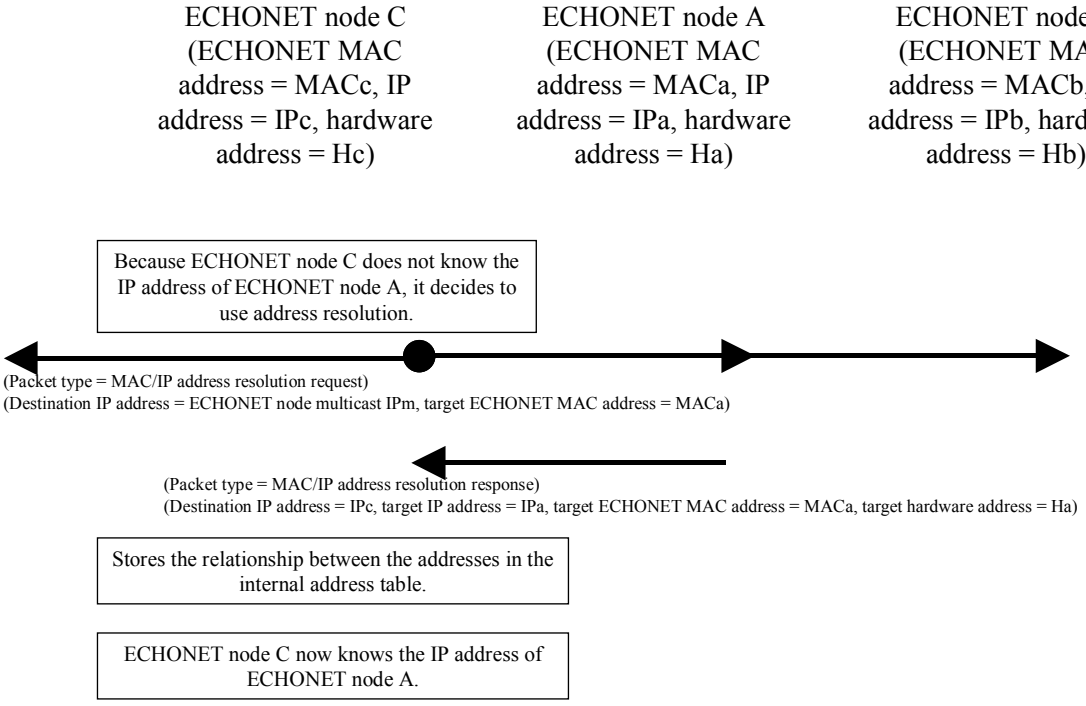


Fig. 7.24 Basic MAC/IP Address Resolution Sequence

(2) IP/MAC address resolution request/response (resolution of IP address into ECHONET MAC address)

(a) The node seeking to resolve the IP address will be referred to as the "requesting node," and the node on which resolution is to be performed will be referred to as the "target node."

(b) An ECHONET node uses an IP/MAC address resolution request packet when it wishes to know the ECHONET MAC address of another ECHONET node with an IP address.

(c) The requesting node sends to the destination IP address an IP/MAC address resolution request packet

containing the target IP address to be resolved. Upon receipt of the request, the target node sends an IP/MAC address resolution response packet containing its ECHONET MAC address, IP address, transmission medium hardware type, hardware address length, and hardware address.

(d) The timeout period between the transmission of the IP/MAC address resolution request packet and the receipt of the IP/MAC address resolution response packet shall be T4. If no IP address resolution response packet is received within the timeout period, an error will occur.

(e) It is recommended that the content of the IP/MAC address resolution response packet be reflected in the address relation table.

(f) To prevent an ARP Flood, the frequency of IP/MAC address resolution request packet transmissions shall be one packet per second or less. The maximum number of IP/MAC address resolution request packets that can be sent in succession shall be 5 (the interval between an IP/MAC address resolution request packet and the succeeding one must be at least 1 second). If no IP/MAC address resolution response packet is received after the 5th attempt, an error will occur.

The formats for IP/MAC address resolution request and response packets are shown in Tables 7.7 and 7.8, respectively.

The basic sequence is as shown in Fig. 7.25 below.

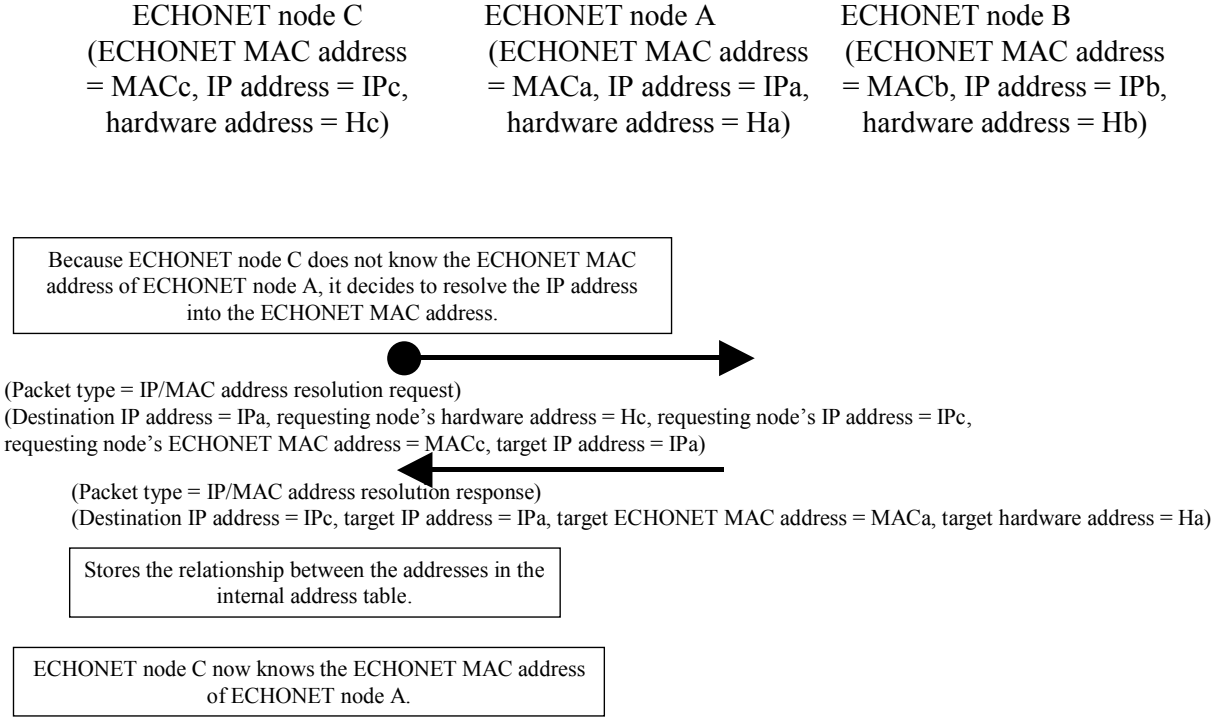


Fig. 7.25 Basic IP/MAC Address Resolution Sequence

(3) Hardware/MAC address resolution request/response

(a) The node seeking to resolve the hardware address (e.g., Bluetooth address) will be referred to as the "requesting node," and the node on which resolution is to be performed will be referred to as the "target node."
(b) An ECHONET node uses a hardware/MAC address resolution request packet when it wishes to know the ECHONET MAC address of another ECHONET node with a hardware address.
(c) The requesting node multicasts to the ECHONET subnet a hardware/MAC address resolution request packet that contains the target hardware type, hardware address length, and hardware address to be resolved. Upon receipt of the request, the target node sends a hardware/MAC address resolution response packet containing its ECHONET MAC address, IP address, transmission medium hardware type, hardware address length, and hardware address.
(d) The timeout period between the transmission of the hardware/MAC address resolution request packet and the receipt of the hardware/MAC address resolution response packet shall be T4. If no IP address resolution response packet is received within the timeout period, an error will occur.
(e) It is recommended that the content of the hardware/MAC address resolution response packet be reflected in the address relation table.
(f) To prevent an ARP Flood, the frequency of hardware/MAC address resolution request packet transmissions shall be one packet per second or less. The maximum number of hardware/MAC address resolution request packets that can be sent in succession shall be 5 (the interval between a hardware/MAC address resolution request packet and the succeeding one must be at least 1 second). If no hardware/MAC address resolution response packet is received after the 5th attempt, an error will occur.

The format for hardware/MAC ECHONET address resolution request packets and hardware/MAC address resolution response packets are shown in Tables 7.9 and 7.10, respectively.

The basic sequence is as shown in Fig. 7.26 below.

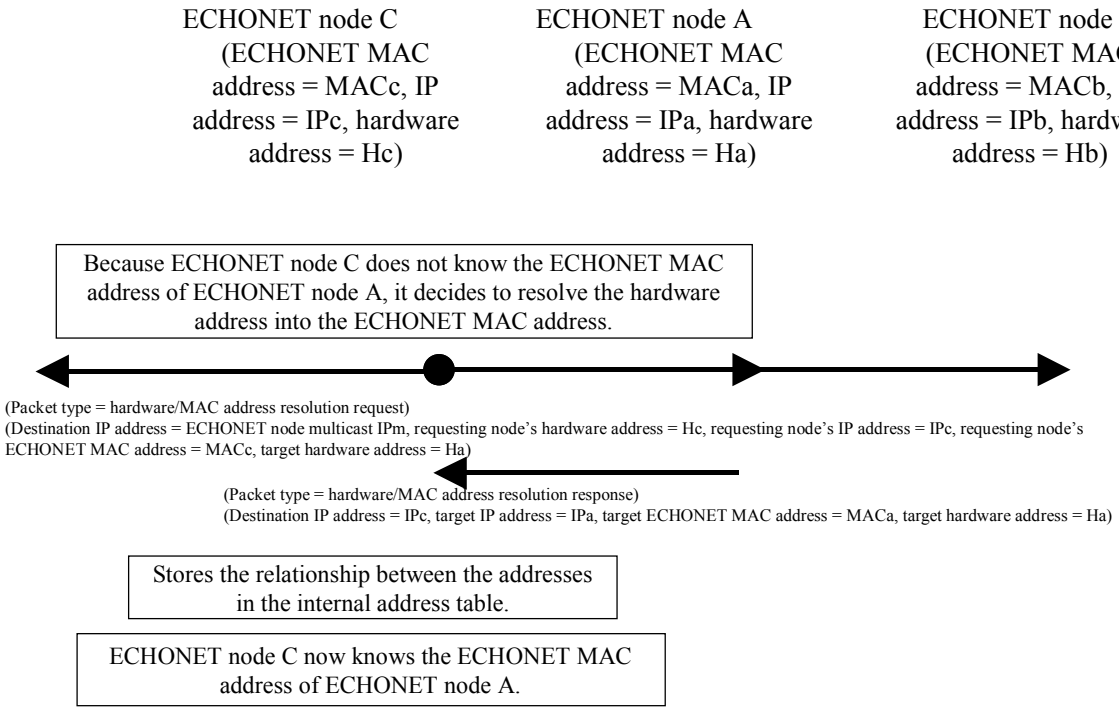


Fig. 7.26 Basic Hardware/MAC Address Resolution Sequence

Unicast ECHONET frames are mapped onto unicast UDP/IP packets. Multicast/broadcast ECHONET frames are mapped onto UDP/IP packets addressed to a dedicated IP multicast address assigned for ECHONET (***; the application for the use of *** is currently being processed).

In ECHONET, the mapping of ECHONET subnets onto IP subnets is always one-to-one. Therefore, no ECHONET node may send an ECHONET frame (or an ECHONET control packet) to a node located in another IP subnet (in other words, no ECHONET node may send an IP packet with a destination IP address representing another IP subnet). By the same token, no ECHONET node may receive an ECHONET frame (or an ECHONET control packet) from a node located in another IP subnet. In other words, an ECHONET node receiving an IP packet with an origination IP address representing another IP subnet must discard the packet.

(4) MAC address entire nodes request / response

(a) The nodes investigating ECHONET MAC address of the entire ECHONET nodes in the ECHONET subnet is called "requesting nodes" and the nodes responding the request is called "responding nodes"

(b) For example, it is used for MAC address server to investigate MAC address of the entire ECHONET nodes belonging to the MAC address server. The MAC address server is a requesting node in this case.

(c) The requesting node multicasts MAC address request packets including the own ECHONET MAC address to the ECHONET subnet. In response to the packet, the entire nodes in the ECHONET subnet transmits the MAC address entire nodes responding packets including the own ECHONET MAC address, the IP address, hardware address length of hardware type of transmission media,

hardware address, etc.

(d) The time out period between transmitting MAC address entire nodes requests and receiving MAC address entire nodes responses is defined to be T14. When the MAC address entire nodes responses cannot be received within the time out period, it means that other ECHONET nodes do not exist in the ECHONET subnet.

(e) It is recommended that the result of MAC address entire nodes responses should be reflected to the address corresponding table.

The format for MAC address entire nodes requesting packets is shown in Table 7.17 and the format for MAC address entire nodes responding packets is shown in Table 7.18 respectively.
The sequence chart is shown below.

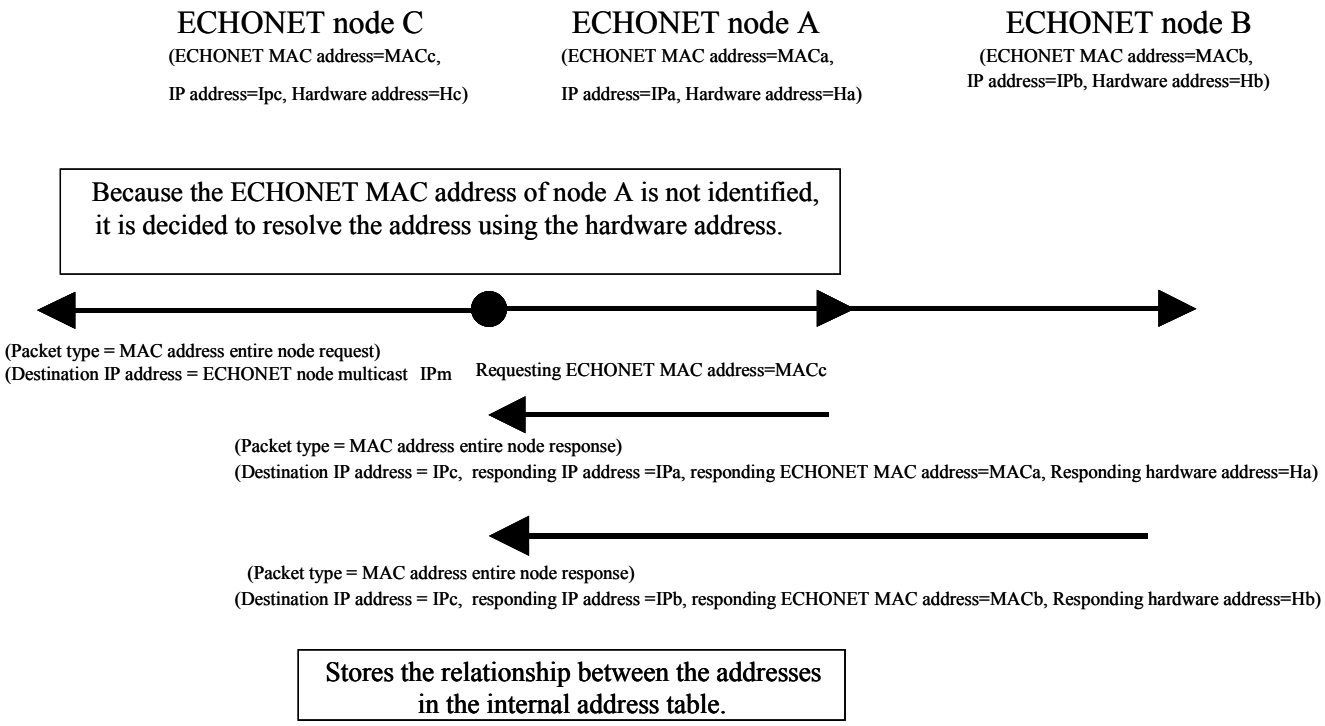


Fig. 7.27 Basic Sequence of MAC Address Entire nodes Request / Response

(5) Network management message (Destination unidentified)

(a) When a DMAC (ECHONET MAC address of receiving node) value of ECHONET frame transfer packet (see Table 7.4) received by a node is different from ECHONET MAC address value of own node, the receiving node cannot give the ECHONET frame transfer packet to upper layers. Instead, the receiving node notifies to the sending node that the received ECHONET frame transfer packet was not identified the destination.

(b) A node that sends the original ECHONET frame transfer packet is called a sending node, a node

that receives the packet is called a receiving node. Therefore, a node that sends the network management message is “receiving node” and a node that receives the message is “sending node”.

(c) The node that received the message (sending node) shall do address resolution (MAC/IP address resolution) again.

The format of network management message (Destination unidentified) is shown in Fig. 7.21. The sequence is shown below.

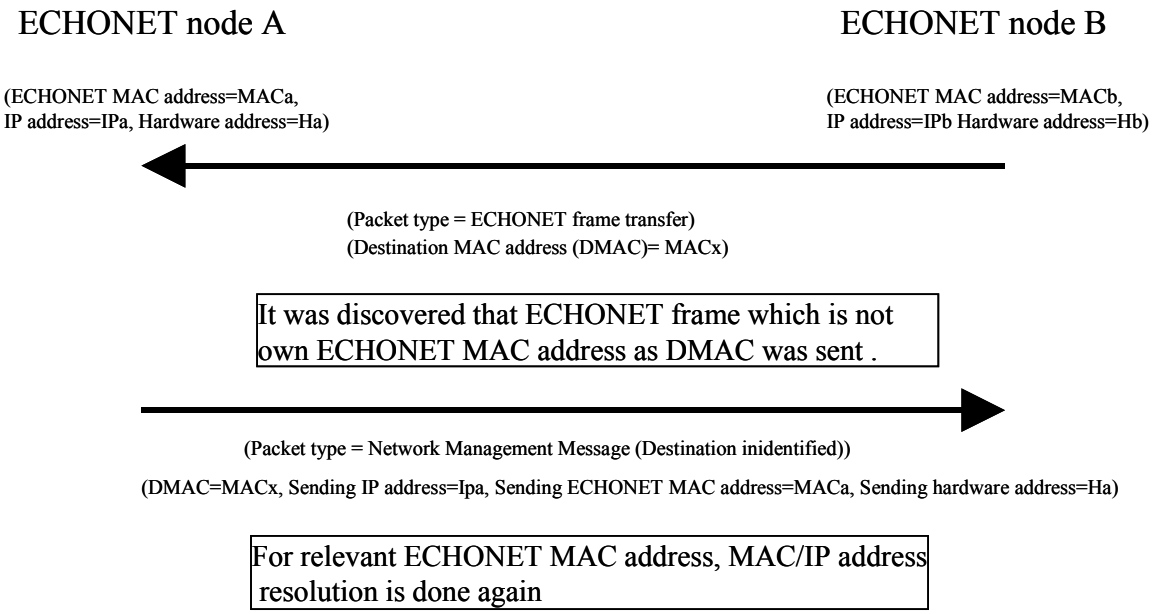


Fig. 7.28 Basic Sequence of process for Destination Unidentified

However, the process needs not be done “when the SMAC value of received ECHONET frame transfer packet is the same as the DMAC value of the packet (when the ECHONET frame is a broadcast packet or a group broadcast packet)”.

It is likely that “the SMAC value of received ECHONET frame transfer corresponds to the address value of own ECHONET MAC address”. In this case, it is likely that the ECHONET MAC address is duplicated in the subnet. The process shown in below (5) shall be done, then.

(6) A Special Case of Destination Unidentified

(a) When a destination unidentified ECHONET frame packet is transferred, the sending address value (SMAC) of the packet corresponds to the address value of own ECHONET MAC address, it is likely

that the ECHONET MAC address is duplicated in the subnet.

(b) In this case, the node received the destination unidentified ECHONET frame transfer packet does MAC/IP address resolution request that the sending address (SMAC) of the packet is set to be an ECHONET MAC address of the target node.

(c) If the MAC/IP address resolution response returns (duplication of ECHONET MAC address was confirmed), the node received the destination unidentified ECHONET frame transfer packet initializes own ECHONET address by cold start by itself.

(d) If the MAC/IP address resolution response does not return, it is judged an uncertain packet was received and the destination unidentified ECHONET frame transfer packet is discarded.

(7) Network Management Message (ECHONET MAC address duplicated)

(a) The code discovering ECHONET MAC address duplicated (a state that 2 or more nodes have the same ECHONET MAC address) notifies this to relevant nodes. An ECHONET MAC address duplicated is detected when MAC/IP address resolution responses for identical ECHONET MAC address are received from plural nodes, and one ECHONET MAC address corresponds to 2 or more IP addresses, etc. In this case, a network management massage (ECHONET MAC address duplicated) is broadcasted in the subnet.

(b) The node sending this message is called a sending node.

(c) The message is broadcasted in the subnet using IP multicast addresses allocated to ECHONET.

(d) A relevant node (ECHONET MAC address duplicated node) received this message shall do address determination again after the duplication is identified. Generally, the confirmation of the duplication will be done MAC/IP address resolution request for own ECHONET MAC address. Furthermore, re-doing of address determination by the node confirming the duplication will take a style of re-doing by cols start.

(e) The packet format of network management message (Destination unidentified) is shown in Table 7.22.

The sequence chart is shown below.

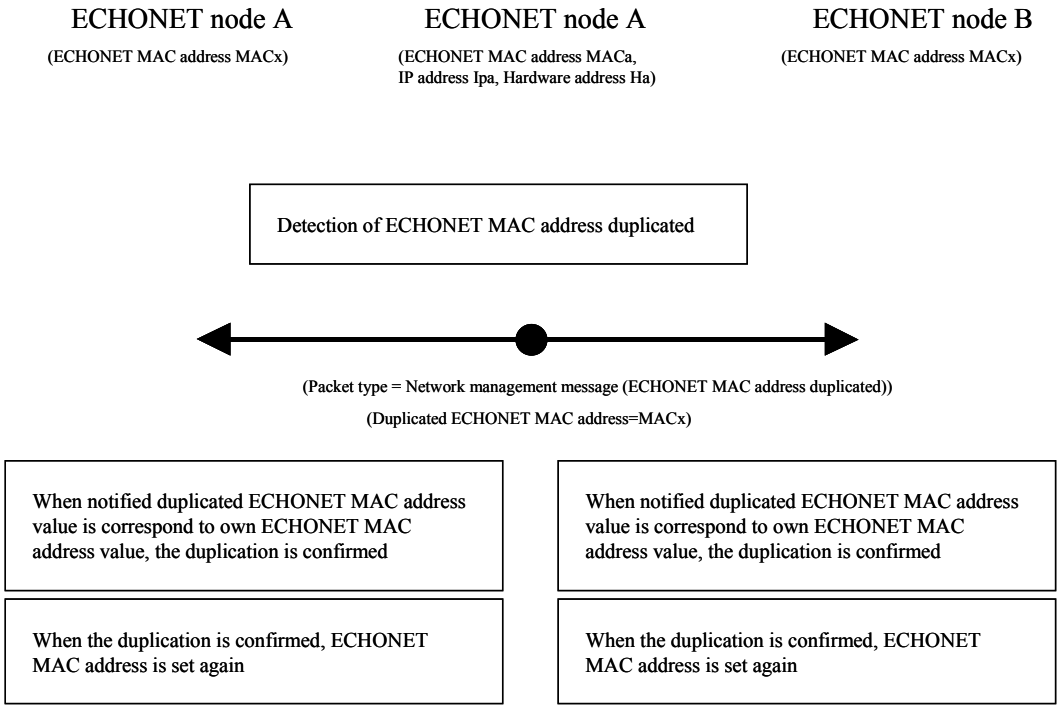


Fig. 7.29 Basic Sequence for detection of ECHONET MAC address duplicated

7.7.4 ECHONET MAC Address Acquisition Startup Sequence

(1) Overview of ECHONET MAC Address Acquisition Booting Sequence

The ECHONET MAC address acquisition booting processing can be divided into the following three processing stages:

・Processing up to establishment of a PANU-NAP/GN connection using BNEP and formation of Piconet (Bluetooth layer);

・ Processing up to acquisition/establishment of IP address (IP layer); and

・ Processing up to acquisition of ECHONET MAC address in IP network (ECHONET/IP layer).

(2) Detailed Description of ECHONET MAC Address Acquisition Booting Sequence

(A) Processing up to Establishment of a PANU-NAP/GN Connection using BNEP and Formation of Piconet (Bluetooth layer)

(i) Overview

This Specification does not specify any connection procedure for processing up to the establishment of a PANU-NAP/GN connection using BNEP and the formation of a Piconet. However, the requirements specified in the Bluetooth Specification "Personal Area Networking Profile" must be satisfied.

(ii) Connection Procedure

The PANU shall select a NAP/GN and connect to it using BNEP to form a Piconet.

A sample processing flow for this stage is given below for reference.

The PANU checks its surroundings using inquiries for Bluetooth devices.

The PANU selects a Bluetooth device from those found and confirms using SDP that it is a NAP/GN.

The PANU connects to the NAP/GN using BNEP.

Steps 1 and 2 above may be omitted when the BD_ADDR of the NAP/GN has already been determined.

(iii) Basic Sequence

Figure 7.27 below shows a sample processing sequence up to the formation of a Piconet. It should be noted that this sequence is not intended to specify an implementation method.

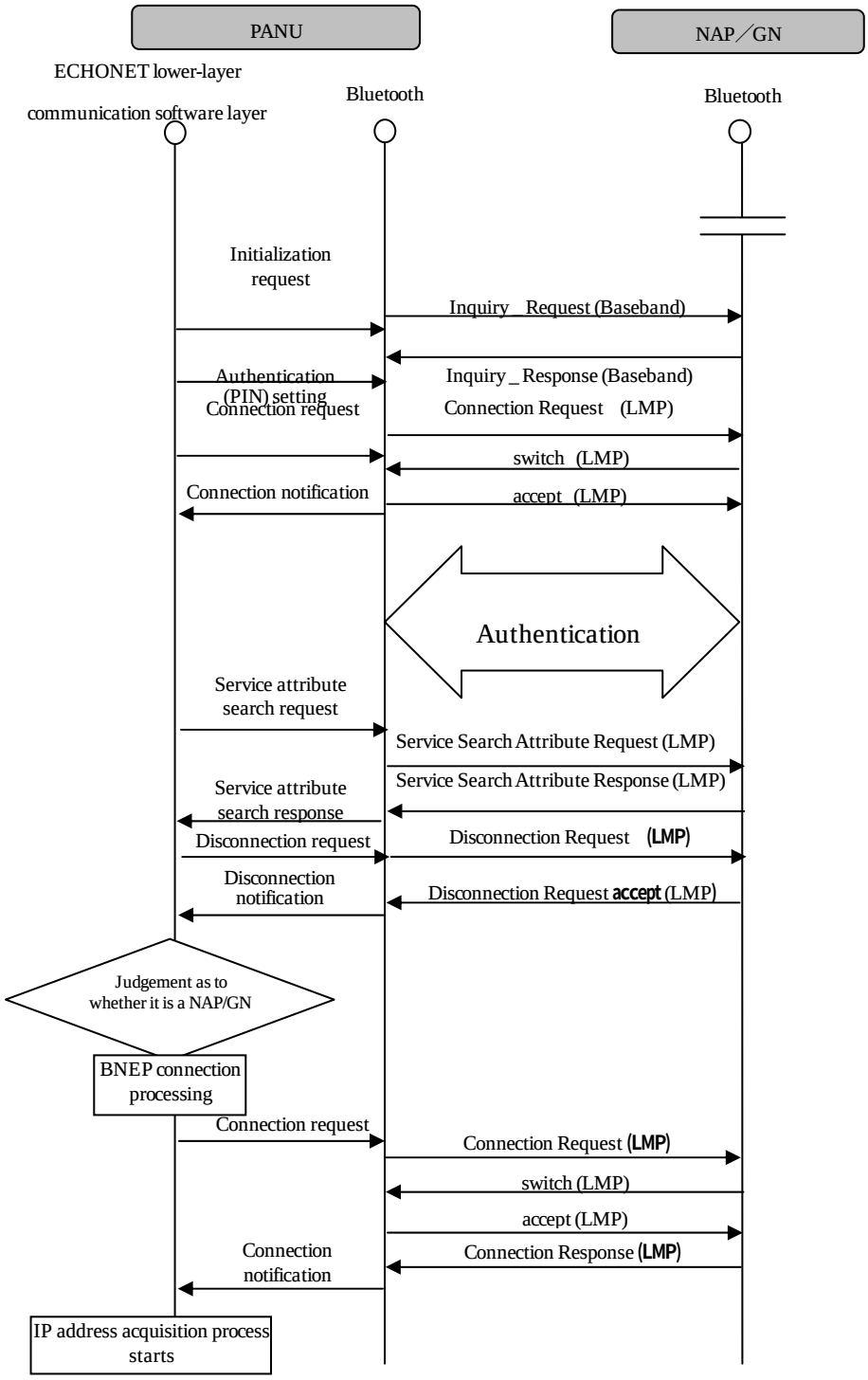


Fig. 7.30 Basic Sequence

(B) Processing up to Acquisition/Establishment of IP Address (IP layer)

For processing up to the acquisition/establishment of an IP address with IPv4, it is recommended that DHCP be used for IP address acquisition. The implementation of a function to acquire addresses as a DHCP client is mandatory.

(C) Processing After Acquisition of IP Address

(i) Overview

All nodes supporting the ECHONET/IP layer shall determine their ECHONET MAC addresses at boot time by performing the sequence specified below. The "Automatic Mode" (A-MODE), "Server Required Mode" (SR-MODE), or "Manual Mode" (M-MODE) can be used to achieve this, depending on the administrator's settings, etc. (This Specification does not specify the configuration of such settings.) However, it is recommended that the Automatic Mode (A-MODE) be used unless another mode has been configured in the settings.

Either the Manual Mode (M-MODE) or the Automatic Mode (A-MODE) must be provided. Provision of the Server Required Mode (SR-MODE) is optional. Mixed use of 1) nodes using the Manual Mode (M-MODE) for booting and 2) nodes using another booting mode should be avoided. This Specification does not specify any functional requirements for such mixed use (e.g., effects of improper settings made in Manual Mode).

Table 7.25 Booting Modes

Booting mode	Explanation
Automatic Mode (A-MODE)	Booting of new nodes using the Address Server Method or Distributed Determination Method. Dynamic ECHONET MAC address acquisition is possible.
Server Required Mode (SR-MODE)	New nodes dynamically acquire ECHONET MAC addresses from the address server. If no MAC address server is found, an error will occur and processing will stop.
Manual Mode (M-MODE)	ECHONET MAC addresses of new nodes are set manually.

An outline of the procedure is given below. For details, see (ii), (iii) and (iv).

1. Booting mode check. If the Manual Mode (M-MODE) is used for booting, the booting node will use the set value as its ECHONET MAC address. Here, the booting node ends this sequence.
2. The booting node sets a provisional ECHONET MAC address.
3. The booting node multicasts a MAC address initialization request packet using the address IPme. When in Server Required Mode (SR-MODE), "1" will be entered in the bit 7 field of the Flag. When in Automatic Mode (A-MODE), "0" will be entered in the bit 7 field of the Flag.

   On receipt of the MAC address initialization request packet, the MAC address server sends a MAC

address server initialization response packet to the booting node. Nodes other than the MAC address server send their MAC address initialization response packets to the booting node only when the bit 7 field of the Flag of the MAC address initialization request packet they received is "0".

4. If the booting node receives the MAC address server initialization response packet, it will use the ECHONET MAC address contained therein as its formal ECHONET MAC address. In this case, the booting node sends a MAC address allocation response packet and ends this sequence.
5. If the booting node does not receive this MAC address server initialization response packet while in Server Required Mode (SR-MODE), the sequence will fail and will be forcefully terminated.
6. The booting node checks all MAC address initialization response packets it received before expiration of the timeout period to determine whether or not there is an ECHONET MAC address in use in the subnet that is the same as the provisional ECHONET MAC address set in Step 2. If no such ECHONET MAC address exists in the subnet, the booting node proceeds to Step 8.
7. If there is an ECHONET MAC address in use in the subnet that is the same as the provisional ECHONET MAC address set in Step 2, the booting node sets a different provisional ECHONET MAC address.
8. The booting node multicasts a MAC address confirmation request packet using the address IPme. Nodes receiving the MAC address confirmation request packet check whether or not their ECHONET MAC addresses are the same as the provisional ECHONET MAC address contained in the packet. A node whose ECHONET MAC address coincides with the provisional ECHONET MAC address sends a MAC address confirmation response packet to the booting node. The booting node repeats Steps 7 and 8 until it no longer receives a MAC address confirmation response packet.

Nodes other than the MAC address server do not send a MAC address server initialization response packet to nodes using the Server Required Mode (SR-MODE) for booting, which means that the initial booting-related packet traffic will be lower when there are nodes using the Server Required Mode (SR-MODE) for booting.

A detailed explanation of the processing sequences for booting nodes, the MAC address server, and operating nodes follows.

(ii) Booting Node

All nodes supporting the ECHONET/IP layer shall determine their ECHONET MAC addresses at boot time with the following sequence:

1. Booting mode check. When in Manual Mode (M-MODE), the booting node will use the set value as its ECHONET MAC address. In this case, the booting node ends this sequence. The booting node proceeds to Step 2 when in Automatic Mode (A-MODE) or Server Required Mode (SR-MODE).
2. The booting node waits until time T10 expires.
3. The booting node sets a provisional ECHONET MAC address. This address shall be determined based on the following algorithm:

(a) When the booting node has in memory the ECHONET MAC address used before the last shutdown,

it will use that ECHONET MAC address as its provisional ECHONET MAC address.

(b) When the booting node does not have in memory the ECHONET MAC address used before the last shutdown, the booting node will use the last 8 bits of the hardware address as its provisional ECHONET MAC address when the booting node is to discard the ECHONET MAC address at the time of initialization (by means of a LowInit/LowInitAll call) or when the booting node is starting up for the first time.

Provision of a function to store the ECHONET MAC address used before the last shutdown is mandatory.

Figure 7.28 below shows the flowchart for determining the provisional ECHONET MAC address to be used.

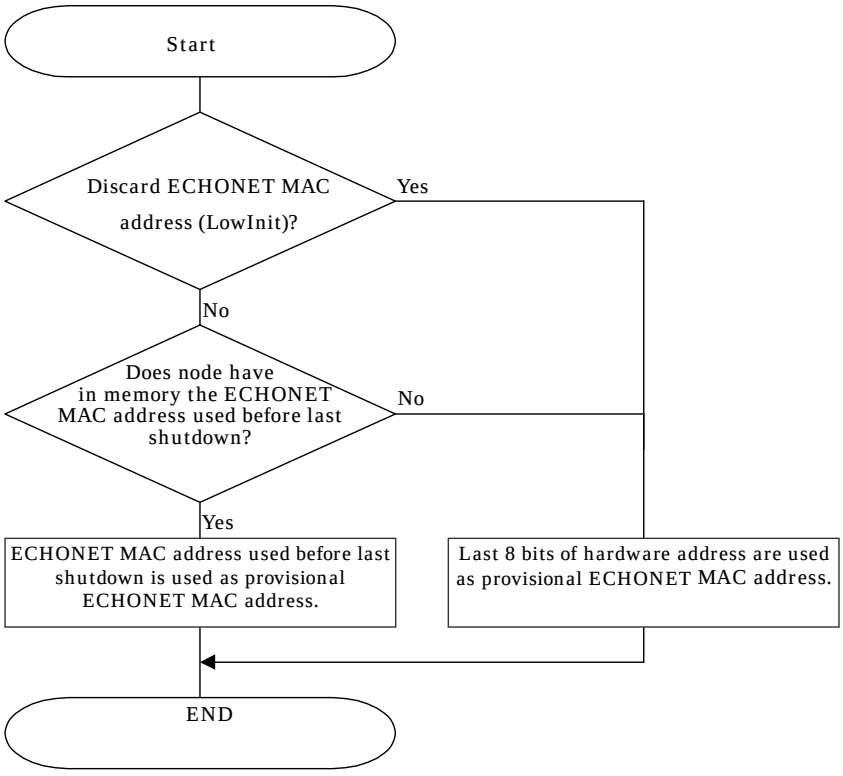


Fig. 7.31 Flowchart for Determining Provisional ECHONET MAC Address to Be Used

4. The booting node multicasts a MAC Address initialization request packet using the address IPme. The packet shall be transmitted twice with an interval of T6. However, the second transmission may be omitted if the booting node received the MAC address server initialization response packet explained below before expiration of period T6.

The format for MAC address initialization request packets is shown in Table 7.11. Set bit 7 of the Flag to "1" for Server Required Mode (SR-MODE) and to "0" for Automatic Mode (A-MODE).

5. If the booting node receives a MAC address server initialization response packet within the timeout period T2 as measured from the transmission of the last MAC address initialization request packet, it will use the ECHONET MAC address (RMAC) contained therein as its ECHONET MAC address.

In this case, the booting node sends a MAC address allocation response packet to the sender of the MAC address server initialization response packet and ends this sequence. If the booting node receives the MAC address server initialization response packet again after determining which ECHONET MAC address to use, it will send the MAC address allocation response packet again. Table 7.14 provides the format for MAC address allocation response packets. The allocated ECHONET MAC address and the MAC address server's ECHONET MAC address are stored in RMAC and SMAC, respectively.

6. If the booting node does not receive this MAC address server initialization response packet by the expiration of timeout time T2 while in Server Required Mode (SR-MODE), the sequence will fail and will be forcefully terminated. When in Automatic Mode (A-MODE), the booting node proceeds to Step 7.
7. The booting node checks all MAC address initialization response packets received by expiration of timeout period T2 for a TMAC address that coincides with the provisional ECHONET MAC address or an ECHONET MAC address represented by "1" in the UsedMAC ("MAC in use") flag which coincides with the provisional ECHONET MAC address. UsedMAC logical sum calculation may be used for implementation (see Fig. 7.29).

If the provisional ECHONET MAC address satisfies any one of the following conditions, the booting node proceeds to Step 8 of this sequence. Otherwise, the booting node skips Step 8 and proceeds to Step 9 of this sequence.

(a) The provisional ECHONET MAC address coincides with a TMAC address (responding node's ECHONET MAC address) contained in a received MAC address initialization response packet (which means that the ECHONET MAC address is being used in the subnet) or the ECHONET MAC address represented by "1"in the UsedMAC ("MAC in use") flag of a received MAC address initialization response packet.

(b) The provisional ECHONET MAC address coincides with a provisional ECHONET MAC address (RMAC) contained in a received MAC address initialization request packet.

(c) The provisional ECHONET MAC address coincides with a provisional ECHONET MAC address (RMAC) contained in a received MAC address confirmation request packet.

Conditions (b) and (c) are necessary to prevent a booting node from using the same ECHONET MAC address that another booting node is attempting to use.

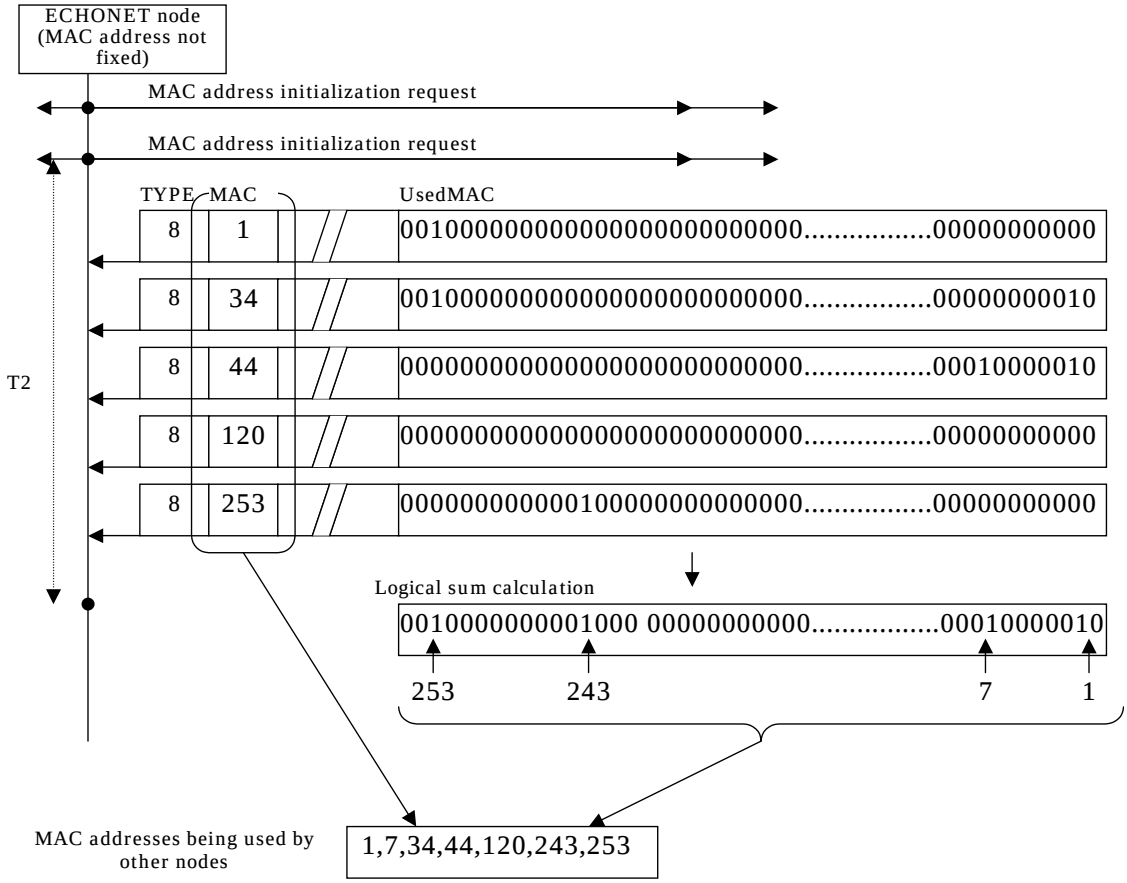


Fig. 7.32 Check for ECHONET MAC Addresses in Use by Other Nodes in Subnet

8. The booting node changes the provisional ECHONET MAC address to a new one using a random number table. The following addresses may not be used as the provisional ECHONET MAC address:
   (a) The responding nodes' ECHONET MAC addresses (TMACs) contained in the MAC address initialization response packets received in this sequence.
   (b) The ECHONET MAC addresses represented by "1" in the UsedMAC ("MAC in use") flags of the MAC address initialization response packets received in this sequence.
   (c) The provisional ECHONET MAC addresses (RMACs) contained in the MAC address initialization request packets received in this sequence (See note below).
   (d) The provisional ECHONET MAC addresses (RMACs) contained in the MAC address confirmation request packets received in this sequence (See note below).
   (e) The provisional ECHONET MAC addresses already used by the booting node in this sequence (See note below).

(Note) There is one exception to conditions (c), (d) and (e). An ECHONET MAC address noted in condition (c) or (d) may be used as the provisional ECHONET MAC address after a period T8 has elapsed since receipt of the packet. This exception is optional and need not be implemented. The exception is provided to prevent a situation in which two or more nodes booting simultaneously attempt

to use the same ECHONET MAC address, making it impossible for any node to use the address in question.

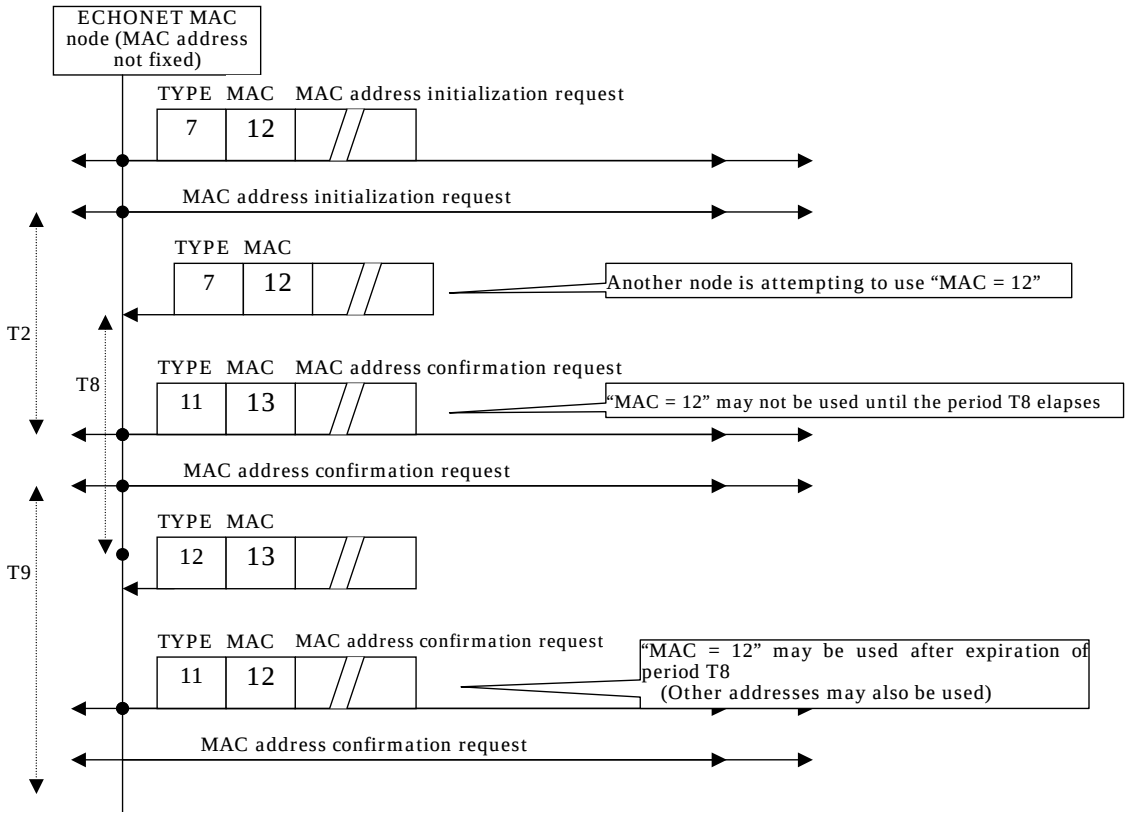


Fig. 7.33 Calculating T8

9. The booting node multicasts a MAC address confirmation request packet using the address IPme. The MAC address confirmation request packet shall be transmitted twice with an interval of T6. However, the second transmission may be omitted if the booting node receives a MAC address confirmation response packet (see below).

Table 7.15 shows the format for MAC address confirmation request packets.

10. If the booting node receives any of the following packets before expiration of timeout period T9 as measured from transmission of the last MAC address confirmation request packet, it will return to Step 8:
    (a) A MAC address confirmation response packet.
    (b) A MAC address initialization request packet whose provisional ECHONET MAC address (RMAC) coincides with the provisional ECHONET MAC address.
    (c) A MAC address confirmation request packet whose provisional ECHONET MAC address (RMAC) coincides with the provisional ECHONET MAC address.
11. If the booting node does not receive any of the packets listed in Step 10, it will use the provisional ECHONET MAC address as its formal ECHONET MAC address. In this case, the booting node ends this sequence.

The ECHONET MAC address information (TMAC), hardware address information (THAddr) and IP address information (TIPAddr) contained in the MAC address initialization response packets, MAC address confirmation response packets, and MAC address server initialization response packets received in this sequence (or after completion of this sequence) may be used as the address relation information referred to in "7.7.3 Basic Communication Sequences." The ECHONET MAC address of the master router (see Part 2 of the Standard) may be obtained from the master flag information (bit 7 of the Flag) of a MAC address initialization response packet or MAC address server initialization response packet.

(iii) MAC Address Server

For the processing sequence for the MAC address server, refer to Section 7.7.5.

(iv) Operating nodes

The operating nodes (i.e., nodes that have completed the sequence described in (ii) of this Section and have an established ECHONET MAC address, except for the address server explained in (iii) of this Section) shall perform the following sequence:

A. If an operating node receives a MAC address initialization request packet and bit 7 of the Flag is "0", it will send a MAC address initialization response packet to the sender after expiration of period T7. This transmission is not necessary when the operating node receives a MAC address initialization request packet containing the same hardware address within period T6.

T7 shall be calculated as follows:

T7 = [Operating node's ECHONET MAC address] x [T1] + [T0]

When bit 7 of the Flag is "1", no node other than the MAC address server is to respond, because it means that the transmission is from a node using the Server Required Mode (SR-MODE) for booting.

Table 7.12 shows the format for MAC address initialization response packets. Figure 7.31 below shows the format for UsedMAC.

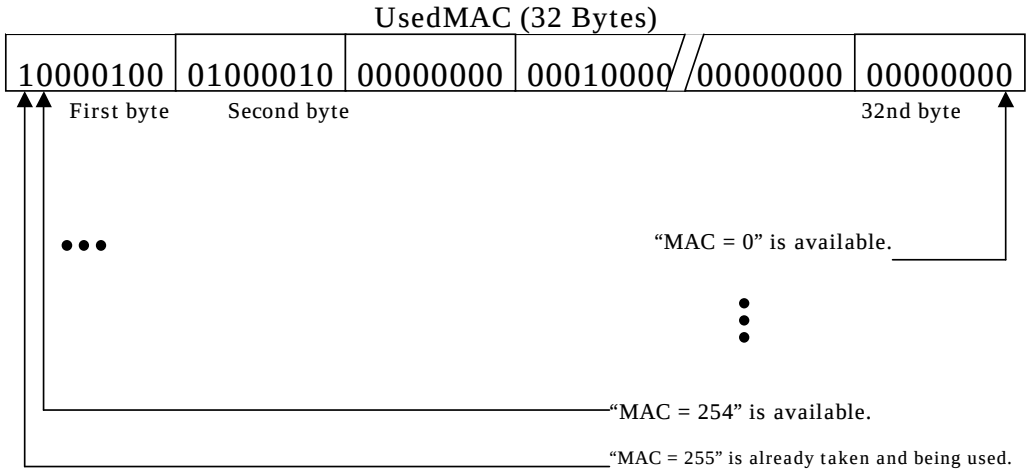


Fig. 7.34 Format for UsedMAC

In UsedMAC, "1" can be entered for bit n (which corresponds to the ECHONET MAC address n) if the bit satisfies the following condition:

* A packet has been received from a node whose ECHONET MAC address is n within the past period T13, except when the hardware address contained in the packet coincides with that contained in the MAC address initialization request packet.

The use of "1" for bit n of UsedMAC prevents the nodes receiving the packet from using ECHONET MAC address n. This allows a device that communicates frequently with a device having a particular ECHONET MAC address to prevent other devices from using that ECHONET MAC address.

In the implementation, a function may be used that 1) records upon receipt of a packet the time of reception and the packet's ECHONET MAC address using the hardware address as the key and 2) checks at the time of request whether or not period T13 has elapsed since reception.

When the power to the operating node has been switched off or shut down within the past period T13, all packets received before the switch-off/shutdown will be ignored.

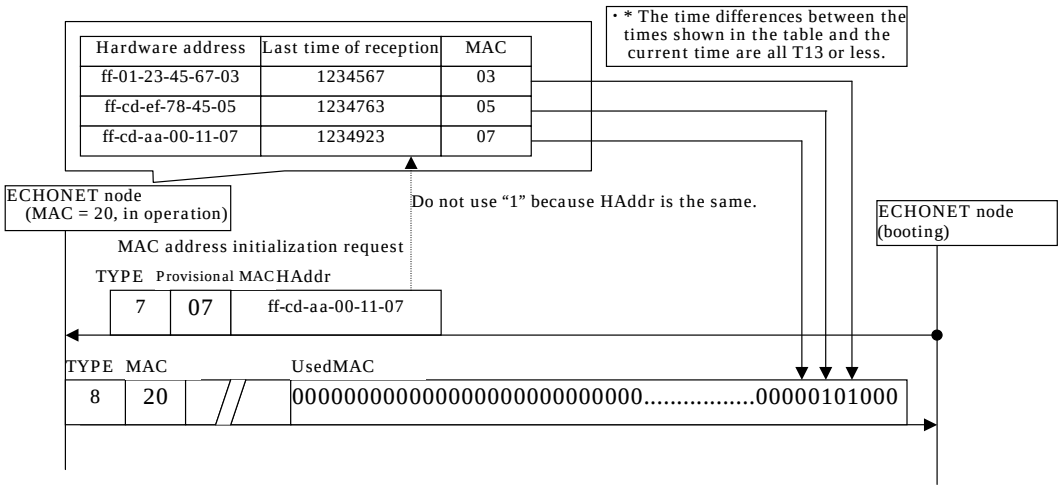


Fig. 7.35 Example of UsedMAC

The implementer must be very careful about using "1" in UsedMAC. Specifically, the implementer must properly compare the hardware addresses, and if they are the same, must never use "1", because it would unnecessarily force the device using that ECHONET MAC address before the last shutdown to change it at boot time.

For a simplified implementation, the implementer may use "0" for all bits of UsedMAC or decide whether to use "0" or "1" only for certain bits and unconditionally enter "0" for the rest.

B. If an operating node receives a MAC address confirmation request packet and the provisional ECHONET MAC address (MAC) contained therein is the same as the operating node's ECHONET MAC address, the operating node will send a MAC address confirmation response packet to the sender.

Table 7.16 shows the format for MAC address confirmation response packets. The data to be stored in UsedMAC are described in the previous paragraph.

(3) Sample Basic Sequences

Sample sequences for the following basic cases are provided below for reference:

* A-MODE booting, MAC addresses not retained (with MAC address server)
* SR-MODE booting, MAC addresses not retained (with MAC address server)
* A-MODE booting, MAC addresses not retained (without MAC address server)
* SR-MODE booting, MAC addresses not retained (without MAC address server)
* A-MODE booting, MAC addresses retained (with MAC address server)
* A-MODE booting, MAC addresses retained (without MAC address server)

(A) A-MODE Booting, MAC Addresses Not Retained (with MAC Address Server)

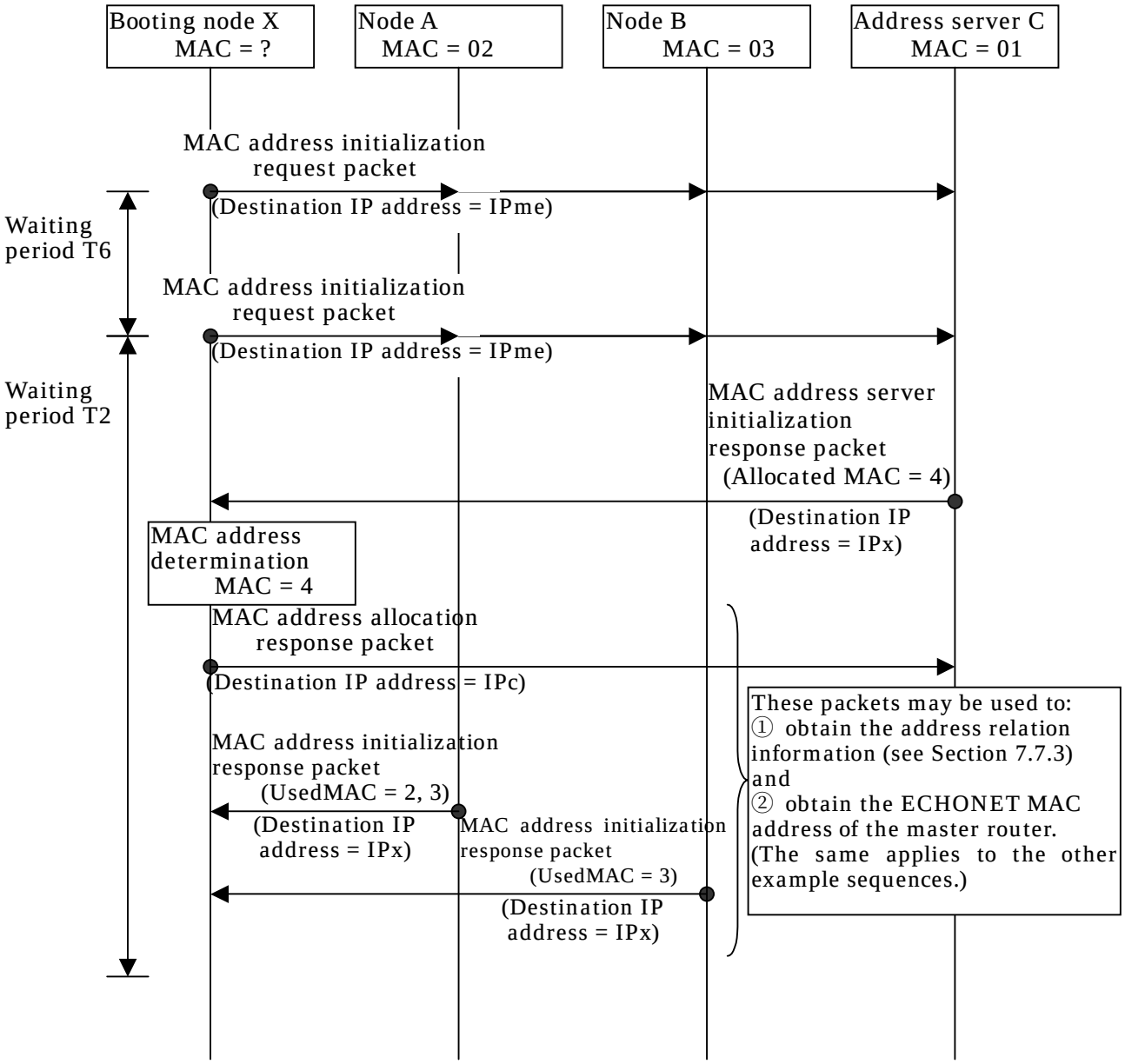


Fig. 7.36 A-MODE Booting, MAC Addresses Not Retained (with MAC Address Server)

(B) SR-MODE Booting, MAC Addresses Not Retained (with MAC Address Server)

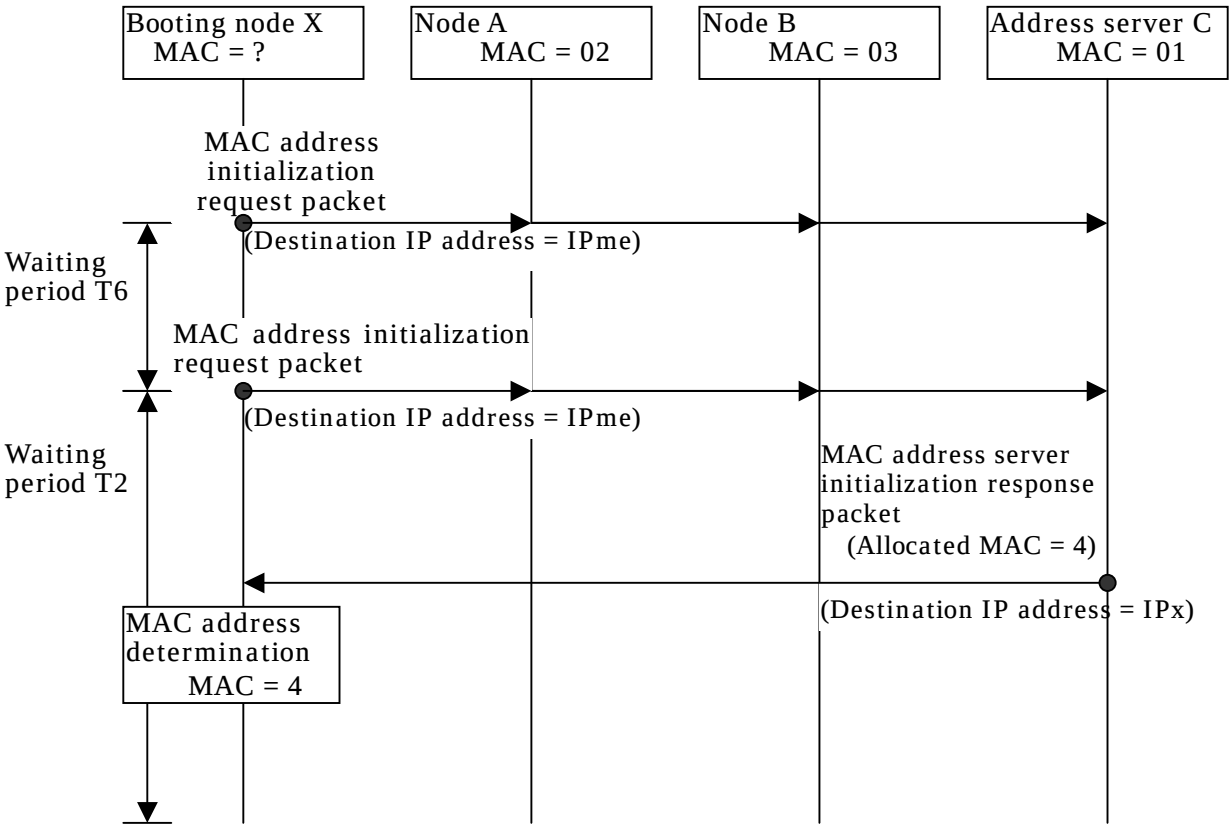


Fig. 7.37 SR-MODE Booting, MAC Addresses Not Retained (with MAC Address Server)

(C) A-MODE Booting, MAC Addresses Not Retained (without MAC Address Server)

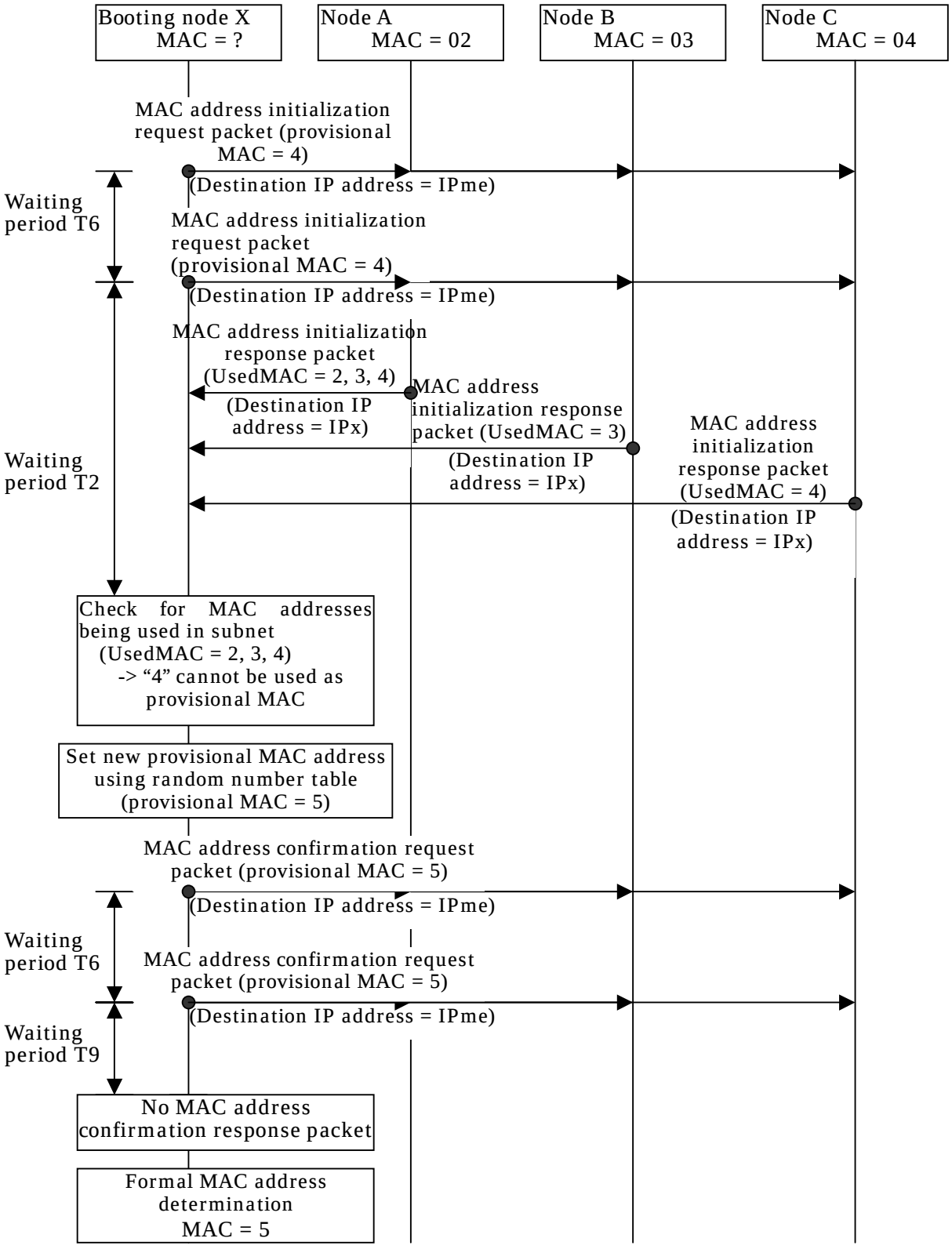


Fig. 7.38 A-MODE Booting, MAC Addresses Not Retained (without MAC Address Server)

(D) SR-MODE Booting, MAC Addresses Not Retained (without MAC Address Server)

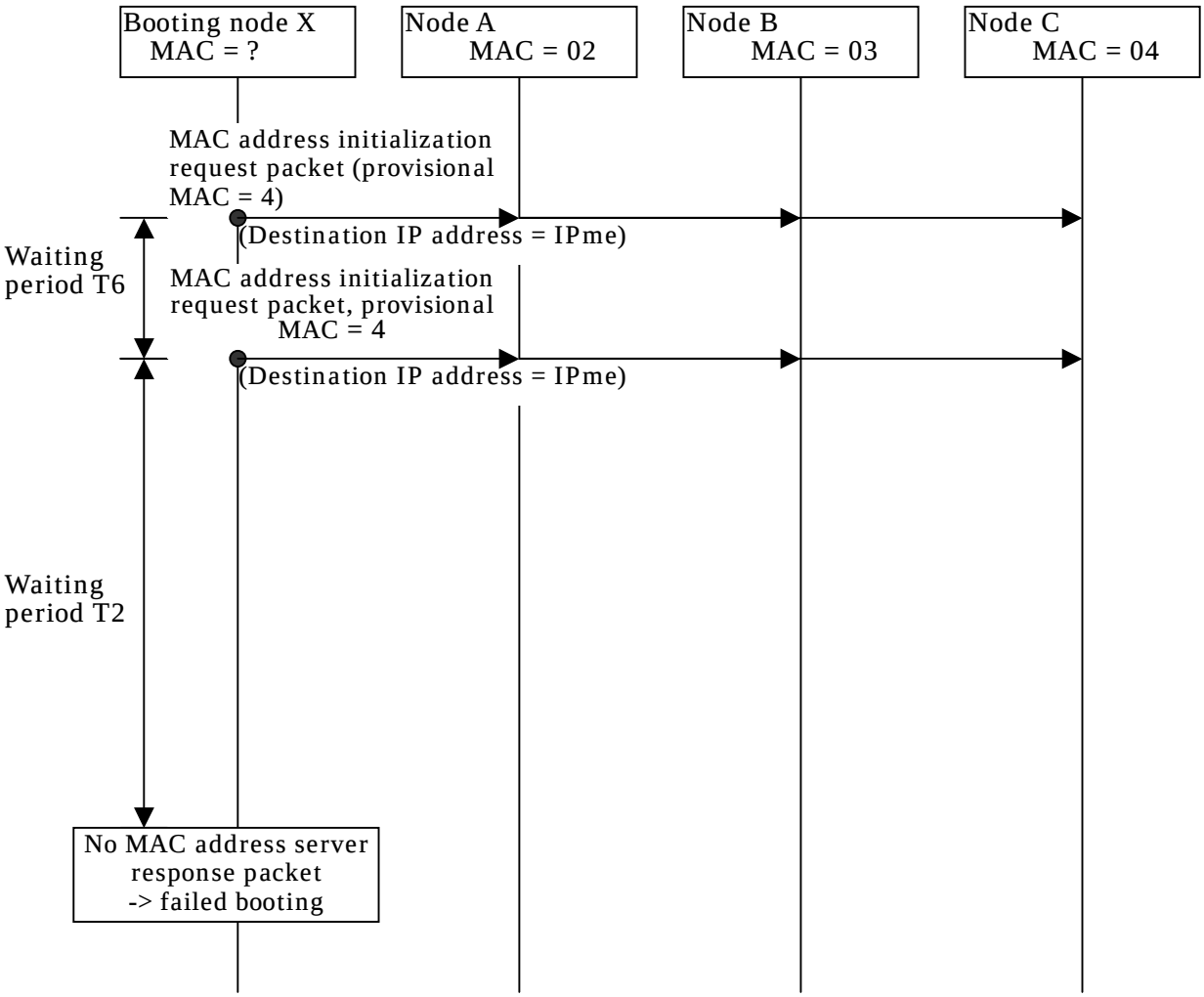


Fig. 7.39 SR-MODE Booting, MAC Addresses Not Retained (without MAC Address Server)

(E) A-MODE Booting, MAC Addresses Retained (with MAC Address Server)

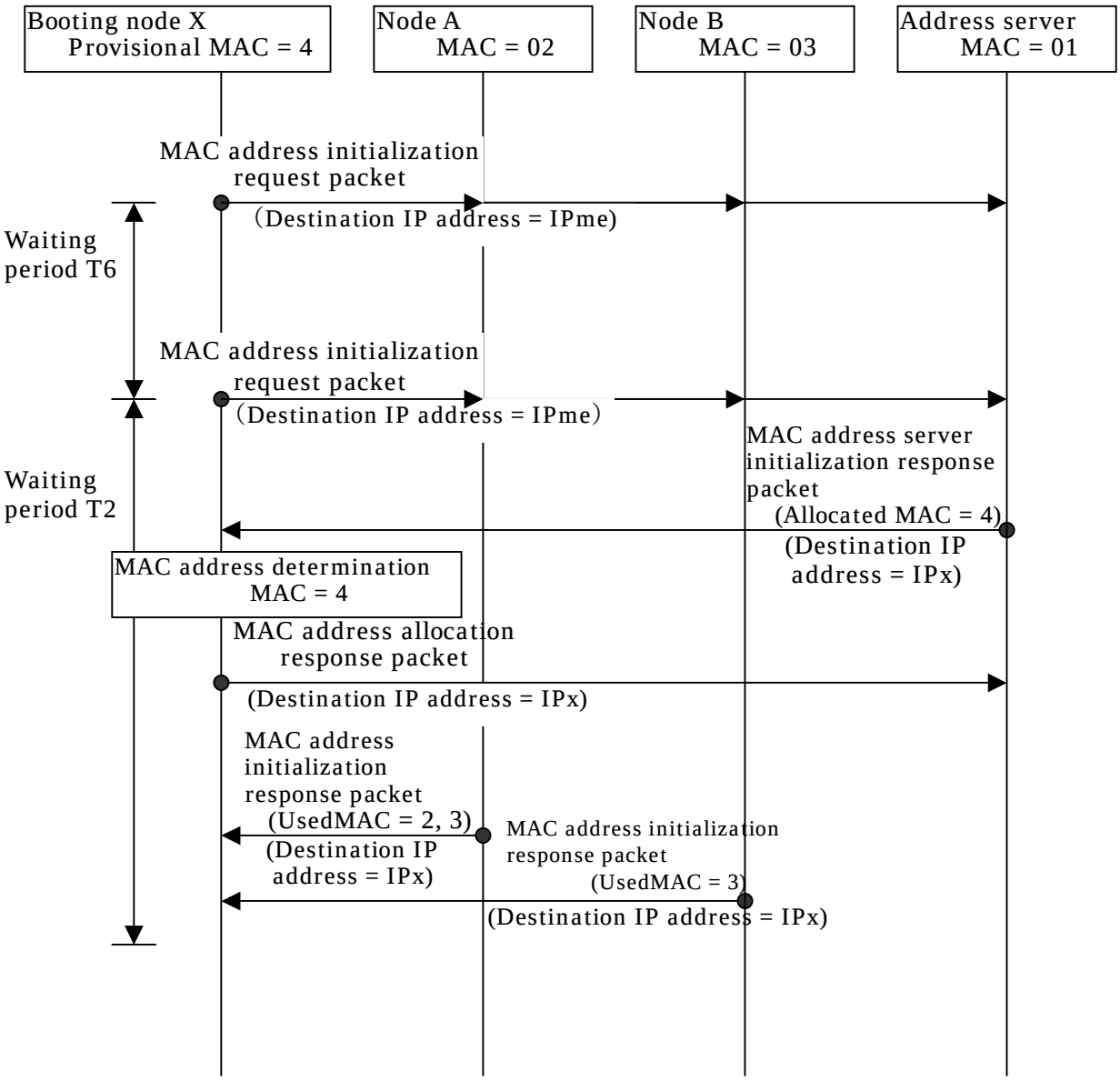


Fig. 7.40 A-MODE Booting, MAC Addresses Retained (with MAC Address Server)

(F) A-MODE Booting, MAC Addresses Retained (without MAC Address Server)

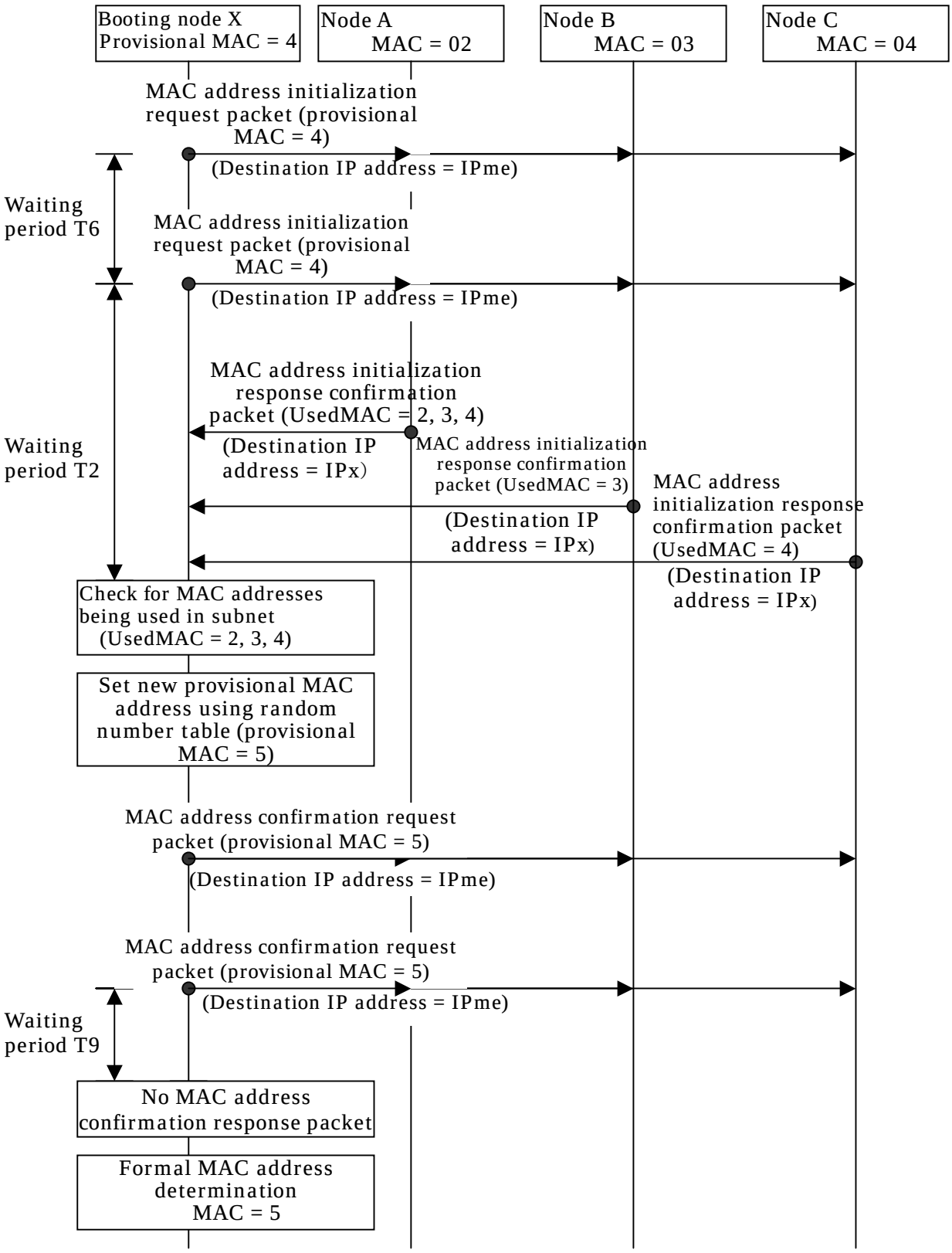


Fig. 7.41 A-MODE Booting, MAC Addresses Retained (without MAC Address Server)

7.7.5 MAC Address Server

(1) Basic Functions of MAC Address Server

All ECHONET MAC addresses in a subnet are controlled by a MAC address server. Each node in a subnet can obtain an ECHONET MAC address from the MAC address server at boot time. The MAC address server must allocate an appropriate ECHONET MAC address to each newly booting node such that the ECHONET MAC address of each device in the subnet is different from that of the other devices.

The MAC address server should allocate the same address to devices of the same type (i.e., nodes having the same hardware address). It is usually necessary for the MAC address server to remember which ECHONET MAC addresses it has allocated to which hardware addresses.

A subnet cannot have more than one MAC address server. When there is no MAC address server in a subnet (or when there is no node seeking to become the MAC address server), any non-MAC address server node (general node) in the subnet can become the MAC address server if and when it desires to do so. However, the MAC address server cannot become a general node.

(2) Processing Sequence for MAC Address Server Booting

When there is no MAC address server in a subnet (or when there is no node attempting to become the MAC address server), any non-MAC address server node (general node) in the subnet can become the MAC address server if and when it desires to do so.

To become the MAC address server, a general node must perform the following sequence:

1. The general node sends a MAC address server detection request packet twice using the address IPme at an interval of T6. The second transmission may be omitted if it receives a MAC address server detection response packet or a MAC address server notification packet (see below).

Table 7.19 shows the format for MAC address server detection request packets.

2. If the general node receives a MAC address server detection response packet or MAC address server notification packet before the expiration of timeout period T5, this sequence will fail and will be forcefully terminated.

3. If the general node receives a MAC address server detection request packet, it will repeat this sequence from Step 1 after a waiting period of T12 or greater. Processing in Step 2 will continue even during the waiting period.

4. The MAC address server begins operation.

5. A MAC address server notification packet is multicast twice or more using the address IPme at intervals of T6.

Table 7.20 shows the format for MAC address server notification packets.

The waiting period T12 is a randomly determined period before the node may redo this sequence when it is determined that another node is attempting to start this sequence simultaneously (see Table 7.25). This waiting period prevent two or more nodes from becoming MAC address servers. Normally this waiting period enables the other node to become the MAC address server, in which case the sequence fails and is

forcefully terminated.

When there is an operating node when the ECHONET MAC address server is booted, the ECHONET MAC address server should not allocate the ECHONET MAC address being used by the node to another device. Therefore, the ECHONET MAC address server should obtain hardware addresses and ECHONET MAC addresses of nodes located in the subnet using ECHONET address all-node request packets, etc.

(3) Processing by Operating MAC Address Server

An operating MAC address server must perform the following:

A. When the operating MAC address server receives a MAC address server detection request packet:

The MAC address server will send a MAC address server detection response packet to the sender.

Table 7.21 shows the format for MAC address server detection response packets.

B. When the operating MAC address receives a MAC address initialization request packet:

The MAC address server will determine the ECHONET MAC address to allocate to the sender and send a MAC address server initialization response packet containing the address to the sender. The ECHONET MAC address allocated to the sender shall be determined as follows:

(a) If the MAC address server has responded in the past using this sequence to the hardware type (RHType), hardware address length (RHLen) and hardware address (RHAddr) contained in the MAC address initialization request packet, the ECHONET MAC address allocated to the sender in that response shall be used.

(b)In all other cases except (a), an ECHONET MAC address not being used in the subnet will be allocated to the sender. The following must not be used: an ECHONET MAC address allocated in the past using this sequence, an ECHONET MAC address determined to be in use in the subnet via received packets, an ECHONET MAC address reserved by the administrator, etc., and the MAC address server's ECHONET MAC address. If the provisional ECHONET MAC address contained in the MAC address initialization request packet satisfies this condition, the provisional ECHONET MAC address should be allocated as the ECHONET MAC address.

Depending on the implementation, a function may be used that checks the received packets and allows ECHONET MAC addresses used prior to a certain point in the past to be used as addresses not currently being used in the subnet.

Table 7.13 shows the format for MAC address server initialization response packets.

If the MAC address server does not receive a MAC address allocation response packet before the expiration of timeout period T11 as measured from the transmission of the MAC address server

initialization response packet, the MAC address server will send the MAC address server initialization response packet again (up to a total of 3 times).

It is usually necessary for the MAC address server to remember which MAC addresses it has allocated to which hardware addresses in order to perform (a). Table 7.24 below shows an example of a table for storing the allocated MAC addresses for reference for the implementer.

Table 7.26 Sample Table for Storing Allocated MAC Addresses

Hardware type	Hardware address	Allocation time	Allocated MAC
1	ff-01-23-45-67-03	1232567	03
1	ff-cd-ef-78-45-05	1231763	05
1	ff-cd-aa-00-11-07	1233923	07

When the MAC address server receives a new allocation request (i.e., a request from a node with a hardware address not included in the table), it should allocate an ECHONET MAC address it has not allocated in the past. However, when the MAC address server has used all the allocatable addresses, it must allocate an appropriate address using an allocation method suitable for the implementation. This can be achieved by:

* A function to use the oldest ECHONET MAC address the MAC address server has allocated in the past; or
* A function that stores all incoming packets, checks all packets received, and allows ECHONET MAC addresses allocated to the hardware addresses of nodes that have not performed communication for a certain period of time to be allocated to new requesting nodes.

Whatever method is chosen should minimize or eliminate the possibility of an address overlap.

(4) Basic Sequences

Sample MAC address server booting sequences for the following basic cases are given below for reference:

* Booting of a single MAC address server
* Near-simultaneous booting of two or more MAC address servers

(A) Booting of a Single MAC Address Server

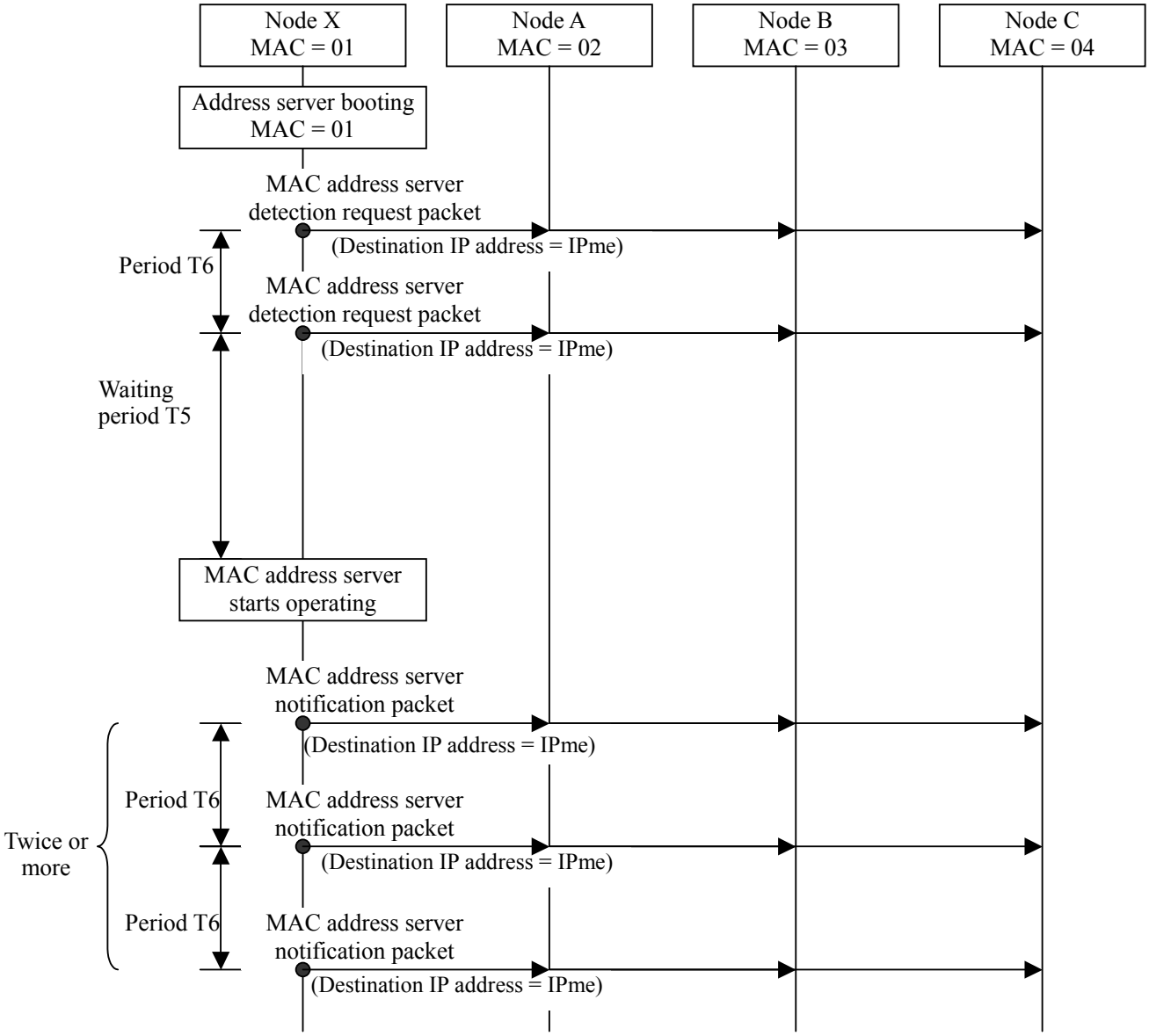


Fig. 7.42 Booting of a Single MAC Address Server

(B) Near-Simultaneous Booting of Two or More MAC Address Servers

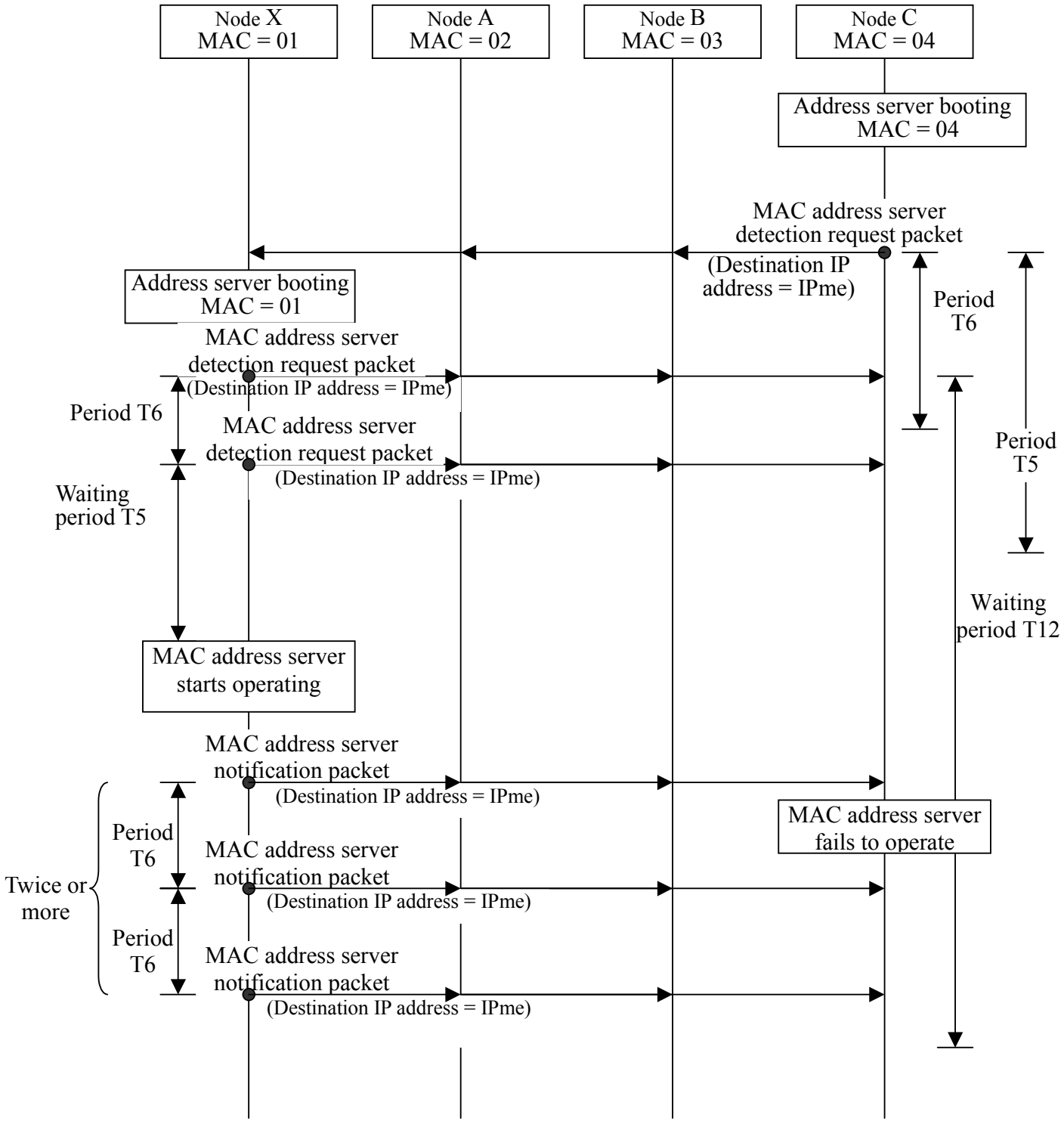


Fig. 7.43 Near-Simultaneous Booting of Two or More MAC Address Servers

7.7.6 Time period parameters

Table 7.25 below shows definitions of and recommended values for the time period parameters used in the Bluetooth Communication Protocol

Table 7.27 Time Period Parameters Used in the Bluetooth Communication Protocol

Parameter	Reference	Value	Type (mandatory or recommended)	Definition
T0	7.7.4(2)(C)(iv)	Within 50 msec	recommended	Waiting period before MAC address initialization response packet may be sent after reception of MAC address initialization request packet when ECHONET MAC address is "0".
T1	7.7.4(2)(C)(iv)	0	mandatory	Incremental unit of waiting period that is multiplied by ECHONET MAC address value and added to T0 to calculate T7.
T2	7.7.4(2)(C)(ii)	3.0sec	mandatory	Interval between multicasting of a MAC address initialization request packet and reception of corresponding MAC address initialization response packet.
T3	7.7.3(1)	3.0sec	mandatory	Timeout period for reception of MAC/IP address resolution response packet after transmission of MAC/IP address resolution request packet.
T4	7.7.3(2)	3.0sec	mandatory	Timeout period for reception of IP/MAC address resolution response packet after transmission of IP/MAC address resolution request packet.
Ditto	7.7.3(3)	3.0sec	mandatory	Timeout period for reception of hardware/MAC address resolution response packet after transmission of hardware/MAC address resolution request packet.
T5	7.7.5(2)	3.0sec	mandatory	Timeout period for reception of MAC address server detection response packet after transmission of MAC address server detection request packet.
T6	7.7.4(2)(C)(ii)	Within 100 msec	recommended	MAC address initialization request packet transmission interval.
Ditto	7.7.4(2)(C)(ii)	Ditto		MAC address confirmation request packet transmission interval.
Ditto	7.7.5(2)	Ditto		MAC address server detection request packet transmission interval.
Ditto	7.7.5(2)	Ditto		MAC address server notification packet transmission interval.
T7	7.7.4(2)(C)(iv)	-	-	Waiting period before MAC address initialization response packet may be sent after reception of MAC address initialization request packet. T7 = [ECHONET MAC address value] x [T1] + [T0]
T8	7.7.4(2)(C)(ii)	24hour	recommended	Waiting period before ECHONET MAC address contained in received MAC address initialization request packet may be reused. Waiting period before ECHONET MAC address contained in received MAC address confirmation request packet may be reused.
T9	7.7.4(2)(C)(ii)	3.0sec	mandatory	Timeout period for reception of MAC address confirmation response packet, MAC Address Initialization request packet (when RMAC is the same as the provisional self ECHONET MAC address) or packet (when RMAC is the same as the provisional self ECHONET MAC address) after transmission of MAC address confirmation request packet.
T10	7.7.4(2)(C)(ii)	0~100msec	recommended	Waiting period before initialization sequence may be performed.
T11	7.7.5(3)	200msec	mandatory	Timeout period for reception of MAC address allocation response packet after transmission of MAC address initialization response packet by MAC address server
T12	7.7.5(2)	20sec	recommended	Waiting period before MAC address server detection request packet may be resent after reception of MAC address server detection request packet.
T13	7.7.4(2)(C)(iv)	24hour	recommended	Waiting period before "1" may be entered in UsedMAC field after reception of packet.

T14	7.7.3	3.0sec	recommended	Time period for waiting for MAC address entire nodes responding packet after MAC address entire nodes requesting packet

7.7.7 Bluetooth Interface

Table 7.26 below shows the relationship between the individual lower-layer communication interface and the Bluetooth Interface for the implementer's reference.

Table 7.28 Example of Relationship Between Individual Lower-Layer Communication Interface and Bluetooth Interface

No.	Individual lower-layer communication interface	Sequence	Bluetooth Interface
1	Request for lower-layer communication software type (LowGetDevID)	* Returns lower-layer communication software ID held by lower-layer communication software.	None
2	Request fot initialization (LowInit)	* Initializes lower-layer communication software. * Discards ECHONET MAC address. * Performs ECHONET MAC address acquisition booting sequence.	* Initialization of Bluetooth devices and Bluetooth protocol stack * Roll switch setting (to cause PANU to act as slave) * Authentication setting (to enable authentication of connection requests from terminal on other end) * PIN code setting (setting of PIN code used for authentication) * BNEP connection
3	Request for operation start (LowRequestRun)	* Enables ECHONET message transmission and reception processing	* BNEP connection (when PANU and NAP/GN are not connected by BNEP connection)
4	Fault notice (LowSetTrouble)	* Does not perform reception processing, or discards message(s). * Returns error message on receipt of any message transmission request from ECHONET Middleware.	None
5	Warm start request (LowStart)	* Initializes lower-layer communication software with MAC address retained. * Performs ECHONET MAC address acquisition booting sequence.	* Initialization of Bluetooth devices and Bluetooth protocol stack * Roll switch setting to cause PANU to act as slave * Authentication setting (to enable authentication of connection requests from terminal on other end) * PIN code setting (setting of PIN code used for authentication) * BNEP connection
6	Request for suspension (LowSuspend)	* Temporarily stops ECHONET message transmission and reception processing until arrival of request to resume operation.	* Park Mode setting (to switch from Active Mode to Park Mode)
7	Request for operation restart (LowWakeUp)	* Performs ECHONET message transmission and reception processing.	* Active Mode setting (to switch from Park Mode to Active Mode)
8	Profile acquisition (LowGetProData)	* Returns lower-layer medium type information and ECHONET MAC address.	None
9	Status acquisition (LowGetStatus)	* Returns transition status information held by lower-layer communication software.	None
10	Request for data transmission (LowSendData)	* Sends ECHONET messages to specified destination ECHONET MAC address(es).	None
11	Transmission result acquisition (LowGetSendResult)	* Returns transmission status information ("transmission in progress," "transmission completed," "transmission aborted due to error," "transmission being cancelled," etc.)	None
12	Request for transmission stop	* Cancels transmission.	None

	(LowSendCansel)		
13	Request for recieved data (LowReceive)	* Returns received message and source ECHONET MAC address.	None
14	Address information acquisition (LowGetAddress)	* Returns self ECHONET MAC address held by lower-layer communication software.	None
15	Address information setting (LowSetAddress)	* Specify self ECHONET MAC address to lower-layer communication software.	None
16	Request for physical address translation (LowReqToMac)	* Returns MAC address corresponding to Node ID on which translation is requested.	None
17	Request for Node ID translation (LowReqToID)	* Returns Node ID corresponding to ECHONET MAC address on which translation is requested.	None
18	Request for broadcast destination acquisition (LowReqBcastID)	* Returns information on broadcast recipients (broadcast addresses, Node IDs, number of Node IDs, etc.)	None
19	Request for full initialization (LowInitAll)	* Initializes lower-layer communication software. * Discards ECHONET MAC address. * Discards IP address and obtains new one. * Performs ECHONET MAC address acquisition booting sequence.	* Initialization of Bluetooth devices and Bluetooth protocol stack * Roll switch setting to cause PANU to act as a slave * Authentication setting to enable authentication of connection requests from terminal on other end * PIN code setting (setting of PIN code used for authentication) * BNEP connection
20	Request for communication stop (LowStop)	* Stops lower-layer communication software's ECHONET message transmission and reception processing.	None
21	Request for full stop (LowHalt)	*Stops lower-layer communication software's ECHONET message transmission and reception processing, IP communication processing, and Bluetooth communication processing.	* BNEP disconnection (when PANU and NAP/GN are connected by BNEP connection)
22	Stop notification reception (LowReceiveStop)	* Gives stop notification to lower-layer communication software.	None
23	Request for address table data of lower-layer communication software (LowGetAddressTableData)	* Returns address table data held by lower-layer communication software.	None
24	Lower-layer communication software address table request (LowGetAddressTableData)	* Obtains lower-layer address table data held to lower-layer communication software.	None
25	Master router notification (LowSetMasterRouterFlag)	* Specifies master router to lower-layer communication software.	None
26	Request for hardware address data (LowGetHardwareAddress)	* Returns hardware address data held to lower-layer communication software.	None

7.8 Basic Sequence (Software Internal Status Transition Specifications)

7.8.1 Introduction

This Section outlines the sequence for each of the seven internal statuses of lower-layer communication software for the Bluetooth Protocol. The seven statuses are as follows:

① Stop
② Cold Start
③ Warm Start
④ Communication Suspension
⑤ Normal Operation
⑥ Error Stop
⑦ Temporary Stop

Figure 7.42 depicts how the lower-layer communication software for the Bluetooth Protocol transitions from one status to another.
Note that the figure shows transitions for cases in which ECHONET uses Bluetooth® exclusively. Different transitions may occur when ECHONET does not use Bluetooth® exclusively.

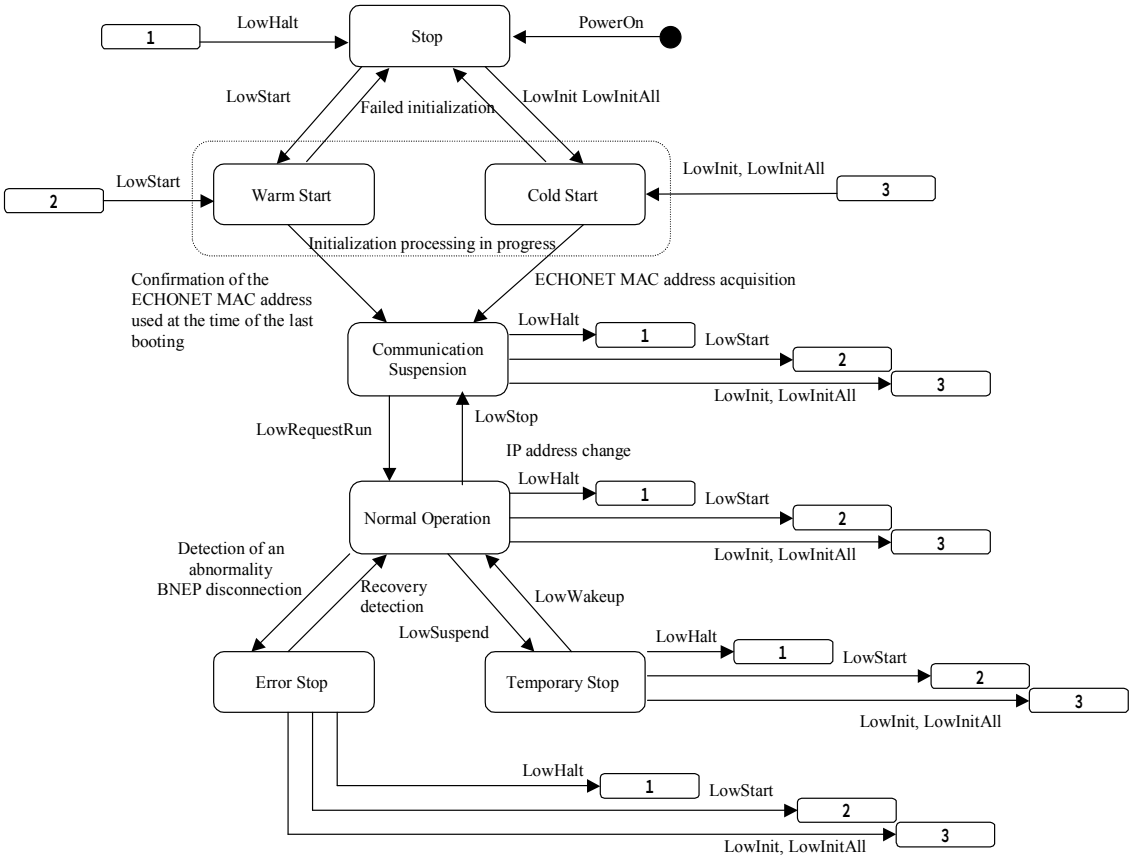


Fig. 7.44 Internal State Transitions in Lower-Layer Communication Software for Bluetooth Protocol

7.8.2 Stop status

In the Stop status, the lower-layer communication software does not perform any lower-layer communication. The software enters this status immediately after power is turned on.

An outline of the processing performed immediately after a status change into this status and the individual lower-layer communication interface services that can be handled by the software in this status follows.

(1) Trigger and Response

The software will transition into this status immediately after power is turned on or when it receives a request to stop operation (LowHalt). The software then waits to receive an individual lower-layer communication interface service request.

(2) Status Acquisition Service (LowGetStatus)

Returns LOW_STS_STOP as the status.

(3) Lower-layer communication software type acquisition service (LowGetDevID)
Returns lower-layer communication software type.

(4) Triggers Causing Transition out of Stop status
* Transition into Cold Start Status:
Initialization request service (LowInit, LowInitAll)
* Transition into Warm Start status:
Warm start request service (LowStart)

7.8.3 Cold Start status

In the Cold Start status, the lower-layer communication software is initialized. When used with the Bluetooth Protocol, the lower-layer communication software in this status performs the MAC address acquisition booting sequence shown in 7.7.4 and waits for a request for one of the individual lower-layer communication interface service request types described below.

(1) Trigger and Response
When an initialization request (LowInit, LowInitAll) is received, the following sequence occurs (see 7.7.4 for details):
Establishment of connection between PANU and NAP/GN using BNEP to form Piconet (Bluetooth layer);
IP address acquisition/determination (IP layer); and
MAC address acquisition on IP network (ECHONET/IP layer). For details, see 7.7.4.

(2) Status Acquisition Service (LowGetStatus)
Returns LOW_STS_INI as the status.

(3) Lower-layer communication software type acquisition service (LowGetDevID)
Returns lower-layer communication software type.

(4) Triggers Causing Transition out of Cold Start Status
* Transition into Communication Suspension Status:
Completion of MAC address acquisition booting processing
* Transition into Stop Status:
Failure to complete MAC address acquisition booting processing

7.8.4 Warm Start status

In the Warm Start status, the lower-layer communication software is initialized without ECHONET MAC address reacquisition. The software then waits for a request for one of the individual lower-layer communication interface service request types described below. For details, see 7.7.4.

(1) Trigger and Response

When a warm start request (LowStart) is received, the following sequence occurs:

① Establishment of connection between PANU and NAP/GN using BNEP to form Piconet (Bluetooth layer);

② Confirmation of whether or not IP address used at last boot and now stored in memory can be used for booting (IP layer);

③ Confirmation of whether or not the MAC address used at last boot and now stored in memory can be used for booting (ECHONET/IP layer); and

④ Booting using IP and MAC addresses described in ② and ③ above.

(2) Status Acquisition Service (LowGetStatus)

Returns LOW_STS_RST as the status.

(3)Lower-layer communication software type acquisition service (LowGetDevID)

Returns lower-layer communication software type.

(4)Triggers Causing Transition out of Warm Start Status

* Transition into Communication Suspension Status:
  Completion of booting processing
* Transition into Stop Status:
  Failure to complete booting processing

7.8.5 Communication Suspension status

In the Communication Suspension status, the initialized lower-layer communication software waits for a request to start operation from the Communication Middleware. An outline of the processing performed immediately after a status change into this status and the individual lower-layer communication interface services that can be handled by the software in this status follows:

(1) Trigger and Response

Waits to receive individual lower-layer communication interface service request.

(2) Status Acquisition Service (LowGetStatus)

Returns LOW_STS_CSTOP as the status.

(3) Lower-layer communication software type acquisition service (LowGetDevID)

Returns lower-layer communication software type.

(4) Physical Address Acquisition Service (LowGetMacAddress)

Returns MAC address.

(5) Profile Data Acquisition Service (LowGetProData)
Returns profile data.

(6) Triggers Causing Transition out of Communication Suspension Status
* Transition into Normal Operation Status:
Operation start service (LowRequestRun)
* Transition into Stop Status:
Operation stop service (LowHalt)
* Transition into Cold Start Status:
Initialization request service (LowInit, LowInitAll)
* Transition into Warm Start Status:
Warm start request service (LowStart)

7.8.6 Normal Operation status

In the Normal Operation status, the lower-layer communication software sends messages to and receives messages from the transmission medium. An outline of the processing performed immediately after a status change into this status and the individual lower-layer communication interface services that can be handled by the software in this status follows.

(1) Trigger and Response
Waits to receive individual lower-layer communication interface service request.

(2) Status Acquisition Service (LowGetStatus)
Returns LOW_STS_RUN as the status.

(3) Lower-layer communication software type acquisition service (LowGetDevID)
Returns lower-layer communication software type.

(4) Physical Address Acquisition Service (LowGetMacAddress)
Returns MAC address.

(5) Profile Data Acquisition Service (LowGetProData)
Returns profile data.

(6) Message Transmission Service (LowSendData)
Converts received protocol difference absorption processing section messages into lower-layer communication software messages and outputs them to the transmission medium.

(7) Message Reception Service (LowRecvData)

Converts lower-layer communication software messages received from transmission medium into protocol difference absorption processing section messages and outputs them to the Protocol Difference Absorption Processing section.

(8) Triggers Causing Transition out of Normal Operation Status

* Transition into Stop Status:
  - Operation stop service (LowHalt)
  or
  - IP address change at IP layer
* Transition into Communication Suspension Status:
  Stop service (LowStop)
* Transition into Cold Start Status:
  Initialization request service (LowInit, LowInitAll)
* Transition into Warm Start Status:
  Warm start request service (LowStart)
* Transition into Error Stop Status:
  - Detection of abnormality by lower-layer communication medium
  or
  - Disconnection of BNEP connection at Bluetooth layer
* Transition into Temporary Stop Status:

Lower-layer communication section stop service (LowSuspend)

7.8.7 Error Stop status

In the Error Stop status, the lower-layer communication software's operation is stopped because of an error. An outline of the processing performed immediately after a status change into this status and the individual lower-layer communication interface services that can be handled by the software in this status follows.

(1) Trigger and Response

Performs processing to deal with error.

(2) Status Acquisition Service (LowGetStatus)

Returns LOW_STS_ESTOP as the status.

(3) Lower-layer communication software type acquisition service (LowGetDevID)

Returns lower-layer communication software type.

(4)Triggers Causing Transition out of Error Stop Status

* Transition into Stop Status:
  Operation stop service (LowHalt)
* Transition into Normal Operation Status:
  Removal of cause of error
* Transition into Cold Start Status:
  Initialization request service (LowInit, LowInitAll)
* Transition into Warm Start Status:
  Warm start request service (LowStart)

7.8.8 Temporary Stop status

In the Temporary Stop status, the lower-layer communication software's operation is temporarily stopped as the result of an instruction from the Communication Middleware. An outline of the processing performed immediately after a status change into this status and the individual lower-layer communication interface services that can be handled by the software in this status follows.

(1) Trigger and Response

Stops lower-layer communication software's operation.

(2) Status Acquisition Service (LowGetStatus)

Returns LOW_STS_SPD as the status.

(3) Lower-layer communication software type acquisition service (LowGetDevID)

Returns lower-layer communication software type.

(4) Triggers Causing Transition out of Temporary Stop Status

* Transition into Normal Operation Status:
  Operation resumption service (LowWakeUp)
* Transition into Stop Status:
  Operation stop service (LowHalt)
* Transition into Cold Start Status:
  Initialization request service (LowInit, LowInitAll)
* Transition into Warm Start Status:
  Warm start request service (LowStart)

7.9 Accommodation Requirements, etc.

7.9.1 Accommodation requirements for NAP, GN, and PANU

This subsection specifies the accommodation requirements for incorporating the ECHONET lower-layer communication software specifications into a NAP, GN or PANU.

(1) Requirement for NAP Equipment with Bridge

In cases where another communication medium is used as an ECHONET communication medium in addition to the Bluetooth medium for a NAP device with a bridge or bridges (explained in Fig. 7.1) that operates as an ECHONET node (Ver. 3 = Ethernet only), the following requirement shall apply:

Regarding the MAC address and lower-layer software type in the properties defined as the lower-layer communication software profile class, the ECHONET/IP layer will determine at initialization which of its two lower-layer media to use and will use the relevant values permanently. As in the other chapters, this Specification does not specify any requirement with respect to the relationship between the ECHONET/IP layer and the lower-layer medium itself.

(2) Plug & Play Man-Machine Requirements

The Plug & Play Man-Machine Requirements are as follows:

1) Operation Mode Indication

It is recommended that a visual indication of the operation mode be provided in each node for network troubleshooting. This Specification does not specify any requirements regarding the indication method, color, position, and so on, but any visual indicator other than an LED should be legible to the user. Any LED indicator shall be in accordance with the rules shown in Table 7.27 below.

Table 7.29 Rules for LED Indicators

LED indicator	Lit	Blinking	Extinguished
Operation mode	Plug & play operation	Abnormal plug & play setting termination	Other than plug & play operation and abnormal plug & play setting termination
Nodes	Nodes in the process of acquiring an ECHONET MAC address MAC address server node	Nodes whose plug & play settings are abnormal, including MAC address server node with abnormal plug & play settings	All nodes

* "Plug & play operation" signifies ECHONET MAC address establishment processing.

When visual indicator should be lit:

・Nodes in process of acquiring ECHONET MAC address

The indicator should be lit from the time an ECHONET MAC address initialization request is made to ECHONET MAC address confirmation until the acquisition of an ECHONET MAC address. The visual indication may also be lit from the time an IP address acquisition request is made to the time an IP address is acquired.

・MAC address server node

The indicator should be lit from the time an ECHONET MAC address initialization request is made to confirmation that the ECHONET MAC address acquisition process is in progress until the completion of transmission of an ECHONET MAC address. The visual indication may also be lit from the time an IP address acquisition request is made to the time an IP address is acquired.

When visual indicator should blink on and off:

The indicator should blink from the time an abnormal plug & play setting termination occurs (i.e., when an ECHONET MAC address acquisition failure occurs) until the cause is removed. The visual indicator may also blink from the time an IP address acquisition fails until the cause is removed.

2) ECHONET MAC Address Acquisition Method

As mentioned in Subsection 7.7.4, there are three different methods for acquiring an ECHONET MAC address. Any node equipped with the Server Required Mode (SR-MODE) function and/or the Manual Mode (M-MODE) function in addition to the mandatory Automatic Mode (A-MODE) function shall have a function to select and specify one mode before connecting to the network based on network characteristics (e.g., number of nodes, allowable traffic level during booting, whether or not the network has an administrator). This Specification does not cover methods for specifying/indicating the mode to be used, etc.

(3) Interoperability with NAP/GU without ECHONET Functions

Interoperability between a NAP/GN that does not have the ECHONET functions specified in this Specification and a PANU that has such ECHONET functions can be achieved if the NAP/GN meets the packet transfer timeout requirements specified in this Specification and also has ECHONET multicast functions. Any NAP/GN that meets these conditions may be used.

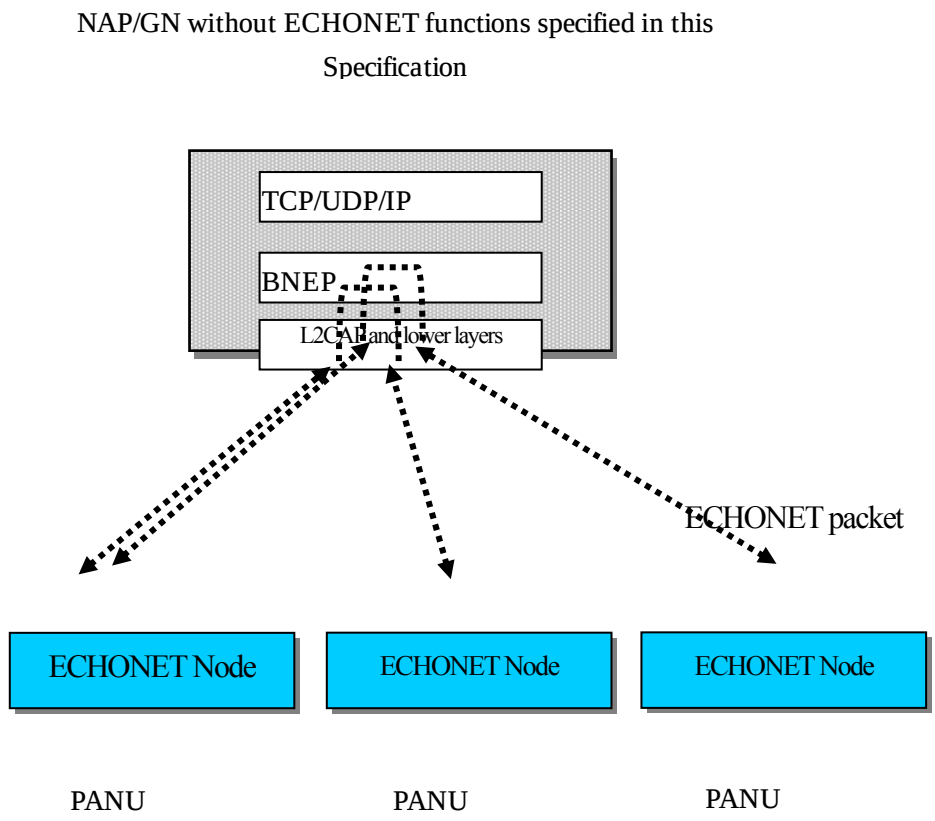


Fig. 7.45 ECHONET Packet Flows in Access Point Without ECHONET Functions

(4) PAN Profile

The mandatory functions defined by the PAN Profile are mandatory in this Specification as well, but the PANU authentication and pairing functions treated as optional functions by the PAN Profile are regarded as mandatory functions in this Specification with respect to implementation. However, inclusion of the authentication and pairing functions at the application level shall be left up to the product specification documentation.

PIN Code:

When an authentication function is implemented in a device that has only the ECHONET functions at the Bluetooth layer, the following requirements shall be satisfied to ensure ease of setting for interconnection:

1) The PANU will read in advance the serial number and the manufacturer code registered in its ECHONET profile object (a total of 15 bytes) to use as the default PIN code value. Since a function allowing the user to assign a desired PIN code is mandatory under the Bluetooth Specification, it is necessary to provide a function that enables switching between this user defined value and the above-mentioned value.

2) The GN/NAP will assign at the time of the PIN request a value equivalent to the PIN value given to

the PANU it intends to connect to. This Specification does not specify any requirements regarding input request method or input user interface.

7.9.2 Special notes

(1) Administration of Nodes after ECHONET MAC Address Acquisition

1) Even the ECHONET MAC address of a node that exists as a fixed node in a subnet may change at boot time, especially when a node using the Distributed Determination Method disconnects from the network because of a power on/off operation or a link shutoff and then reconnects to the network. Therefore, it is recommended that a mechanism be provided at the appropriate layer to allow nodes reconnecting to the network to update (using ECHONET MAC address initialization response packets and address resolution request packets) the ECHONET MAC and ECHONET address values contained in the information databases managed and maintained by them, together with the relevant ECHONET MAC and ECHONET addresses. For the Communication Middleware layer, use of the address resolution service middleware defined in Part 8 "ECHONET Service Middleware Specification" is strongly recommended.

2) When a node disconnects temporarily from the network and the ECHONET MAC address of another node located in another subnet changes during the period of temporary disconnection, the former node will be unable to recognize the address change. Therefore, it is recommended that a function be provided which updates at the time of reconnection (initialization) the information databases containing the ECHONET MAC and ECHONET addresses in the node using the same method as 1).

Appendix 7.1 Bluetooth Utility Layer

The functions unique to Bluetooth are performed independently of ECHONET communication processing even when they are mapped onto ECHONET. Therefore, it is almost meaningless to define them as part of the service middleware, except for those that can be mapped onto the individual lower-layer communication interface (i.e., those that are associated with status transitions). This is why they are not defined in the part of this Specification that relates to the ECHONET/IP layer. In the implementation stage, however, it is natural to provide above SDP, ME, and BNEP a Bluetooth Utility layer having the functions unique to Bluetooth as shown in Fig. A7.1 and to have it perform the processing associated with the application software.

The main functions of the Bluetooth Utility layer include a function to read user-friendly names found, a function to write user-friendly names, a function to read the maximum allowable number of connections, a PACKET_TYPE alteration function, a transmission power change function, a function to read link keys, a link key deletion function, a link key designation function, and a PIN designation function. These functions are not defined in this Specification because in some cases an API will be provided for each of the Bluetooth protocol stacks used. In addition, the following functions should be considered to be necessary:

A function to perform processing for reconnection after an unexpected link shutoff.

A Bluetooth layer control function for ECHONET and non-ECHONET applications.

A function to select the most appropriate GN or NAP and establish a connection when two or more GNs or NAPs with a link key have been found using a PANU inquiry, and/or a function to connect to another unconnected GN or NAP found when the node specified by the ECHONET layer as the destination node is not found in the current Piconet and look for the specified destination node.

Table A7.1 shows a sample interface between the Bluetooth Utility layer and the Bluetooth layer, and Table A7.2 shows sample status notifications from the Bluetooth layer to the Bluetooth Utility layer.

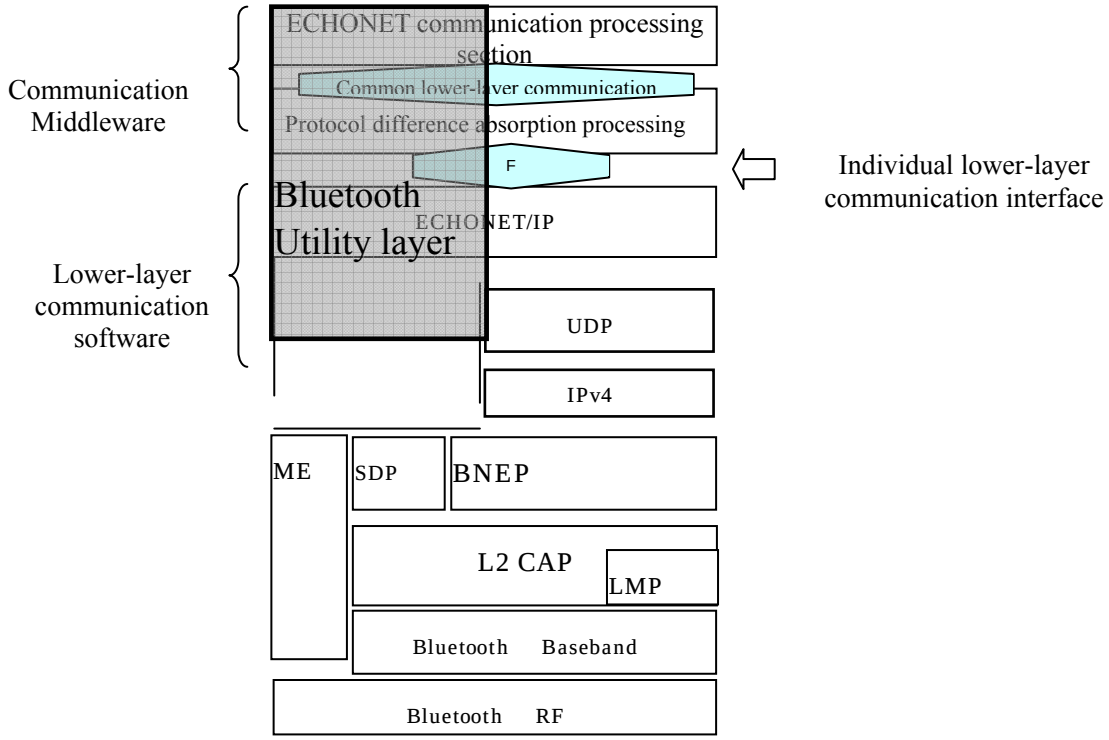


Fig. A7.1 Bluetooth Utility Layer

Table A7.1 Sample Interface Between Bluetooth Utility Layer and Bluetooth Layer

Content of command or event	Parameter passed		Purpose and related individual lower-layer communication interface
	Argument	Return value	
Reads the self BD_ADDR.		BD_ADDR	Used for initialization processing. Is related to “address information acquisition.”
Reads user-friendly name and BD_ADDR of master found using device search.		BD_ADDR, user friendly name	Used for initialization processing, candidate device selection in multiple connection cases, and reconnection of a device with a link key. Is related to “initialization request.”
Reads usable packets.		PacketType code string	Used for initialization processing. Is related to “profile acquisition.”
Reads device version information.		BT version No.	Used to create lower-layer communication software profiles. An ECHONET layer setting value may be used in lieu of this. Is related to “profile acquisition.”
Maximum allowable number of connected nodes For master only.		MAX_Connection	Used for initialization processing. Two or more are required.
Occurs when mode (Active/Park) of specified connection handle has been switched.		Status,Connection_Handle,Current _Mode	Used for switching between Active and Park. Is related to “status acquisition.”
Connection shutoff	Connection_Handle, Reason		Used for operation mode transitions and security purposes as well as to refuse inappropriate/improper connections.
Places BT layer into standby mode and resets value to default.		Status	Used for initialization processing and operation mode transitions.
Indicates change in packet type.		Status,Connection_ Handle, Packet_Type	Used to change packet type.
Changes Packet_Type.	Connection_Handle, Packet_Type		Used to change packet type.
Occurs when specified encryption change has been completed.		Status,Connection_Handle,Encrypt ion_Enable	Used for encryption.
Enables/disables encryption at link level.	Connection_ Handle, Encryption_Enable,		Used for encryption.
Reads encryption mode.		Status,Encryption_Mode	Used for encryption.
Notifies when error has occurred at PAN or lower layer.		status	Used to create a lower-layer communication software profile. Is related to “status acquisition.”
Status notification for PAN and lower layers (on standby, initialization in progress, initialization		status	Used for BT upper-layer control and upper-layer booting. Is related to “status acquisition.”

completed, in normal operation, Park mode, error stop, link key request, PIN code request).			
Indicates that connection has been broken.		Status,Connection_Handle,Reason	Used for booting and BT upper-layer control. Is related to "status acquisition."
Indicates that new connection has been established (both hosts).		Status,Connection_Handle, Link_Type,Encryption_Mode, BD_ADDR	Used for BT upper-layer booting.
Transition into Park mode	Connection_Handle, Beacon_Max_Interval, Beacon_Min_Interval		Temporary stop
Cancels Park mode.	Connection_Handle,		Used to cancel a temporary stop.
Transmission power change	Connection_Handle, Type	Status,Connection_Handle,Transmit_Power_Level	Used for power control.
Reads stored link key.	BD_ADDR, Read_All_Flag	Status,Max_Num_Keys, Num_Keys_Read	Used for authentication.
Deletes link key.	BD_ADDR, Delete_All_Flag	Status,Num_Keys_ Deleted	Used for authentication.
Specifies link key.	BD_ADDR, Link_Key	Status, BD_ADDR	Used for authentication.
Specifies PIN.	BD_ADDR, PIN_Code_Length, PIN_Code	Status, BD_ADDR	Used for authentication.

(Note) The definitions of the parameters passed shown in the table shall be as per the Bluetooth Specification Version 1.1 (Core Specification) HCI.

Table A7.2 Sample Status Notifications from Bluetooth Layer to Bluetooth Utility Layer

Nonexistent connection
Hardware failure
Page timeout
Authentication failed
Connection timeout
Maximum number of connections
Existing ACL connections
Host timeout
Connection broken by other node: connection shutoff by user
Connection broken by other node: resource shortage
Connection broken by other node: power about to be turned off or shut off
Connection broken by local host

Attempt repeated
LMP response timeout
Reconnection attempt failed

Chapter 8 IP/Ethernet/IEEE802.3 Communication Protocol Specifications

8.1 System Overview

In Chapter 7, the requirements were specified for systems that accommodate media under UDP/IP. This chapter specifies the requirements for systems that accommodate Ethernet and IEEE802.3 networks, which are currently the most popular types of UDP/IP media, under UDP/IP as an ECHONET transmission medium. The requirements specified in this chapter are for a protocol that serves as an UDP/IP application, and the ECHONET-UDP/IP interface requirements specified in Chapter 7 are applied.

Ethernet was standardized as a DIX standard (DIX = Digital Equipment Corporation, Intel Corporation and Xerox Corporation) in 1980, but this Specification calls the DIX standard "Ethernet," as is customary.

IEEE (Institute for Electronics and Electrical Engineers) has standardized, based on DIX standards, the logical link control layer (hereinafter referred to as the "LLC layer"), the "Media Access Control" (hereinafter referred to as "MAC") layer that corresponds to the data link layer and the physical layer, for IEEE802.3 networks. These have now become extremely popular as ISO/IEC standards. Because Ethernet and IEEE802.3 networks essentially differ only in terms of certain transmission frame fields, this Specification is designed to accommodate both.

Transmission methods that use IP datagrams have already been defined individually for both Ethernet and IEEE802.3 networks in Internet standards; therefore, this Specification employs these methods.

Figure 8.1 shows the relationship between the layers. ECHONET transmission frames are transmitted between nodes after being encapsulated, as UDP/IP frames, in data areas of Ethernet or IEEE802.3 network frames. The role of the part covered by this ECHONET Specification as seen from UDP/IP is that of the application layer, and the role of the part below the part covered by this chapter as seen from the ECHONET Communication Middleware is that of Layers 1 and 2. This version of the ECHONET Specification only covers the Internet Protocol Version 4 (hereinafter referred to as "IPv4"), and does not cover the Internet Protocol Version 6 (hereinafter referred to as "IPv6").

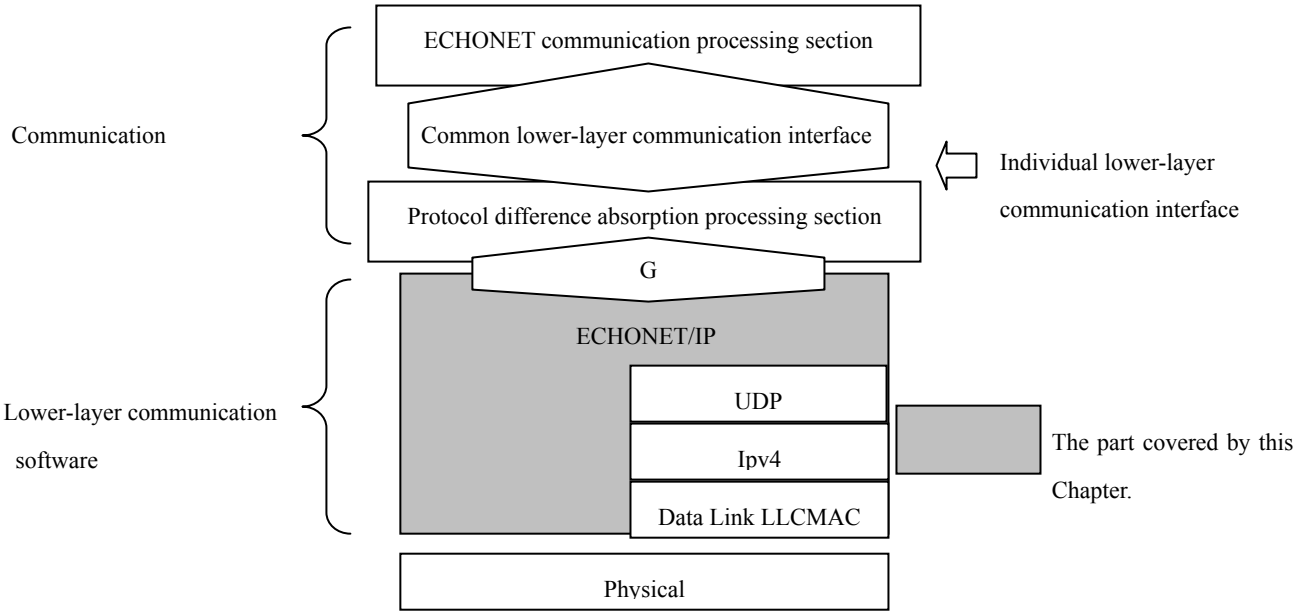


Fig. 8.1 Relationship Between Layers

8.1.1 Communication model

(1) Topology

For the physical layer, the requirements for the star and bus topologies have already been defined for different types (10Base-5, 10Base-T, etc.). Therefore, this ECHONET Specification stipulates that such requirements be satisfied for the types used. The connection to different types of media shall be achieved by means of ECHONET routers. The IP media accommodation requirements specified in 7.1.1 (i.e. the requirements relating to 1) Layer 2 bridge connection and 2) the prohibition of constructing an ECHONET subnet using two or more IP subnets connected by IP routers) shall also apply.

(2) Maximum allowable number of terminals

The maximum allowable number of terminals shall be 256 per subnet (due to the upper limit on the number of ECHONET addresses).

(3) Packet length

In the case of an IEEE802.3 network, each ECHONET frame is put into a MAC frame before transmission together with the “MAC header + trailer” (up to 18 bytes), “LLC + SNAP header” (up to 8 bytes), “IP header (IPv4, up to 24 bytes) + UDP header (up to 8 bytes)” and “FCS” (up to 4 bytes). Because the sum of these must be

1518 bytes or less, the maximum data length is 1518 – (18 + 8 + 24 + 8 +4) = 1456 bytes, which means that the maximum packet length for ECHONET transmission frames (the sum of the maximum ECHONET frame size of 262 bytes, the SA/DA data size and the EDC size) can be easily supported and there is no need to split an ECHONET transmission frame into two or more split frames

(4) Timeout period

The time allowed to wait for an incoming packet sent by another node in response to a packet transmitted by the home node varies between systems and between statuses depending on a variety of factors such as the performance level of the bridges and the processing speed of the nodes located in the subnet including the bridges (when there are bridges) and the total number of nodes. Taking into consideration these conditions and the need to achieve a sufficient level of interconnectivity, this version of the ECHONET Specification defines fixed timeout periods that can be used for other types of media (including Bluetooth) as well. A method to dynamically determine timeout periods and other necessary techniques shall be defined in succeeding versions of the ECHONET Specification as necessary.

8.1.2 Applicable standards

When using an 802.3 network, the requirements specified in the applicable sections of the following standards shall be satisfied:

IEEE Std. 802 Overview and Architecture

ANSI/IEEE Std. 802.2 Logical Link Control (ISO/IEC 8802-2)

ANSI/IEEE Std. 802.3 CSMA/CD Access Method and Physical Layer Specifications (ISO/IEC 8802-3)

When using Ethernet, the requirements specified in the applicable sections of the following standards shall be satisfied:

D-I-X, “The Ethernet – A Local Area Network: Data Link Layer and Physical Layer Specifications,” Digital, Intel, and Xerox, November 1982

All Ethernet-type fields must satisfy the applicable requirements specified as numerical values in Ethernet Numbers (copies of Ethernet Numbers can be obtained at http://www.iana.org/numbers.htm).

The UDP/IP-related standards are listed in Section 8.6.

8.1.3 Coverage of the ECHONET Specification

This ECHONET Specification defines the specifications for the interface between the

ECHONET Communication Middleware and Ethernet/IEEE802 networks and between the ECHONET Communication Middleware and the UDP/IP layers (see Fig. 8.1). The detailed mechanical, physical, electrical and logical specifications for Ethernet/IEEE802 networks and the UDP/IP layers shall be as defined in the relevant standards.

The ECHONET/IP layer references the status of the Ethernet or IEEE 802 network layer as necessary in the course of performing the necessary processing and control commands are exchanged with the ECHONET/IP layer as necessary, but this ECHONET Specification does not specify any interface requirement for these.

This ECHONET Specification does not specify any requirement for a method to allow the coexistence on the Ethernet or IEEE802 network layer of the ECHONET layer with other UDP/IP applications.

8.2 Mechanical and Physical Specifications

When using an IEEE802 network, the requirements specified in the applicable sections of ANSI/IEEE Std. 802.3 “CSMA/CD Access Method and Physical Layer Specifications” shall be satisfied.

10 Mbps	ANSI/IEEE Std. 802.3 (Chapters 8 through 20)	10Base-T, etc.
100 Mbps	ANSI/IEEE Std. 802.3 (Chapters 21 through 29)	100Base-T, etc.
1000 Mbps	ANSI/IEEE Std. 802.3 (Chapters 36 through 42)	1000Base-T, etc.

When using Ethernet, the requirements specified in D-I-X “The Ethernet – A Local Area Network: Data Link Layer and Physical Layer Specifications” (Digital, Intel, and Xerox, November 1982) shall be satisfied.

8.3 Electrical Specifications

When using an IEEE802 network, the requirements specified in the applicable sections of ANSI/IEEE Std. 802.3 “CSMA/CD Access Method and Physical Layer Specifications” shall be satisfied.

10 Mbps	ANSI/IEEE Std. 802.3 (Chapters 8 through 20)	10Base-T, etc.
100 Mbps	ANSI/IEEE Std. 802.3 (Chapters 21 through 29)	100Base-T, etc.
1000 Mbps	ANSI/IEEE Std. 802.3 (Chapters 36 through 42)	1000Base-T, etc.

When using Ethernet, the requirements specified in D-I-X “The Ethernet – A Local Area Network: Data Link Layer and Physical Layer Specifications” (Digital, Intel, and Xerox, November 1982) shall be satisfied.

8.4 Overview of the Logical Specifications

The logical specifications for the IP layer and the overlying layers shall be as stated in

Section 7.4. ECHONET packets are encapsulated, before being sent out to the transmission path, first by UDP/IP and then by the protocol defined for the Ethernet or IEEE802.3 network. This section describes the logical specifications for the layers below the IP layer. As shown in Figs. 8.2 and 8.3, the difference between Ethernet and IEEE802.3 resides in the data link layer. That is, an IEEE802.3 network uses frames containing headers added by the Sub-Network Access Protocol (hereinafter referred to as "SNAP") and the LLC layer. Figure 8.4 shows the difference between Ethernet and IEEE802.3 in the frame content (at the termination of the physical layer). Ethernet and IEEE802.3 are different in terms of the type of data contained in the field marked "Length or type" in Fig. 8.4, which means that the coexistence of the two systems can be achieved by discriminating between the value ranges. The contents of the headers of the individual layers are explained in Section 8.5.

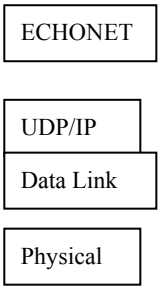


Fig. 8.2 Logical Layers (Ethernet)

ECHONET	
UDP/IP	
DATA Link	LLC+SNAP (802.2) Sub-layer
	MAC (802.3)
Physical	Physical Signaling (802.3) Sub-layer
	Media (802.3)

Fig. 8.3 Logical Layers (IEEE802.3)

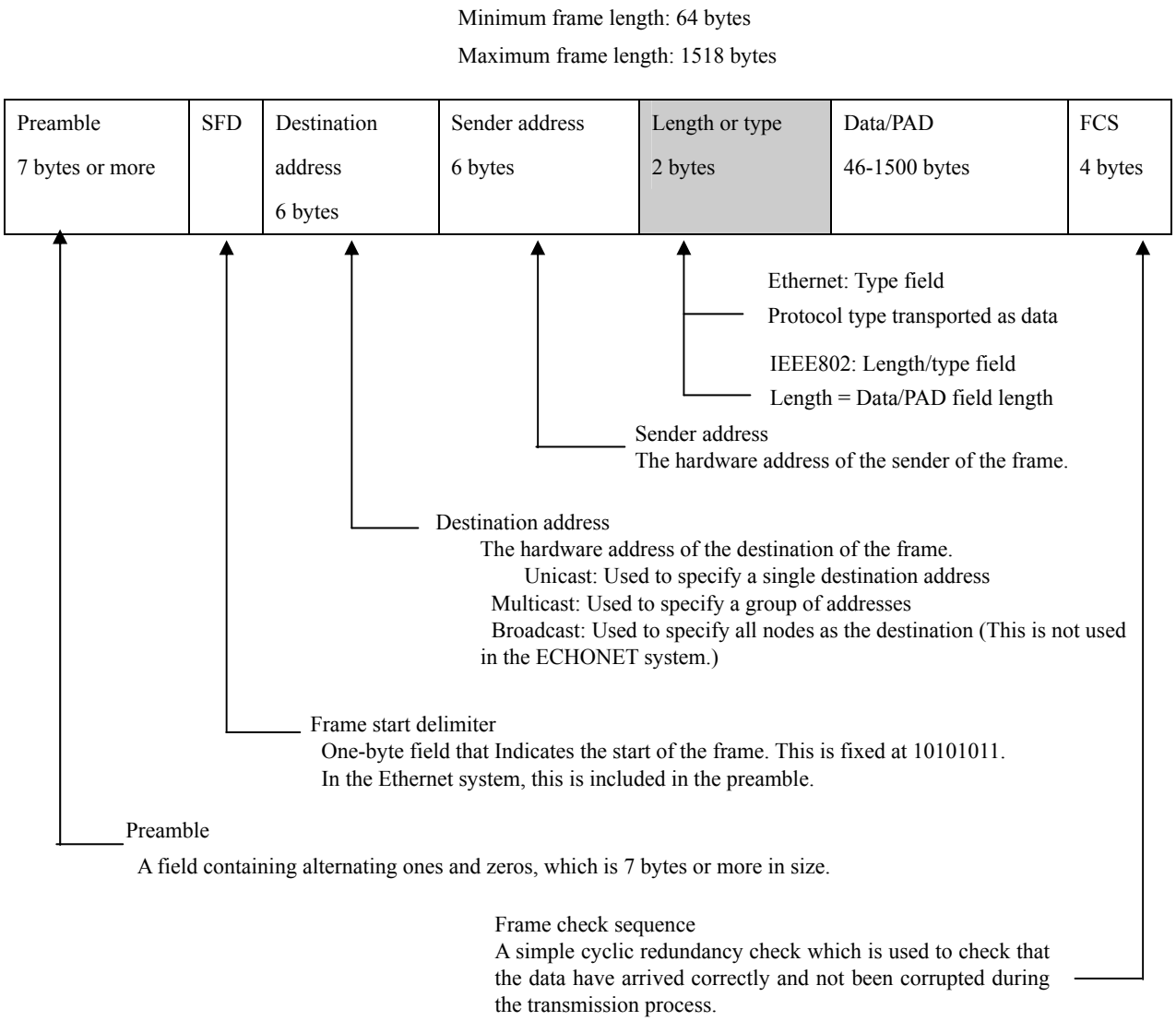


Fig. 8.4 Difference between Ethernet and IEEE802.3 in Frame Content

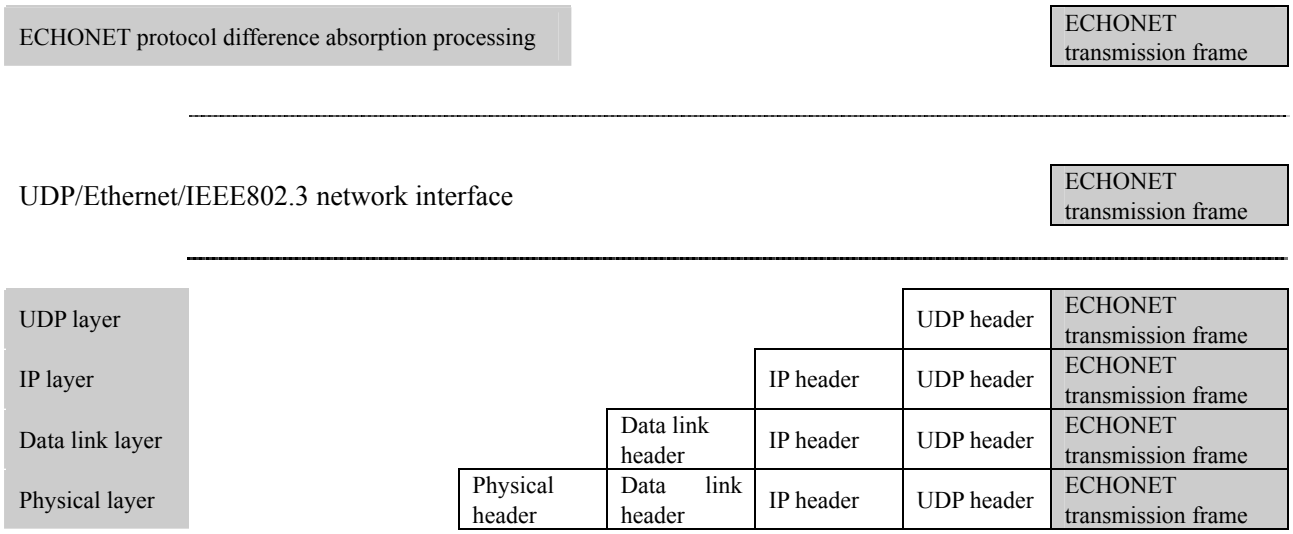


Fig. 8.5 Headers (Ethernet)

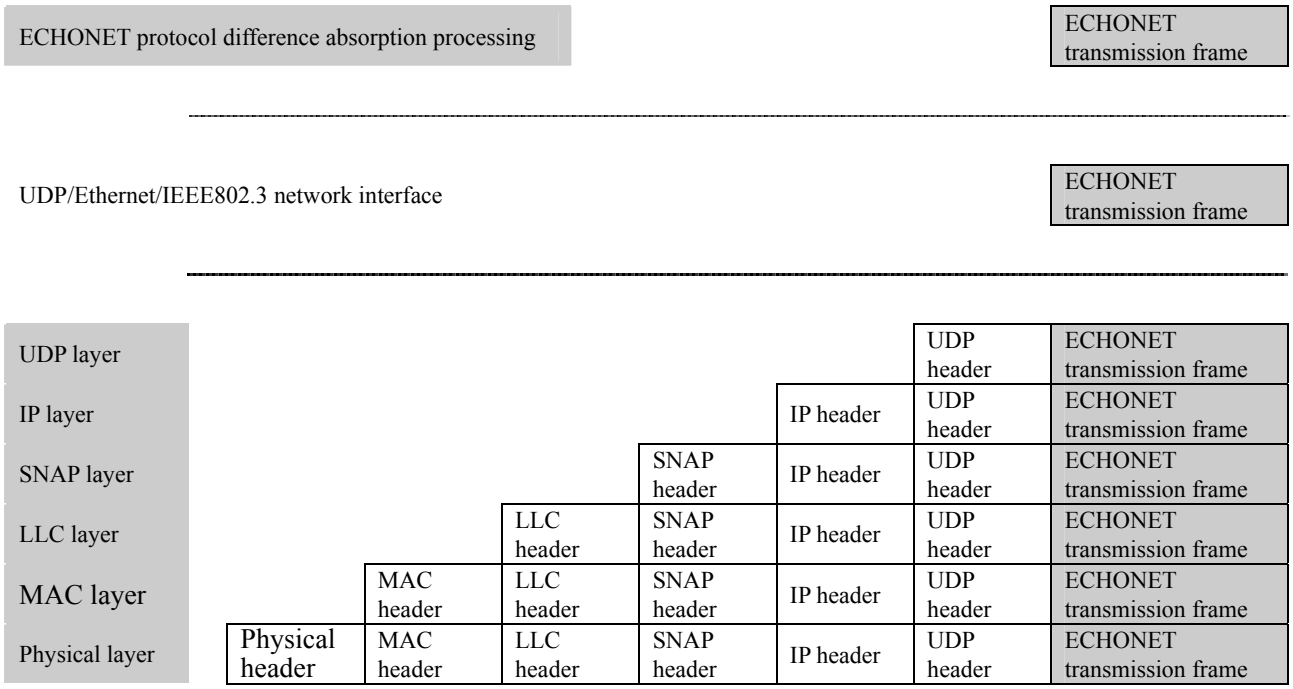


Fig. 8.6 Headers (IEEE802.3)

8.5 Logical Specifications (Ethernet/IEEE802.3 Network Layer)

This section specifies the requirements for the MAC and LLC layers (data link layer).

The LLC layer of an IEEE802.3 network provides the IP layer with the Type 1 (connectionless) and Type 2 (connection) services and accepts the data request, data indication and status indication services provided by the MAC layer. In an IEEE802.3 network or Ethernet, different protocols are used depending on the functions available (i.e. SNAP, LLC). For the detailed requirements for the LLC layer and SNAP, refer to ANSI/IEEE Standard 802.2 “Logical Link Control” (ISO/IEC 8802-2). For the IPv4 requirements, refer to RFC 1122, RFC 894 and RFC 948.

The IEEE802.3 system adds more headers than the Ethernet system. In the IEEE802.3 system, a protocol ID or an organization code together with type information is added by SNAP and destination and sender service access point information and a control code are added by the LLC layer. The destination service access point information consists of 8 bits and the highest-order bit indicates whether a unicast (0) or group address (1) is used. ANSI/IEEE Standard 802.2 requires that the sender service access point information consist of 8 bits and that the highest-order bit indicates whether the content is a command (0) or response (1). RFC 948 requires the use of 0xAA as both the destination and sender service access point information, 0x00 as the protocol ID or organization code and 0x03 as the control code.

For the detailed requirements for the MAC layer (data link layer), refer to the applicable sections of ANSI/IEEE Standard 802.3 when using an IEEE802.3 network and the applicable sections of D-I-X “The Ethernet – A Local Area Network: Data Link Layer and Physical Layer Specifications” when using Ethernet. As the CSMA/CD (Carrier Sense Multiple Access / Collision Detect) system is used for the MAC layer, any collision of packets sent from two or more nodes is detected within the preamble at the hardware level and the packets are retransmitted by a backoff algorithm-based automatic retransmission function in a such manner that they do not collide with each other. The backoff time is determined by multiplying the one slot time (512 bit time) by r, which is an integer satisfying the following condition:

$$0 \leq r < 2^k$$

where k and n represent “min (n, 10)” and the number of retries, respectively (If n exceeds 16, no more attempts will be made to transmit the frame).

With regard to the destination and sender addresses, it is required to embed the contents of the 23 lower-order bits of any Class D IP multicast address in the bits following “0x01005e” under RFC 1112 (IPv4). In addition, the lowest-order bit of the highest-order byte of a multicast address must be 1 and the size of a unicast address must be 48 bits. It is also required that all bits of a broadcast address contain 1, but no broadcast address is used in the ECHONET system.

Ethernet Numbers (copies of Ethernet Numbers can be obtained at http://www.iana.org/numbers.htm) requires, for Ethernet, that the field marked “Length or type” in Fig. 8.4 contain 0x0600 or a larger value as the “type” data (The value is 0x0800 in the case of IPv4 and 0x0806 in the case of ARP). Because the IEEE802.3 system uses

0x05dc (1500 in the decimal notation) or a smaller value as the “length” data, the coexistence of the two systems can be achieved by discriminating between values contained in the “type” and “length/type” fields. The requirements in relation to the coexistence of Ethernet and IEEE802.3 networks on the IP layer that are specified in RFC 1122 are as follows:

1) A function to send and receive Ethernet packets must be provided;

2) A function to receive IEEE802.3 packets intermixed with Ethernet packets should be provided; and

3) A function to send IEEE802.3 packets may be provided.

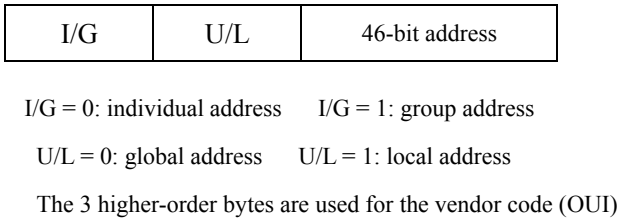


Fig. 8.7 Format for Destination and Sender Addresses (IEEE802 Network)

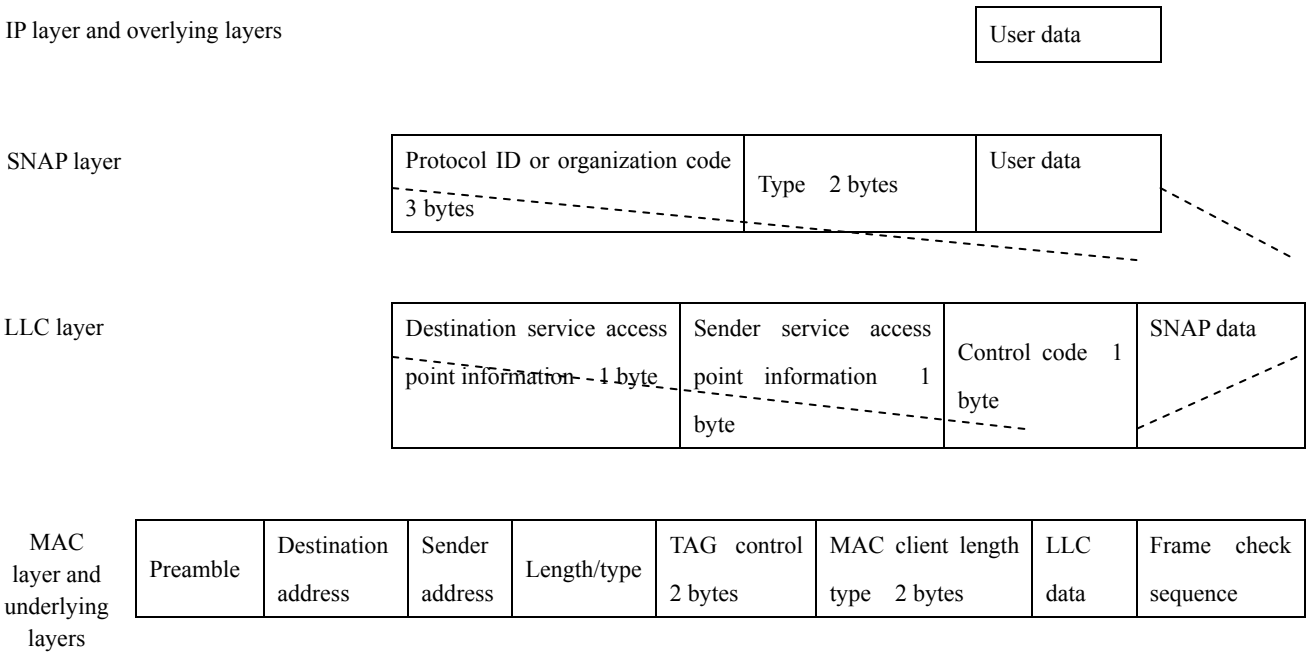


Fig. 8.8 IEEE802.3 Frame Composition

8.6 Logical Specifications (UDP/IP Layer)

The logical specifications for the UDP/IP layer shall be as specified in Section 7.6, with the exception that the following RFC documents shall be used in addition to those listed in

(1)(A) Protocols Used and Applicable RFC Documents:

RFC-826 "An Ethernet Address Resolution Protocol or Converting Network Protocol Addresses to 48bit Ethernet Address for Transmission on Ethernet Hardware"
RFC-1700 "Assigned Numbers"

When using an IEEE802 network, the requirements specified in the following RFC document shall be satisfied:
RFC948 "A Standard for the Transmission of IP Datagram over IEEE802 Networks"

When using Ethernet, the requirements specified in the following RFC document shall be satisfied:
RFC894 "A Standard for the Transmission of IP Datagram over Ethernet"

8.7 Logical Specifications (ECHONET/IP Layer)

The logical specifications for the ECHONET/IP Layer shall be as specified in Section 7.7, with the exception that the following values shall be used for UDF packets:

Hardware type	Ethernet (10Mb)	1
	IEEE802.3	6
Hardware address	Ethernet	48-bit Ethernet address
	IEEE802.3	48-bit IEEE802 address

8.7.1 Time requirements

The time requirements specified in Section 7.7.6 shall apply, with the exception that the T1 value in "T7 = ECHONET MAC address value x T1 + T10" must be 5 ms.

8.8 Basic Sequences

This section outlines the sequences for the following states of individual lower-layer communication software (The sequences are for cases in which the ECHONET/IP layer is the only layer using the transmission medium):

① Stop
② Initialization processing in progress
③ Communication stop
④ Normal operation
⑤ Error stop
⑥ Suspension

Figure 8.9 shows how state transitions are made among the 6 states.

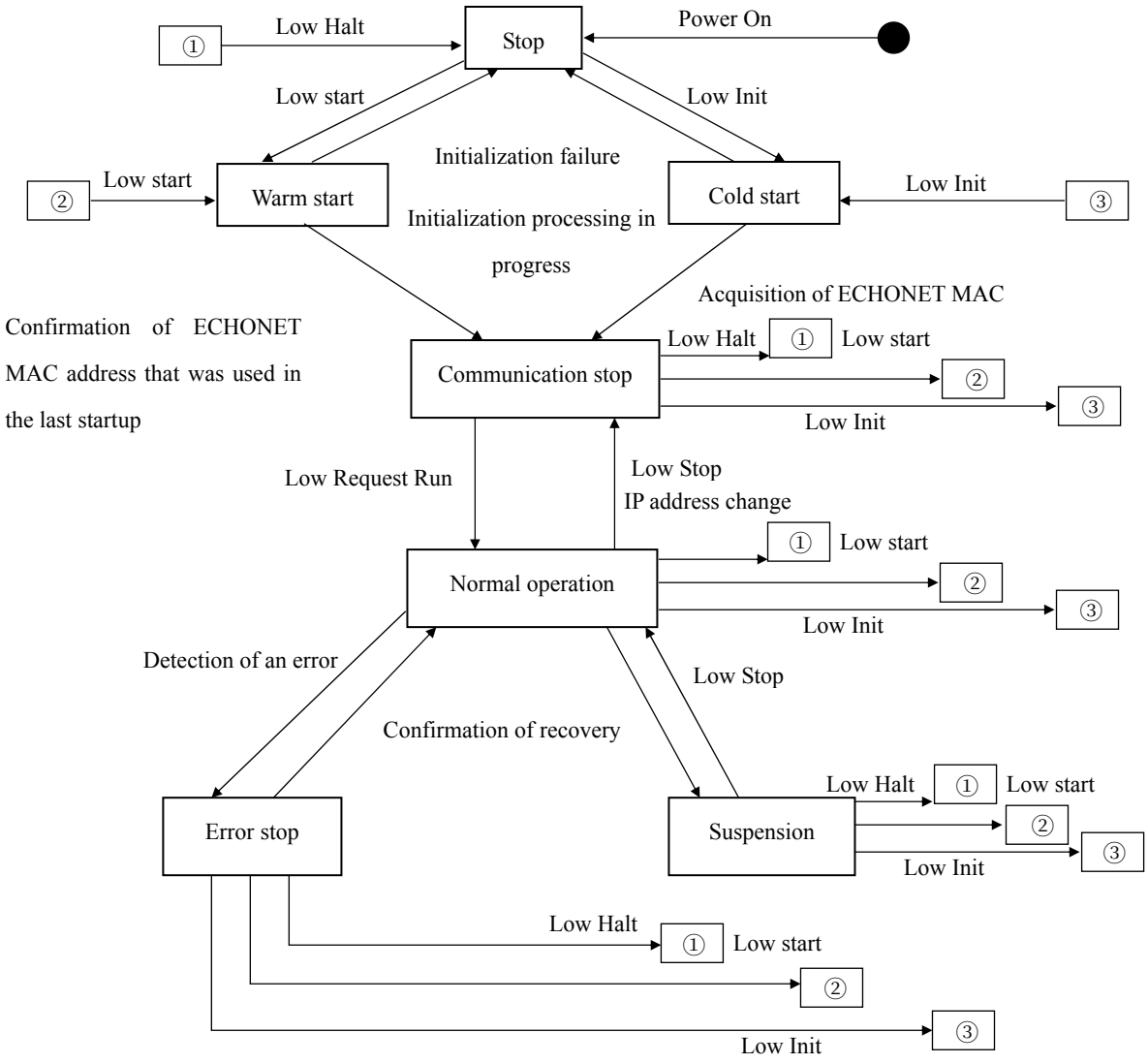


Fig. 8.9 State Transition Diagram

8.8.1 "Stop" status

The "stop" status is a state in which the lower-layer communication software has stopped operating and all data except the ECHONET MAC address has been initialized. Whenever the power is turned on, this state will be entered. An overview of the processing to be performed immediately after the state transition is given below together with a list of the individual lower-layer communication interface services that can be accepted during the "stop" state with brief explanations of the processing in relation to the services.

(1) Trigger and the response behavior

Waits for an individual lower-layer communication interface service.

All but the lower-layer communication software is initialized upon Power On.

The IP address is acquired during the "initialization processing in progress" state.

(2) Status acquisition service (LowGetStatus)

Returns LOW_STS_STOP as the status.

(3) Lower-layer communication software type acquisition service (LowGetDevID)

Returns the type of lower-layer communication software.

Trigger for a transition to the "initialization processing in progress" state:

Initialization request service (LowInit), warm start request service (LowStart)

8.8.2 "Initialization Processing in Progress" status

The "initialization processing in progress" status is a state in which addresses are being acquired. An overview of the processing to be performed immediately after the state transition is given below together with a list of the individual lower-layer communication interface services that can be accepted during the "initialization processing in progress" state with brief explanations of the processing in relation to the services.

(1) Trigger and the response behavior

First acquires an IP address and then an ECHONET MAC address as instructed by LowStart/LowInit from the Communication Middleware. The warm start mode is a mode in which the acquisition process is started using the stored ECHONET MAC address, and the cold start mode is a mode in which a new ECHONET MAC address is acquired after discarding the stored ECHONET MAC address. If it is discovered during a warm start that the stored ECHONET MAC address is already being used by another node, the processing to acquire a new ECHONET MAC address will be started automatically. This ECHONET Specification does not specify any requirements in relation to functions to reacquire IP addresses during warm and cold starts. If any other abnormal condition

occurs, a transition will be made to the “stop” state.

(2) Status acquisition service (LowGetStatus)
Returns, as the status, LOW_STS_INI in the case of a cold start and LOW_STS_RST in the case of a warm start.

(3) Lower-layer communication software type acquisition service (LowGetDevID)
Returns the type of lower-layer communication software.

State transition triggers:

(1) Trigger for a transition to the “communication stop” state
Completion of acquisition of an ECHONET MAC address

(2) Trigger for a transition to the “stop” state
Failure to acquire an IP or ECHONET MAC address

8.8.3 “Communication Stop” status

The “communication stop” status is a state in which an operation start request from the Communication Middleware is being waited for after completion of the initialization of the lower-layer communication software. An overview of the processing to be performed immediately after the state transition is given below together with a list of the individual lower-layer communication interface services that can be accepted during the “communication stop” state with brief explanations of the processing in relation to the services.

(1) Trigger and the response behavior
Waits for an individual lower-layer communication interface service.

(2) Status acquisition service (LowGetStatus)
Returns LOW_STS_CSTOP as the status.

(3) Physical address acquisition service (LowGetAddress)
Returns the ECHONET MAC address.

(4) Profile data acquisition service (LowGetProData)
Returns the profile data.

(5) Lower-layer communication software type acquisition service (LowGetDevID)

Returns the type of lower-layer communication software.

State transition triggers:
(1) Trigger for a transition to the “normal operation” state
Operation start instruction service (LowRequestRun)

(2) Trigger for a transition to the “initialization processing in progress” state
Initialization request service (LowInit), warm start request service (LowStart)

(3) Trigger for a transition to the “stop” state
Stop service (LowHalt)

8.8.4 “Normal Operation” status

The “normal operation” status is a state in which a message is being transmitted to or received from the transmission medium (i.e. a state in which the primary function of the lower-layer communication software is being performed). An overview of the processing to be performed immediately after the state transition is given below together with a list of the individual lower-layer communication interface services that can be accepted during the “normal operation” state with brief explanations of the processing in relation to the services.

(1) Trigger and the response behavior
Waits for an individual lower-layer communication interface service.

(2) Status acquisition service (LowGetStatus)
Returns LOW_STS_RUN as the status.

(3) Physical address acquisition service (LowGetAddress)
Returns the ECHONET MAC address.

(4) Profile data acquisition service (LowGetProData)
Returns the profile data.

(5) Message transmission service (LowSendData)
Converts the provided protocol difference absorption processing section message into a lower-layer communication software message and outputs it to the transmission medium (The message will be split into two more split messages before the conversion when the original message size requires such splitting).

(6) Message reception service (LowReceiveData)
Converts the lower-layer communication software message received from the transmission medium into a protocol difference absorption processing section message and outputs it to the Protocol Difference Absorption Processing section.

(7) Lower-layer communication software type acquisition service (LowGetDevID)
Returns the type of lower-layer communication software.

State transition triggers:
(1) Trigger for a transition to the “suspension” state
Lower-layer communication section stop service (LowSuspend)

(2) Trigger for a transition to the “stop” state
Stop service (LowHalt) or an IP address change on the IP layer

(3) Trigger for a transition to the “error stop” state
An error (lower-layer communication software)

(4) Trigger for a transition to the “initialization processing in progress” state
Initialization request service (LowInit), warm start request service (LowStart)

(5) Trigger for a transition to the “communication stop” state
Stop service (LowStop)

8.8.5 “Error Stop” status

The “error stop” status is a state in which the operation of the software has been stopped as a result of an error. An overview of the processing to be performed immediately after the state transition is given below together with a list of the individual lower-layer communication interface services that can be accepted during the “error stop” state with brief explanations of the processing in relation to the services.

(1) Trigger and the response behavior
Performs the error processing. Any message being received will be discarded and any new or outstanding message transmission request will be rejected and an error will be returned, before the transition to the “error stop” state becomes effective.

(2) Status acquisition service (LowGetStatus)
Returns LOW_STS_ESTOP as the status.

(3) Lower-layer communication software type acquisition service (LowGetDevID)
Returns the type of lower-layer communication software.

State transition triggers:
(1) Trigger for a transition to the “stop” state
Stop service (LowHalt)

(2) Trigger for a transition to the “initialization processing in progress” state
Initialization request service (LowInit), warm start request service (LowStart)

(3) Trigger for a transition to the “normal operation” state
Removal of the cause of the error

8.8.6 “Suspension” status

The “suspension” status is a state in which the operation of the software has been temporarily stopped in response to an instruction from the communication middleware. An overview of the processing to be performed immediately after the state transition is given below together with a list of the individual lower-layer communication interface services that can be accepted during the “suspension” state with brief explanations of the processing in relation to the services.

(1) Trigger and the response behavior
Stops the operation of the lower-layer communication software.
Any message being received will be discarded and any new or outstanding message transmission request will be rejected and an error will be returned.

(2) Status acquisition service (LowGetStatus)
Returns LOW_STS_SPD as the status.

(3) Lower-layer communication software type acquisition service (LowGetDevID)
Returns the type of lower-layer communication software.

State transition triggers:
(1) Trigger for a transition to the “normal operation” state
Operation restart service (LowWakeUp)
The lower-layer communication software will restart its transmission and reception functions immediately.

(2) Trigger for a transition to the “stop” state

Stop service (LowHalt)

(3) Trigger for a transition to the “initialization processing in progress” state
Initialization request service (LowInit), warm start request service (LowStart)

8.9 Accommodation Requirements

For the accommodation requirements, refer to 7.9.1(2), (5) and 7.9.2, with the exception that MAC address server equipment shall be accepted as “ECHONET MAC address server equipment for IP/Ethernet/IEEE802.3.” For the specific requirements, refer to Part 7.

Chapter 9 IEEE802.11/11b Communication Protocol Specifications

9.1 System Overview

This chapter defines the requirements for the system for accommodating for ECHONET transmission media the 2.4GHz band radio LAN defined in the IEEE802.11 and IEEE802.11b standards.

The IEEE802.11 standard defines the requirements for the physical and MAC layers, and the IEEE802.11b standard defines the physical layer requirements relating to communication speeds that are higher than the communication speeds defined in IEEE802.11. The IEEE802.11 physical layer requirements cover direct sequence spread (DSS) system-based 2.4 GHz band radio and infrared system- and FH system-based 2.4 GHz band radio. The IEEE802.11b physical layer requirements cover direct sequence spread system-based 2.4 GHz band radio, thereby ensuring upward compatibility with IEEE802.11.

The ECHONET Specification shall adopt the direct sequence spread system-based radio LAN defined in the IEEE802.11 and IEEE802.11b standards.

Chapters 7 and 8 of this ECHONET Specification define the requirements for the systems for accommodating IEEE802.3, Ethernet and Bluetooth® together with UDP/IP. On the other hand, the IEEE802.11 and IEEE802.11b standards only define the requirements for the radio LAN physical and MAC layers, thereby permitting access from the IEEE802.2 LLC layer in the same manner as in the case of IEEE802.3.

Therefore, in the ECHONET Specification, IEEE802.11/11b shall be accommodated in the same manner as in the case of Ethernet and IEEE802.3.

The relationship between the layers is as shown in the figure below.

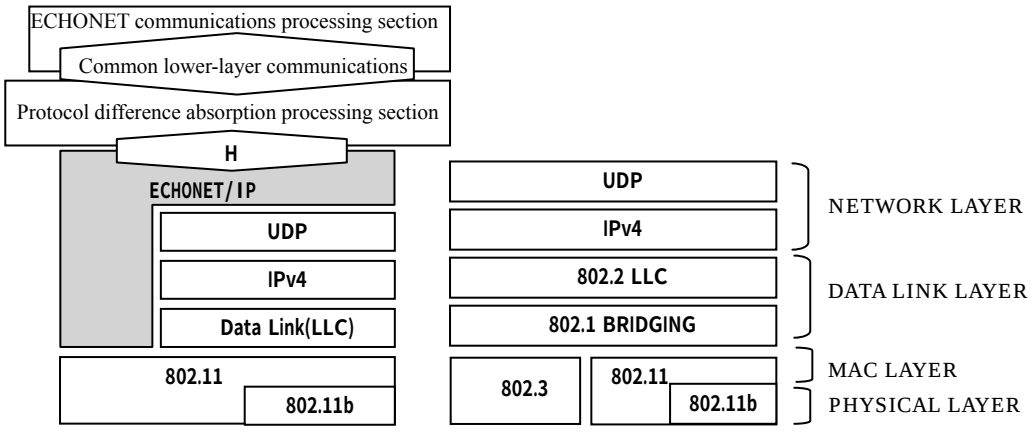


Fig. 9-1 Relationship between the Layers

9.1.1 Definitions of Terms

(1) Communication mode

In radio LAN communications based on the IEEE802.11/11b standard, networks are classified into 2 types; “infrastructure networks,” which use an intermediary access point for communications, and “ad-hoc networks,” in which radio LAN clients communicate with each other without using an intermediary access point. These 2 communication methods shall hereinafter be referred to as “infrastructure communication mode” and “ad-hoc communication mode.”

(2) ECHONET MAC addresses and hardware addresses

MAC addresses used in communications based on the IEEE802.11 standard are called “hardware addresses” in communications based on the ECHONET Specification. In the context of the ECHONET Specification, the term “MAC addresses” normally refers to ECHONET MAC addresses.

In this chapter, too, MAC addresses used in communications based on the IEEE802.11 standard shall in principle be referred to as hardware addresses, and MAC addresses used in communications based on the ECHONET Specification shall in principle be referred to as ECHONET MAC addresses.

However, the term “MAC addresses” used without the prefix “ECHONET” in the parts of this chapter which quote IEEE or OSI reference models shall mean hardware addresses, because such parts may confuse the reader if network terms were not used.

ECHONET MAC addresses: MAC addresses used in communications based on the ECHONET Specification

MAC addresses and hardware addresses: MAC addresses that fall under the definition of “MAC addresses” as a standard network term

9.1.2 Communication model

Processing relating to the IEEE802.11/11b communication modes are handled by the MAC LAYER shown in Fig.9.1. Therefore, the IEEE802.11/11b communication modes are hidden to the ECHONET by the lower-layer communication software, and can only be accessed through individual lower-layer communication interfaces. In communications based on the ECHONET Specification, switching between the IEEE802.11/11b communication modes shall be by means of initialization parameters for the IEEE802.11/11b lower-layer communication software (individual lower-layer communication software). Appendix 9.1 provides the detailed scenarios for starting up ECHONET nodes equipped with IEEE802.11/11b media.

(1) Topology

1) Infrastructure networks

Infrastructure networks are networks that use an access point. These networks are called

"basic service sets" (BSS's). In this ECHONET Specification, these networks shall be referred to as infrastructure mode networks.

All communication traffic between nodes, or terminal stations (STA's), shall go through a base station (called an "access point" (AP)). As mentioned above, since the communication modes are managed by the MAC layer, the AP is not recognized by the ECHONET devices.

AP's shall be such that they can be used in communications based on the ECHONET Specification to the extent specified by the ANSI/IEEE STD 802.11, which is one of the applicable standards specified in "9.1.3 Applicable standards."

Fig. 9-2 shows the network topology of an IEEE802.11 infrastructure mode network in the ECHONET. In the IEEE-802.11/11b-based part of the topology, packets travel between nodes along the paths represented by the arrowed solid lines. In the ECHONET Specification-based part of the topology, it looks as if nodes are directly communicating with each other (arrowed white thick lines).

The relationship between IEEE802.11/11b-based IP addresses and ECHONET Specification-based EA's shall be as specified in the sections entitled "ECHONET MAC address acquisition initialization sequence" and "ECHONET MAC address servers" of Chapter 7 of Part 3.

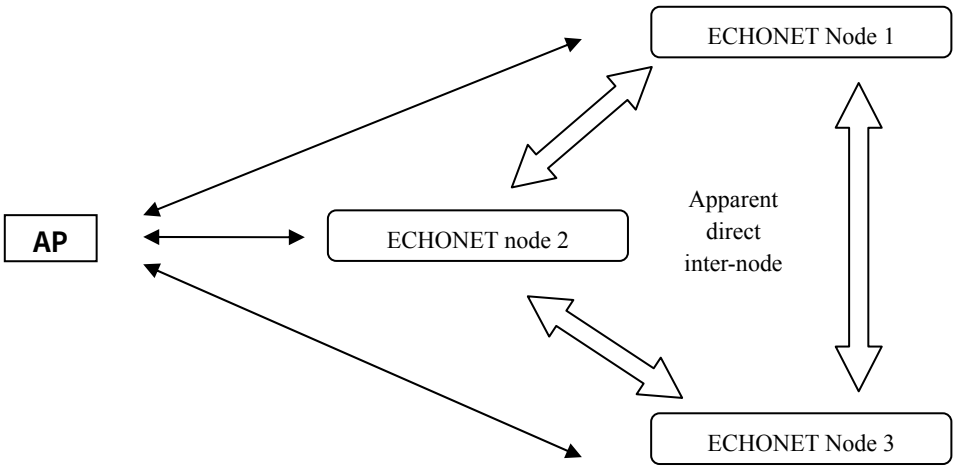


Fig. 9-2 Network Topology of an Infrastructure Mode Network

2) Ad-hoc networks

Ad-hoc networks are networks that consist of terminal stations (STA's) only and do not require a base station (AP). These networks are called "independent basic service sets" (IBSS's) so that they can be distinguished from BSS's (infrastructure mode networks).

In this ECHONET Specification, these networks shall be referred to as ad-hoc mode networks.

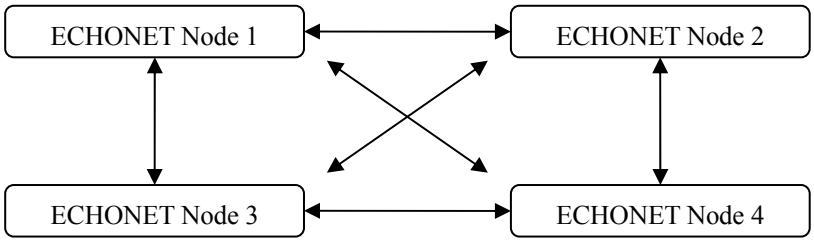


Fig. 9-3 Network Topology of an Ad-hoc Mode Network

In an ad-hoc mode network, nodes communicate with each other by directly exchanging radio packets. As is clear from a comparison of Fig. 9-2 with Fig. 9-3, there is no difference in topology between the two communication modes as far as communications based on the ECHONET Specification are concerned.

3) Other communication modes

In communications based on IEEE802.11, other types of networks, including extended infrastructure networks (ESS's) and wireless distribution systems (WDS's), are also used in addition to infrastructure networks (BSS's) and ad-hoc networks (IBSS's) mentioned above.

A) Extended infrastructure network (ESS)

The AP is usually connected to a backbone network such as an Ethernet network. A network with 2 or more AP's connected to the same backbone network is called an "extended service set" (ESS) to make it clear that it is not a basic service set (BSS).

The figure below shows the topology of an ESS.

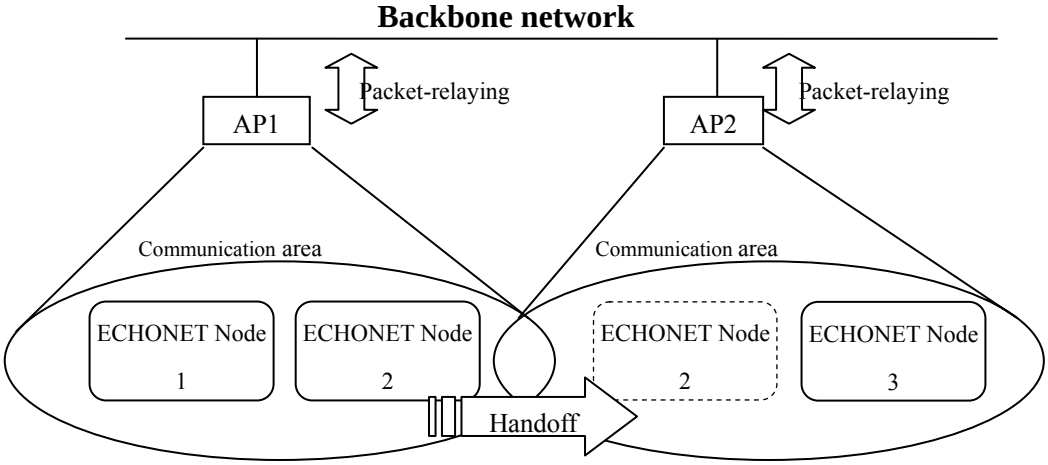


Fig. 9-4 Network Topology of an Extended Infrastructure Network

Each AP has its own "communication area." The AP can communicate with its nodes within this area (which is usually called the "radio cell"). An ECHONET node that can move between radio cells should preferably be equipped with a function that allows it to independently detect each of its inter-radio cell positional changes and switch to the AP for the destination radio cell. Such a function is usually called the "handoff" function.

Each AP can also relay packets to the backbone network. However, the IEEE802.11 standard does not specify a specific method to achieve handoffs or packet relaying.

As shown by the arrowed white thick lines in the upper half of Fig. 9-4, each AP relays packets between a radio LAN network and a different type of network. As per the first requirement specified in the "Common accommodation requirements for all IP medium types" section below, the Node 1 to Node 3 in Fig. 9-4 shall be regarded to belong to the same ECHONET subnet.

B) Wireless distribution system (WDS)

A wireless distribution system is a network that consists of AP's only. Fig. 9-5 shows a typical wireless distribution system topology.

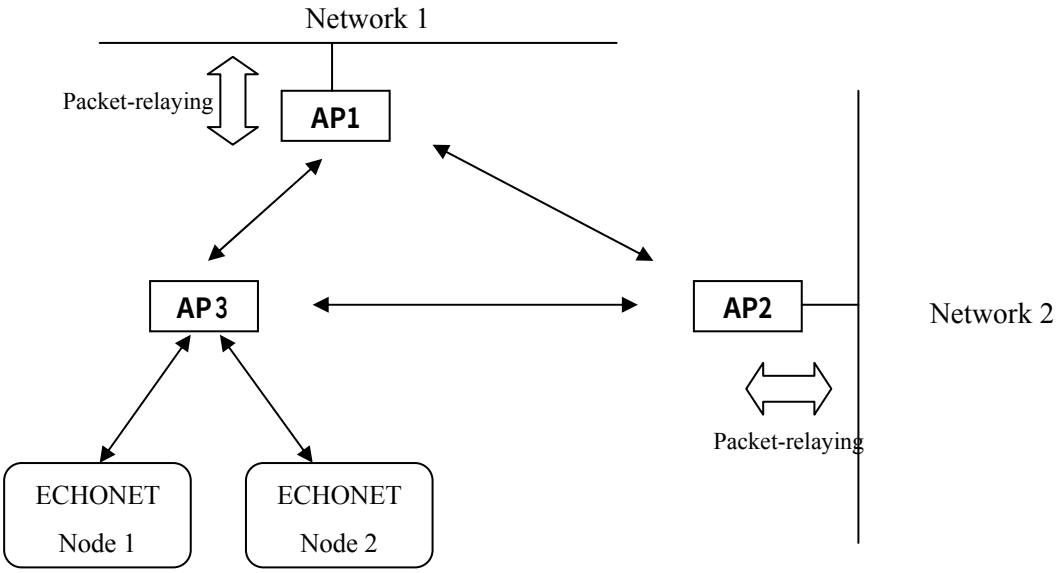


Fig. 9-5 Network Topology of a Wireless Distribution System

The wireless distribution system shown in Fig. 9-5 consists of two AP's; AP1 and AP2. AP3 is an ordinary infrastructure network. The IEEE802.11 standard thus permits the coexistence of a WDS and infrastructure networks. However, the ECHONET Specification does not permit the networks 1 and 2 in Fig. 9-5 to be different subnets, on the basis of the second requirement of the "Common accommodation requirements for all IP medium types" subsection below.

(2) Common accommodation requirements for all IP medium types

The "common accommodation requirements for all IP medium types" specified in Section 7.1.1 shall apply to IEEE802.11/11b media as well. That is, the following two accommodation requirements shall apply.

- Node sets using different ECHONET media but connected using Layer 2 bridges are defined as a single ECHONET subnet.
- It is not allowed to use an IP network comprising two or more IP subnets connected by means of an IP router or IP routers as one ECHONET subnet, and connection between ECHONET subnets shall be by means of an ECHONET router or ECHONET routers.

(3) IEEE802.11/11b-specific accommodation requirement

If the network 1 and network 2 in Fig. 9-5 are turned into subnets, then the Node 1 and Node 2 in the figure will belong to both subnets, which will constitute a discrepancy. To avoid this, the following IEEE802.11/11b-specific accommodation requirement is provided:

- IP networks that have been turned into subnets shall not be connected together using a WDS, and connection between subnets shall be by means of an ECHONET router. However, this requirement does not apply in the case where a WDS is used as a repeater in the same subnet.

(4) IEEE802.11/11b-specific accommodation requirement

The purpose of this part of the ECHONET Specification is to accommodate radio LAN communications based on the IEEE802.11 standard as well as communications based on the IEEE802.11b standard (which is aimed at achieving higher-speed communications in the physical layer within the scope of the IEEE802.11 standard) in the ECHONET. Therefore, it is necessary to provide requirements to ensure the interconnectivity between IEEE802.11 and IEEE802.11b.

A standard already exists which is aimed at achieving this. It is the "Wireless Fidelity" standard established by an industry group called the Wireless Ethernet Compatibility Alliance (WECA). This standard is commonly referred to as the "WiFi" standard. For the purposes of ensuring interconnectivity, the ECHONET Specification is based on this standard.

The WiFi Standard on which this ECHONET Specification is based is the "WiFi System Interoperability Test Plan Version 1.1a" (published on December 11, 2001).

Should future changes to the WiFi standard cause an interconnectivity problem in communications based on the ECHONET Specification, other standards will be considered for use for the purposes of ensuring interconnectivity.

- IEEE802.11/11b-compliant off-the-shelf communication devices installed in or on full ECHONET devices, ECHONET device adapters and ECHONET middleware adapters shall be WiFi-compliant. However, this requirement does not apply in the case where the off-the-shelf communication device in question cannot be detached from an

ECHONET communications processing section.

- The network search method shall be WiFi-compliant, regardless of whether an IEEE802.11/11b-compliant off-the-shelf communication device is used or not. That is, STA's shall transmit beacon frames in a distributed and autonomous manner in the case of an ad-hoc mode network, and only the AP shall transmit beacon frames in the case of an infrastructure mode network.

(5) Upper limit for the number of terminals

Due to the upper limit for the number of addresses in the ECHONET, the number of subnets must be 256 or less.

(6) Packet length

In the case of an IEEE802.3 network, ECHONET frames are accommodated in MAC frames together with the MAC header＋trailer (max18 bytes), LLC＋SNAP header (max8 bytes), IP header (IPv4, max24 bytes)＋UDP header (max8 bytes) and FCS (4 bytes).

Because the sum of these is 1518 bytes at the maximum, the maximum data length is "1518 － (18＋8＋24＋8＋4) = 1456 bytes," which means that the maximum packet length of 262 bytes for ECHONET frames can be supported without problem and that there is no need to use split ECHONET frames.

(7) Upper limit for the length of time used to receive response packets

The length of time that can be used to receive response packets after transmitting a packet to a node differs depending on various factors and hence between systems and between states. This is particularly the case when an intermediary bridge is present, because factors such as the bridge performance, the processing speeds of the nodes in the subnet that has the bridge and the total number of nodes affect the length of time that can be used to receive response packets. This version of the ECHONET Specification specifies, taking into consideration the various factors that affect the length of time that can be used to receive response packets and the need to ensure interconnectivity, a fixed common upper limit for the length of time used to receive response packets, which also applies to other media including Bluetooth® and IEEE802.11/11b media. Additional requirements such as requirements relating to the method to dynamically determine the upper limit shall be defined in succeeding versions of the ECHONET Specification as necessary.

9.1.3 Applicable standards

When using an 802.11/11b network, the requirements specified in the applicable sections of the following standards shall be satisfied:

- IEEE Std. 802 "Overview and Architecture"
- ANSI/IEEE Std. 802.2 "Logical Link Control" (ISO/IEC 8802-2)
- ANSI/IEEE Std. 802.11 "Wireless LAN Medium Access Control (MAC) and

Physical Layer (PHY) Specifications" (ISO/IEC 8802-11)

- ANSI/IEEE 802.11b "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications － Higher-Speed Physical Layer Extension in the 2.4GHz Band"

The infrared system (Infrared (IR) PHY specification) and the frequency hopping system (Frequency-hopping spread spectrum (FHSS) PHY specification for the 2.4GHz Industrial, Scientific and Medical (ISM) band) for the physical layer in the IEEE Std. 802.11 are beyond the scope of this chapter.

In addition, the following standards shall apply in Japan:

- RCR STD-33/ARIB STD-T66

9.1.4 Scope of this chapter

This ECHONET Specification defines the specifications for the interface between the ECHONET Communication Middleware and IEEE802.11 networks and between the ECHONET Communication Middleware and the UDP/IP layers (see Fig. 9-1). The detailed mechanical, physical and electrical specifications for IEEE802.11/11b networks and the UDP/IP layers shall be as defined in the relevant standards, as is the case with Chapter 8.

9. 2 Mechanical and Physical Requirements

The mechanical and physical requirements shall be the same as the applicable requirements of RCR STD-33/ARIB STD-T66.

9. 3 Electrical Requirements

The electrical requirements shall be the same as the applicable requirements of RCR STD-33/ARIB STD-T66.

9.3.1 Transmission method and transmission signals

(1) Radio wave type

G1D

G: Method to modulate the main carrier = phase modulation

1: Type of the signal used to modulate the main carrier = a digital signal that does not use a sub carrier (single channel)

D: Transmission information type = data transmission, remote measurements, remote control

(2) Power output

Minimum output: 1mW

Maximum output: 10mW/MHz

(3) Communication system

Direct sequence spectrum spread (DS-SS) system

(4) Modulation speeds and modulation systems

The following modulation speeds and modulation systems are used (phase modulation is used as the base technique):

1Mbps: DBPSK (differential binary phase shift keying)

2Mbps: DQPSK (differential quadrature phase shift keying)

5.5Mbps/11Mbps: CCK (complementary code keying)

(5) Reception sensitivity

The frame error rate (FER) shall be 8×10^{-2} or less (−76dBm antenna input conversion) (for the 11Mbps CCK modulation condition).

9.3.2 Frequency

(1) Operating frequency band

2400 to 2497MHz

Of the 2.4 GHz ISM (Industrial Scientific Medical) Band that can be used without a license, the 2.400–2.497 GHz band (in which devices can be used as “radio stations for wireless LAN systems” shall be used.

(2) Communication channels

The IEEE802.11b standard defines 14 channels. Table 9.1 shows the central frequencies of these channels. Each channel shall use the frequency band between the central frequency minus 11MHz and the central frequency plus 11MHz.

In the ECHONET Specification, the channels 1 to 11 in Table 9-1, which are the mandatory channels in WiFi, are the mandatory channels and the channels 12 to 14 in Table 9-1 are optional.

Table 9-1 Central Frequencies of the Communication Channels

Channel	Central frequency (MHz)
1	2412
2	2417

3	2422
4	2427
5	2432
6	2437
7	2442
8	2447
9	2452
10	2457
11	2462
12	2467
13	2472
14	2484

9.4 Overview of the Logical Specifications

ECHONET packets are first UDP/IP encapsulated as with the logical specifications for the Ethernet described in Section 8.4, and then encapsulated using the protocol specified by the IEEE802.11/11b standard, before being sent out to the transmission channel.

The address requirements are the same as the address requirements specified in Section 7.4. The destination address (source address) of an ECHONET transmission frame shall be an ECHONET address (EA), and the destination address (source address) of an IP header shall be an IP address.

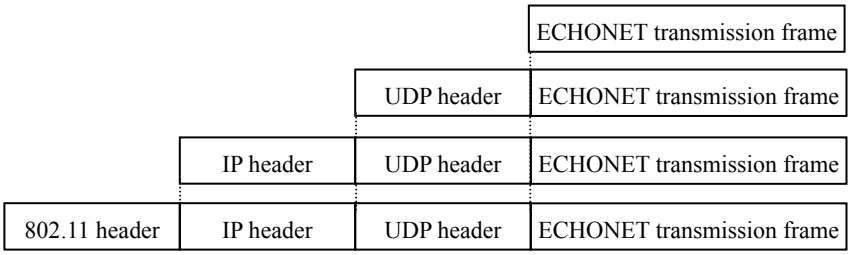


Fig. 9-6 Encapsulation of ECHONET Transmission Frame

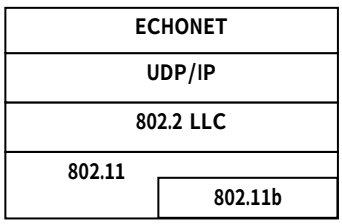


Fig. 9-7 IEEE802.11/11b Layers

9.5 Logical Specifications (IEEE802.11/11b network layer)

The IEEE802.11 and IEEE802.11b standards only specify the requirements for the physical and MAC layers, and the higher layers are shared with wired LAN. The basic format for MAC frames is as follows:

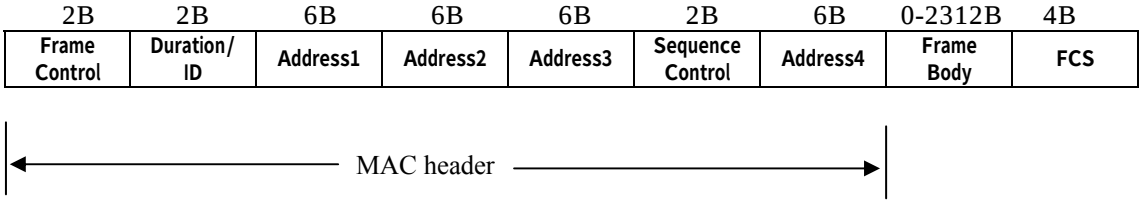


Fig. 9-8 IEEE802.11/11b MAC Frame

Frame control (Frame Control): Contains control information such as the protocol version number, the frame type, “more fragments” information, retransmission identification information, power management information, “more data” information and information on whether encryption is implemented or not.

Duration ID (Duration/ID): The estimated length of time the radio circuit will be used.

Address fields (Address 1 to Address 4: Up to 4 address fields are provided. The number of address fields varies depending on the frame type.)

Sequence control (Sequence Control): Indicates the MAC frame sequence number and the fragment number.

Frame body (Frame Body): Stores the data to be transmitted.

FCS (Frame Check Sequence): The error detection signal for the frame body and the MAC header.

IEEE802.11 MAC frames are basically classified into the following 3 categories:

1) Management frames

2) Control frames

3) Data frames

The management frame types that have been defined are as follows:

* Beacon frames, which are used to notify the presence of access points
* Authentication frames
* Association frames, which are used for information exchange between access points and terminals

The control frame types that have been defined are as follows:

* ACK frames, which are used to return acknowledgement responses
* RTS and CTS frames, which are used for the RTS/CTS control to prevent the “hidden terminal” problem

Data frames are defined as frames to transfer user data.

The IEEE802.11 standard defines authentication- and privacy protection-related services so

that functions equivalent to the inherent functions of wired LAN can be provided. These services allow a level of reliability to be achieved which is equivalent to the level of reliability of wired LAN in terms of shielding.

9.6 Logical Specifications (UDP/IP Layer)

The logical specifications for the UDP/IP Layer shall be as specified in Section 7.6 and Section8.6.

9.7 Logical Specifications (ECHONET/IP Layer)

The logical specifications for the ECHONET/IP Layer shall be as specified in Section 7.7, except that the following values shall be used for UDP packets:

Hardware type	IEEE802.11	6
Hardware address Ethernet 48-bit	IEEE802.3	address

9.7.1 Time requirements

The time requirements specified in Section 7.7.6 shall apply, except that the T1 value in "T7 = ECHONET MAC address value x T1 + T0" must be 5 ms.

9. 8 Basic Sequences

This section outlines the sequences for the following states of IEEE802.11/11b lower-layer communication software:

1) Stop
2) Initialization processing in progress
3) Communication stop
4) Normal operation
5) Error stop
6) Suspension

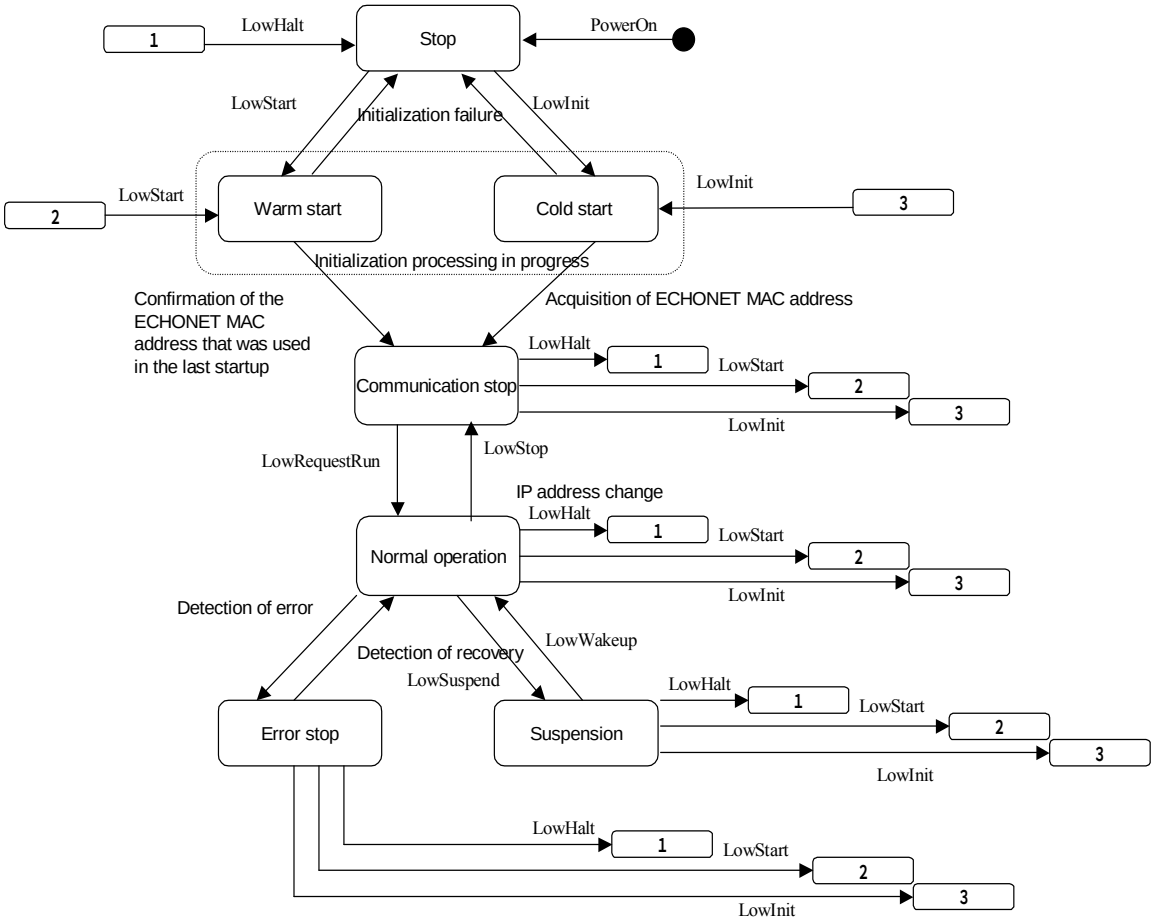


Fig. 9-9 State Change Diagram

9.8.1 "Stop" status

The "stop" status is a state in which the lower-layer communication software has stopped operating and all data except the ECHONET MAC address has been initialized.

Whenever the power is turned on, this state will be entered. An overview of the processing to be performed immediately after the state change is given below together with a list of the individual lower-layer communication interface services that can be accepted during the "stop" state with brief explanations about the processing relating to the services.

(1) Trigger and the response behavior

Waits for an individual lower-layer communication interface service.

All but the lower-layer communication software is initialized immediately after Power On.

(2) Status acquisition service (LowGetStatus)

Returns LOW_STS_STOP as the status.

(3) Lower-layer communication software type acquisition service (LowGetDevID)

Returns the type of lower-layer communication software.

The state change triggers are as follows:

(1) Trigger for a state change to the "initialization processing in progress" state:

Initialization request service (LowInit), warm start request service (LowStart)

9.8.2 Initialization Processing in Progress" status

The "initialization processing in progress" status is a state in which addresses are being acquired. An overview of the processing to be performed immediately after the state change is given below together with a list of the individual lower-layer communication interface services that can be accepted during the "initialization processing in progress" state with brief explanations about the processing relating to the services.

(1) Trigger and the response behavior

An IP address is acquired first and then an ECHONET MAC address is acquired, as instructed by LowStart/LowInit from the Communication Middleware. The warm start mode is a mode in which the acquisition process is started using the stored ECHONET MAC address, and the cold start mode is a mode in which a new ECHONET MAC address is acquired after discarding the stored ECHONET MAC address. If it is discovered during a warm start that the stored ECHONET MAC address is already being used by another node, the processing to acquire a new ECHONET MAC address will be started automatically. This ECHONET Specification does not specify requirements

relating to functions to reacquire IP addresses during warm and cold starts. If any other abnormal condition occurs, a state change to the “stop” state will be made.

(2) Status acquisition service (LowGetStatus)

Returns, as the status, LOW_STS_INI in the case of a cold start and LOW_STS_RST in the case of a warm start.

(3) Lower-layer communication software type acquisition service (LowGetDevID)

Returns the type of lower-layer communication software.

The state change triggers are as follows:

(1) Trigger for a state change to the “communication stop” state

Completion of acquisition of an ECHONET MAC address

(2) Trigger for a state change to the “stop” state

Failure to acquire an IP or ECHONET MAC address

9.8.3 “Communication Stop” status

The “communication stop” status is a state in which an operation start request from the Communication Middleware is being waited for after completion of the initialization of the lower-layer communication software. An overview of the processing to be performed immediately after the state change is given below together with a list of the individual lower-layer communication interface services that can be accepted during the “communication stop” state with brief explanations of the processing relating to the services.

(1) Trigger and the response behavior

Waits for an individual lower-layer communication interface service.

(2) Status acquisition service (LowGetStatus)

Returns LOW_STS_CSTOP as the status.

(3) Physical address acquisition service (LowGetAddress)

Returns the ECHONET MAC address.

(4) Profile data acquisition service (LowGetProData)

Returns the profile data.

(5) Lower-layer communication software type acquisition service (LowGetDevID)

Returns the type of lower-layer communication software.

The state change triggers are as follows:

(1) Trigger for a state change to the “normal operation” state

Operation start instruction service (LowRequestRun)

(2) Trigger for a state change to the “initialization processing in progress” state

Initialization request service (LowInit), warm start request service (LowStart)

(3) Trigger for a state change to the “stop” state

Stop service (LowHalt)

9.8.4 “Normal Operation” status

The “normal operation” status is a state in which a message is being transmitted to or received from the transmission medium (i.e. a state in which the primary function of the lower-layer communication software is being performed). An overview of the processing to be performed immediately after the state change is given below together with a list of the individual lower-layer communication interface services that can be accepted during the “normal operation” state with brief explanations about the processing relating to the services.

(1) Trigger and the response behavior

Waits for an individual lower-layer communication interface service.

(2) Status acquisition service (LowGetStatus)

Returns LOW_STS_RUN as the status.

(3) Physical address acquisition service (LowGetAddress)

Returns the ECHONET MAC address

(4) Profile data acquisition service (LowGetProData)

Returns the profile data.

(5) Message transmission service (LowSendData)

Converts the provided protocol difference absorption processing section message into a lower-layer communication software message and outputs it to the transmission medium (The message will be split into two more split messages before the conversion when the original message size requires splitting).

(6) Message reception service (LowRecvData)

Converts the lower-layer communication software message received from the transmission medium into a protocol difference absorption processing section message and outputs it to the protocol difference absorption processing section.

(7) Lower-layer communication software type acquisition service (LowGetDevID)

Returns the type of lower-layer communication software.

The state change triggers are as follows:

(1) Trigger for a state change to the “suspension” state

Lower-layer communication section stop service (LowSuspend)

(2) Trigger for a state change to the “stop” state

Stop service (LowHalt) or an IP address change in the IP layer

(3) Trigger for a state change to the “error stop” state

An error (lower-layer communication software)

(4) Trigger for a state change to the “initialization processing in progress” state

Initialization request service (LowInit), warm start request service (LowStart)

(5) Trigger for a state change to the “communication stop” state

Stop service (LowStop)

9.8.5 “Error Stop” status

The “error stop” status is a state in which the operation of the software has been stopped as a result of an error. An overview of the processing to be performed immediately after the state change is given below together with a list of the individual lower-layer communication interface services that can be accepted during the “error stop” state with brief explanations about the processing relating to the services.

(1) Trigger and the response behavior

Performs error processing. Any message being received will be discarded and any new or outstanding message transmission request will be rejected and an error will be returned, before the state change to the “error stop” state becomes effective.

(2) Status acquisition service (LowGetStatus)

Returns LOW_STS_ESTOP as the status.

(3) Lower-layer communication software type acquisition service (LowGetDevID)
Returns the type of lower-layer communication software.

The state change triggers are as follows:
(1) Trigger for a state change to the “stop” state
Stop service (LowHalt)

(2) Trigger for a state change to the “initialization processing in progress” state
Initialization request service (LowInit), warm start request service (LowStart)

(3) Trigger for a state change to the “normal operation” state
Removal of the cause of the error

9.8.6 “Suspension” status

The “suspension” status is a state in which the operation of the software has been temporarily stopped in response to an instruction from the communication middleware. An overview of the processing to be performed immediately after the state change is given below together with a list of the individual lower-layer communication interface services that can be accepted during the “suspension” state with brief explanations about the processing relating to the services.

(1) Trigger and the response behavior
Stops the operation of the lower-layer communication software.
Any message being received will be discarded and any new or outstanding message transmission request will be rejected and an error will be returned.

(2) Status acquisition service (LowGetStatus)
Returns LOW_STS_SPD as the status.

(3) Lower-layer communication software type acquisition service (LowGetDevID)
Returns the type of lower-layer communication software.

The state change triggers are as follows:
(1) Trigger for a state change to the “normal operation” state
Operation restart service (LowWakeUp)
The lower-layer communication software will restart its transmission and reception functions immediately.

(2) Trigger for a state change to the “stop” state

Stop service (LowHalt)

(3) Trigger for a state change to the "initialization processing in progress" state

Initialization request service (LowInit), warm start request service (LowStart)

9.9 Accommodation Requirements

9.9.1 ECHONET MAC address servers

ECHONET MAC address server devices as described in Section 7.9.1 (5) shall be deemed to qualify as ECHONET MAC address server devices for IEEE802.11. For the specific requirements, refer to Part 7.

9.9.2 Layer management function accommodation requirements

The IEEE802.11 standard defines, in addition to physical and MAC layer protocol requirements, requirements for layer management functions for each layer.

Because these are contained within the ECHONET lower layer communication software, control from ECHONET applications is only possible through individual lower-layer communication interfaces.

This section defines the requirements for the attributes (listed below) that should be notified to IEEE802.11/11b communication media implemented in ECHONET devices as well as the requirements relating to the timing to change the attribute values.

1. Management of the belonging relationship between terminal and base stations
2. Authentication and encryption of terminal stations
3. Handoff

(1) Belonging relationship between terminal stations (STA's) and base stations (AP's) radio LAN networks

In radio LAN communications based on the IEEE802.11 standard, networks are classified into 2 types; "infrastructure mode networks" and "ad-hoc mode networks." Wireless distribution systems, which consist of AP's only, are also used, but this is a packet-relaying function for base stations only and does not represent an additional communication mode.

In the ECHONET, the selection between the ad-hoc and infrastructure modes shall be made at the time of the initialization of the lower-layer communication medium.

Mode selection shall be achieved by means of the reading (during the initialization processing) of the initialization parameters by the individual lower-layer communication software upon an initialization request from the ECHONET application.

(2) Authentication and encryption of terminal stations

The IEEE802.11 standard defines the 2 types of authentication listed below. In the WiFi standard, open system authentication is mandatory, but shared key authentication is implementation-dependent.

(A) Shared key authentication

Challenge-response-based authentication. An encryption key is set in advance in both the base station (AP) and terminal stations (STA's).

(B) Open system authentication

Because all authentication requests will be granted, the use of this authentication method is practically the same as not performing authentication at all.

In the ECHONET, the authentication method selection (i.e. between the shared key authentication and open system authentication described above), the size of the shared key used for shared key authentication and the shared key itself shall be provided from the individual lower-layer communication interface during the initialization of the lower-layer communication medium.

(3) Handoff

Handoff is a function that allows terminal stations (STA's) of an infrastructure mode network (ESS) having two or more base stations (AP's) to automatically detect each of their positional changes from an AP's radio cell to another AP's radio cell and switch to the AP for the destination radio cell.

Because the IEEE802.11 standard does not define requirements for the implementation method, the ECHONET Specification does not provide specific requirements for interfaces used during initialization.

9.9.3 Initialization parameters

Medium-specific initialization data shall be provided at the time of the initialization request (individual lower-layer interface) and shall include at least the following attribute values:

・ Pointers to the transmission and reception buffers of the communication medium used by the common lower-layer communication interface.

・ Values of the IEEE802.11/11b attributes defined in "9.9.2 Layer management function accommodation requirements."

This ECHONET Specification does not specify the method to create, update or delete the initialization parameters.

The initialization parameters may be fixed values or may be rewritten by the lower-layer communication software through utilities.

Table 9-2 shows the attributes that the IEEE802.11/11b initialization parameters must include at the minimum. For the initialization parameter structures for the individual lower-layer interfaces for the C language, refer to "Section 4.3 Initial data setting specifications" of Part 6.

Table 9-2 IEEE802.11/11b Initialization Parameters

No.	Name	Function
1	Transmission buffer size	The size of the transmission buffer of the communication medium.
2	Transmission buffer	Communication medium's transmission buffer.
3	Reception buffer size	The size of the reception buffer of the communication medium.
4	Reception buffer	Communication medium's reception buffer
5	IP address setting method	Specifies whether the IP address is set automatically or manually.
6	Communication mode	Specifies whether the ad-hoc or infrastructure mode is used.
7	SSID	Service set identifier.
8	Authentication method	Specifies whether open system authentication or shared key authentication is used.
9	Authentication key size	Specifies the size of the authentication key.
10	Authentication key	The authentication key that is set.
11	Communication channel	The communication channel used by the communication medium.

9.9.4 Lower-layer communication software initialization data notification requirements

As explained in the previous section, how the initialization data provided to the lower-layer communication software is handled is implementation-dependent and is not specified by this ECHONET Specification.

However, a means to allow users to notify this data is necessary because most ECHONET devices except certain controllers do not have a user interface. This section specifies the requirements for methods of notifying the initialization data before an ECHONET device equipped with an IEEE802.11/11b medium joins the ECHONET domain.

(1) Initialization data notification methods

ECHONET devices equipped with an IEEE802.11/11b medium shall be implemented with either of the following IEEE802.11/11b initialization data notification methods:

A) Simplified web server-based notification

B) UDP/IP packet transmission-based notification

For simplified web server-based notification, this ECHONET Specification does not specify any specific format for data transmission. However, ECHONET devices implemented with this method must be capable of providing an HTTP-based initialization data setting screen for external devices.

ECHONET devices implemented with the UDP/IP packet transmission-based notification method shall be capable of receiving initialization data using a UDP frame that meets the requirements specified below with an IP-type network connected.

(2) Communication conditions for initialization data exchanges

ECHONET devices equipped with an IEEE802.11/11b medium shall exchange initialization data under the following communication conditions:

Table 9-3 Communication Conditions for Initialization Data Exchanges

		Simplified web server-based notification	UDP/IP packet-based notification	
			Wired	Radio
Prescribed UDP exchange functions		Not necessary	Compulsory	Compulsory
Initialization data reception device's IP address (Newly joining ECHONET device)		192.168.0.250 (guideline)	169.254.1.0 to 169.254.254.255 Any link-local address	
Initialization data transmission device's IP address (Existing ECHONET device or external device)		Class C 192.168.0.0 Any address within the network	Same as above, except that the ECHONET multicast address reception function is compulsory.	
Radio LAN settings	SSID	ECHONET	Not necessary	ECHONET
	Communication mode	Ad-hoc	Not necessary	Ad-hoc
	Authentication method	Open authentication	Not necessary	Open authentication
	WEP key	None	Not necessary	None
	Communication channel	1	Not necessary	1
ECHONET MAC address acquisition		M-MODE or A-MODE		

* The term "link-local address" here refers to an IPv4-based address.
(Reference: Dynamic Configuration of IPv4 link-local addresses)

The term "initialization data reception device" used in the table above refers to an ECHONET device equipped with an IEEE802.11/11b medium which is attempting to newly join the ECHONET domain, and the term "initialization data transmission device"

refers to an existing ECHONET device or an external device. An “external device” is a non-ECHONET device that is capable of performing communication using an IEEE802.11/11b medium (such as a radio LAN-capable PC).

In the case where the simplified web server-based notification method is used, the initial values of initialization data reception devices shall be a published fixed IP address (it is recommended that the guideline value of 192.168.0.250 be used), and initialization data transmission devices shall be equipped with a means to access such addresses, reference web pages provided by initialization data reception devices and notify the specified initialization data.

When using radio LAN as the means of communication between devices, the communication conditions shown in the table above shall be used.

In the case where the UDP/IP packet transmission-based notification method is used, the IP addresses of initialization data reception devices shall be any addresses within the network between 169.254.1.0 and 169.254.254.255. This address space is called the “IPv4 Link-Local Address Space” and is usable only with a single local network segment which is not connected to the outside.

When transmitting or receiving initialization data, each device shall determine the address that will be used as the provisional address within this address space. This ECHONET Specification does not specify the method of determining the provisional IP address, but it is recommended that the Address Resolution Protocol (ARP) be used to check to confirm that the determined provisional IP address is not already being used.

Initialization data reception devices shall multicast an initialization data packet that meets the requirements specified below to request initialization data transmission devices located in the same network to provide the initialization data.

The ECHONET Specification requires “224.0.23.0” be used as the multicast address for such a multicast. Initialization data transmission devices must be capable of recognizing this address as an ECHONET multicast address.

When using radio LAN as the means of communication between devices, the communication conditions shown in the table above shall be used.

(3) Initialization data notification packet types

The setting up of the IEEE802.11/11b initialization data shall be done via network by connecting to a device having the initialization data under the communication conditions described above.

UDP/IP packet-based notifications shall use the packet frame structure specified below. The format shall be the packet format specified in Section 7.7, the version number shall be “0x01” and packet type number assignment shall be from No.32.

Table 9-4 Initialization data notification packet types

Packet type number	Packet type	Whether or not support is required
32	Initialization data request	Support is required in the case of UDP/IP packet-based notification
33	Initialization data response	

(4) Composition of initialization data notification packets

【Packet type No. 32】Initialization data request

An initialization data request packet is a packet that is multicast into the ECHONET domain under the conditions specified in Table 9-3 by a newly joining ECHONET device equipped with an IEEE802.11/11b medium.

The frame structure shall be as follows:

Table 9-5 Frame Structure of Initialization Data Request Packets

Item	Size	Explanation
Version	1	The value “0x01” shall be entered (Version 1).
Type	1	The value “0x20” shall be entered.
HAddr	6	Hardware address of the newly joining node.

The initialization data reception node shall send an initialization data request packet to the address “224.0.23.0” using the UDP protocol after the preparations for setting up the IEEE802.11/11b connection conditions have been completed in the lower-layer communication software, and wait for the “initialization data response” packet described below.

“Version” indicates the packet frame version number. The value shall always be “0x01.”

“Type” indicates the packet type. In the case of an initialization data request, the value “0x20” shall be assigned.

“HAddr” indicates the hardware address of the newly joining device, that is, the initialization data reception node (network byte order).

【Packet type No.33】Initialization data response

An initialization data response packet is a packet that is sent by an initialization data transmission node in responses to a Type 32 packet received from an initialization data reception node.

The frame structure shall be as follows:

Table 9-6 Frame Structure of Initialization Data Response Packets

Item	Size	Explanation
Version	1	The value “0x01” shall be entered (Version 1).

Type	1	The value “0x21” shall be entered.
HAddr	6	Hardware address of the newly joining node.
IP Method	1	IP address setting method: IP address distribution function is provided (automatic mode): 0x00 IP address distribution function is not provided (manual mode): 0x01
IP Address	4/0	Fixed IP address. This shall be omitted when IP Method = “0x00” (size = 0).
IP Netmask	4/0	Netmask for use in the case where a fixed IP address is used. This shall be omitted when IP Method = “0x00” (size = 0).
IP Broadcast	4/0	Broadcast address for use in the case where a fixed IP address is used. This shall be omitted when IP Method = “0x00” (size = 0).
WLAN Mode	1	Radio LAN communication mode: Ad-hoc mode: 0x00 Infrastructure mode: 0x01
SSID Length	1	Size of SSID (M)
SSID	M/0	Contains SSID (byte array). This shall be omitted when SSID Length = “0” (size = 0).
Auth Method	1	Authentication method: Open system authentication: 0x00 Shared key authentication: 0x01
WEP Length	1	Size of the authentication key (N).
WEP Key	N/0	Contains the authentication key (byte array). This shall be omitted when WEP Length = “0” (size = 0).
Channel	1	Communication channel.

“Version” indicates the packet frame version number. The value shall always be “0x01.”

“Type” indicates the packet type. In the case of an initialization data response, the value shall be “0x21.”

“HAddr” indicates the hardware address (6 bytes) of the newly joining device, that is, the initialization data reception node (network byte order).

HAddr is followed by the initialization data needed by the lower-layer communication software for the IEEE802.11/11b media:

IP Method indicates the IP address setting method.

IP-type ECHONET nodes are required to have the DHCP client function. However, with regard to operation, the DHCP service function is a recommended function. For this reason, there can be systems in which DHCP-based IP address distribution is not

available.

Therefore, if an IP type ECHONET node whose only means of setting the IP address is the use of a DHCP client joins a system that does not provide the DHCP service, there will be no means of acquiring an IP address.

For this reason, it is permitted that an IP address distribution function be implemented in initialization data transmission nodes and external devices. The value “0x00” shall be assigned in the case where it is necessary to automatically acquire an IP address using DHCP (“automatic mode”), and the value “0x01” shall be assigned in the case where the user sets the IP address manually through the use of an initialization data transmission node or external device (“manual mode”).

IP Address indicates, when the value of IP Method (IP address setting method) is “0x01” (= manual mode), the IP address to be assigned to the initialization data reception node, in the form of a 4-byte array in network byte order.

IP Netmask indicates, when the value of IP Method (IP address setting method) is “0x01” (= manual mode), the netmask for the fixed IP address to be assigned to the initialization data reception node, in the form of a 4-byte array in network byte order.

IP Broadcast indicates, when the value of IP Method (IP address setting method) is “0x01” (= manual mode), the broadcast IP address to be assigned to the initialization data reception node, in the form of a 4-byte array in network byte order.

When the value of IP Method is “0x00,” IP Address, IP Netmask and IP Broadcast are omitted.

It shall be possible to set the IP method, IP Address, IP Netmask and IP Broadcast settings regardless of whether the system has a DHCP service or not.

An IP address assignment function is a function that allows existing ECHONET devices and external devices to manage the IP addresses of IP type ECHONET devices and distribute unique IP addresses to newly joining ECHONET devices.

This function shall be provided in external devices in the form of a separate software program, or, alternatively, the IEEE802.11/11b initialization data setup service object (this is separately defined as an ECHONET object) shall be used.

When the value of IP Method is “0x00” (= automatic mode), the DHCP service must be in operation in the systems in operation and the initialization data reception node(s) must acquire an IP address using DHCP under the acquired communication conditions.

WLAN Mode indicates the IEEE802.11/11b communication mode. The value “0x00” corresponds to the ad-hoc mode, and the value “0x01” corresponds to the infrastructure mode.

SSID Length indicates the data size, excluding the end NULL, of the radio LAN service set identifier.

SSID indicates the radio LAN service set identifier in the form of an array in network byte order. In the case where the service set identifier is specified in ASCII code characters, the last NULL (byte code 0) is removed. When the value of SSID Length is 0, SSID is omitted.

Auth Method indicates the radio LAN authentication method in the form of a 1-byte value. The value “0x00” shall be assigned in the case where the authentication method for the ECHONET systems equipped with IEEE802.11/11b media in operation is open system authentication, and the value “0x01” shall be assigned in the case where the authentication method for the ECHONET systems equipped with IEEE802.11/11b media in operation is shared key authentication.

WEP Length indicates the size of the authentication key.

WEP Key gives the system authentication key in the form of an array in network byte order.

When the WEP Length value is 0, WEP Key is omitted.

Channel indicates the communication channel for IEEE802.11/11b adopted by the system in the form of a 1-byte value. For details of Channel, refer to (2) of “9.3.2 Frequency.”

(5) Preconditions for UDP/IP packet transmission-based notification

There is a precondition for the use of the UDP/IP packet-based initialization data notification method that there must be one or more existing ECHONET node or external device in operation which can serve as an “initialization data transmission node.”

For an existing ECHONET node to have this function, it must be equipped with a user interface to setup the IEEE802.11/11b initialization data as well as a means to notify the initialization data to newly joining ECHONET nodes according to the initialization data notification sequence described in the next section.

Since this function is separately defined as the “IEEE802.11/11b initialization service object,” this section does not define any detailed requirement for this function.

For an external device to have this function, it must be equipped with one or more IEEE802.11/11b communication medium as well as software to provide newly joining ECHONET nodes with the initialization data in accordance with the initialization data notification sequence.

Since this software is deemed to be part of the individual lower-layer communication software, the ECHONET Specification does not specify any requirement for it.

Regardless of whether the initialization data notification method is “simplified web server-based notification” or “UDP/IP packet transmission-based notification,” the communication conditions specified in “2) Communication conditions for initialization data exchanges” shall always apply.

Even in the case where IEEE802.11/11b media-based ECHONET systems are already in operation, the communication settings of existing ECHONET nodes and external devices must be temporarily switched to the above-mentioned “communication conditions for initialization data exchanges” whenever they serve as an “initialization data transmission node,” so that the vendor dependence during the initialization of ECHONET devices

equipped with IEEE802.11/11b media is eliminated.

(6) Initialization data notification sequence

The figure below shows the initialization data notification sequence.

This sequence is not necessary in the case where the initialization data notification method for ECHONET nodes equipped with IEEE802.11/11b media is "simplified web server-based notification," but is necessary in the case where the initialization data notification method is "UDP/IP packet transmission-based notification."

The steps of the sequence are explained in detail below in the order they occur.

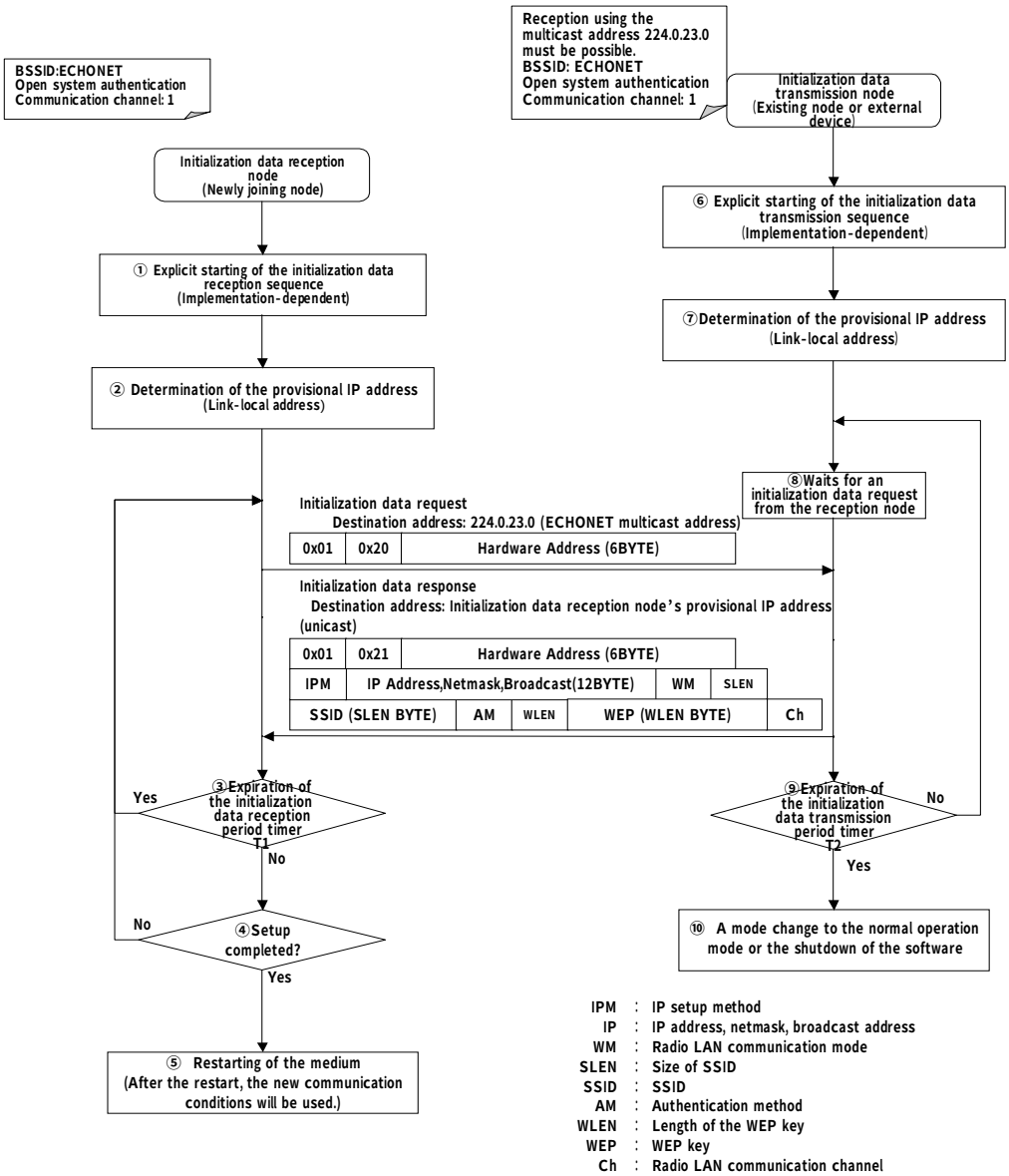


Fig. 9-10 Initialization Data Notification Sequence

The left half of the figure represents an initialization data reception node, that is, an

ECHONET node attempting to newly join the ECHONET domain. The right half represents an initialization data transmission node, that is, an existing ECHONET node or an external device that can provide the newly joining node with the initialization data.

① Explicit starting of the initialization data reception sequence

The "initialization data reception sequence" of a newly joining ECHONET node shall be explicitly started by the user. That is, this sequence shall only be started by a deliberate action of the user or installer. This sequence shall not be allowed to start when the user does not intend to start it.

The method of explicitly starting the initialization data reception sequence is implementation-dependent and is not specified by this ECHONET Specification.

(For example, the initialization data reception sequence could be started by means of an initialization data reception sequence start switch installed on the ECHONET device or of a hidden command that is activated through the use of the reset button and power switch in combination.)

② Determination of the provisional IP address of the newly joining ECHONET node

Device IP addresses that are used during initialization may be determined freely within the Link-Local Address Space. The Link-Local Address Space is an address space that is usable only with a single local network segment which is not connected to the outside. The effective range is between 169.254.1.0 and 169.254.254.255. The method of IP address determination is implementation-dependent, but it is recommended that a search for unused IP addresses be made using an ARP request, etc. (One possible approach is to use the APIPA (Automatic Private IP Addressing) technique.)

③ Expiration of the initialization data reception period timer

A newly joining ECHONET node that has started its initialization data reception sequence shall multicast a Packet Type No.32 initialization data request packet into the domain and wait for a response from the initialization data transmission nodes that are presumed to be present in the domain.

If a certain period of time (the T1 period) expires with no response received, the newly joining ECHONET node shall repeat the transmission of the Packet Type No.32 initialization data request packet until the user explicitly instructs that the sequence be terminated.

The method of explicitly instructing that the sequence be terminated is implementation-dependent and is not specified by this ECHONET Specification.

④ Completion of the initialization data reception sequence

If the newly joining ECHONET node receives a Packet Type No.33 initialization data response packet within the specified period of time (the T1 period) after transmission of the initialization data request packet, it shall analyze the response packet, take in the

necessary initialization data and then proceed to the "⑤ Restarting of the newly joining ECHONET node's medium" step.

If the attempt of the newly joining ECHONET node to acquire the initialization data fails, the newly joining ECHONET node shall resend the Packet Type No.32 initialization data request packet and repeat the sequence.

If the newly joining ECHONET node receives an explicit instruction to terminate the sequence, it shall proceed to the "⑤ Restarting of the newly joining ECHONET node's medium" step, regardless of whether the initialization data has been successfully acquired or not. The method of providing an explicit instruction to terminate the sequence is implementation-dependent and is not specified by this ECHONET Specification.

⑤ Restarting of the newly joining ECHONET node's medium

If the newly joining ECHONET node succeeds in acquiring the initialization data after receiving an initialization data response packet, it shall terminate the initialization data reception sequence and restart the communication medium using the acquired initialization data.

The acquired initialization data shall be stored in the newly joining device. This ECHONET Specification does not specify the length of time for which the acquired initialization data must be stored or the method of deleting the acquired initialization data, but at least it must be possible to start communication under the communication conditions specified in the acquired initialization data.

⑥ Explicit starting of the initialization data transmission sequence

The "initialization data transmission sequence" of the initialization data transmission node shall also be started by an explicit instruction of the user, as in the case of the initialization data reception node' initialization data reception sequence. That is, the initialization data transmission sequence shall not be allowed to start when the user does not intend to start it.

The method of explicitly starting the initialization data transmission sequence is implementation-dependent and is not specified by this ECHONET Specification.

⑦ Determination of the provisional IP address of the initialization data transmission node

The initialization data transmission node shall wait for an initialization data request packet in a network with a network IP address that is different from the address being used by the system.

The IP address may be determined freely within the Link-Local Address Space.

The method of IP address determination is implementation-dependent, but it is recommended that a search for unused IP addresses be made using an ARP request, etc.

⑧ Waiting for an initialization data request

After starting the initialization data transmission sequence, the initialization data transmission node shall wait for an initialization data request packet from the newly joining ECHONET node.

If the initialization data transmission node receives a Packet Type No. 32 initialization data request packet during the waiting period, it shall write the necessary data into a Packet Type No. 33 initialization data response packet and send it as a unicast packet to the newly joining node.

In the case where the initialization data transmission node has an IP address assignment function, the initialization data transmission node must be capable of presenting an IP address that is unique within the domain to the newly joining node.

Since IP address assignment functions are deemed to be part of the individual lower-layer communication software, the ECHONET Specification does not specify any requirement for them.

⑨ Expiration of the initialization data transmission period timer

If the initialization data transmission node does not receive an initialization data request packet within the specified period of time (the T2 period) after the initialization data transmission sequence was started and the initialization data transmission node started waiting for an initialization data request packet, it shall immediately switch to the normal operation mode or shut down the software.

The initialization data notification sequence should only be performed at the time of introduction of a device, and the initialization data transmission sequence should not be kept active for a long period of time.

This is because, while the initialization data notification sequence is being performed, any other device can transmit an initialization data request packet to obtain the radio LAN communication conditions at the time.

⑩ Termination of the initialization data transmission sequence

The initialization data transmission sequence shall be terminated when no initialization data request is received within the specified period of time (the T2 period) described above or when an explicit instruction is given by the user to terminate the sequence.

The method of providing an explicit instruction to terminate the sequence is implementation-dependent and is not specified by this ECHONET Specification.

The operation of the initialization data transmission period timer after an initialization data request is received within the specified period of time (the T2 period) and a response to that request is returned as specified shall be implementation-dependent. That is, the timer may reset itself and newly start measuring time towards the expiration of the T2 period, in which case the initialization data transmission node can wait for an initialization data request for the whole T2 period, or the timer may continue measuring time towards the expiration of the T2 period, in which case the time available for the initialization data transmission node to wait for an initialization data request before the

expiration is the T2 period minus the time elapsed by the time the initialization data transmission node returned the response.

(7) Expiration times of the initialization data reception and transmission period timers

The recommended expiration times of the timers used in the initialization data notification sequence (i.e. the initialization data reception and transmission period timers) are as shown in the table below. The values shown are guideline values, not requirements.

Table 9-7 Expiration times of the timers used in the initialization data notification sequence

Timer expiration time	Definition	Recommended value
T1	The period of time for which the initialization data reception node waits for an initialization data response after transmitting an initialization data request.	2 seconds
T2	The period of time for which the initialization data transmission node waits for an initialization data request after starting to wait for such a request. If the initialization data transmission node does not receive an initialization data request within this period of time, it will deem that no request has been made.	It is recommended that 5 minutes be used as the default expiration time and a function be provided that allows the default value to be changed, because longer expiration times may be necessary depending on the installation location.

Supplement 9.1 Scenarios for Starting up ECHONET Nodes Equipped with IEEE802.11/11b Media

ECHONET nodes equipped with IEEE802.1/11b communication media must go through the initialization sequence specified in the IEEE802.11 standard and the initialization sequence specified in the ECHONET Specification, before they can join a domain and start operating.

This section (Appendix 9.1) clarifies the scopes of the IEEE802.11 standard and ECHONET Specification in relation to the starting up of ECHONET nodes equipped with IEEE802.1/11b communication media by providing an overview of the operation of an ECHONET node equipped with an IEEE802.11/11b communication medium from the point of time the node is introduced to the point of time the node starts operating together with explanations about specific situations that may be encountered in the process based on the requirements and explanations given in Chapter 9.

This section provides 3 scenarios; the “scenario for the operation of the node from radio LAN startup to IP address determination,” “scenario for the startup and operation of the node in the ad-hoc and infrastructure modes” and “scenario for the operation of the node from IP address determination to ECHONET MAC address acquisition.”

1. Radio LAN startup

In the case of radio LAN cards etc. used on PC’s, the operating system installed on the PC manages the data necessary to start up radio LAN.

To use radio LAN, at least the following data must have been provided to the device before communication is started:

a. Communication mode

Whether the communication mode is the ad-hoc mode or the infrastructure mode

b. IP address setting method

Whether to acquire an IP address automatically (DHCP) or set the IP address manually

c. ECHONET MAC address acquisition method

Which of the A-MODE, SR-MODE and M-MODE to use to acquire an ECHONET MAC address

d. SSID

Service set identifier

e. Authentication method

Whether the authentication method is open system authentication or shared key authentication

f. Authentication key

WEP key

g. Communication channel

The channel to be used by the medium defined by the IEEE802.11/11b standard

In the case where radio LAN is used for ECHONET devices, a mechanism becomes necessary that allows the individual lower-layer communication software to independently manage the above-mentioned data, because there is no system that is equivalent to personal computer operating system.

The figure below shows the operation of the node from incorporation of the above-mentioned data into the individual lower-layer communication software via the individual lower-layer communication interface to radio LAN startup to IP address determination.

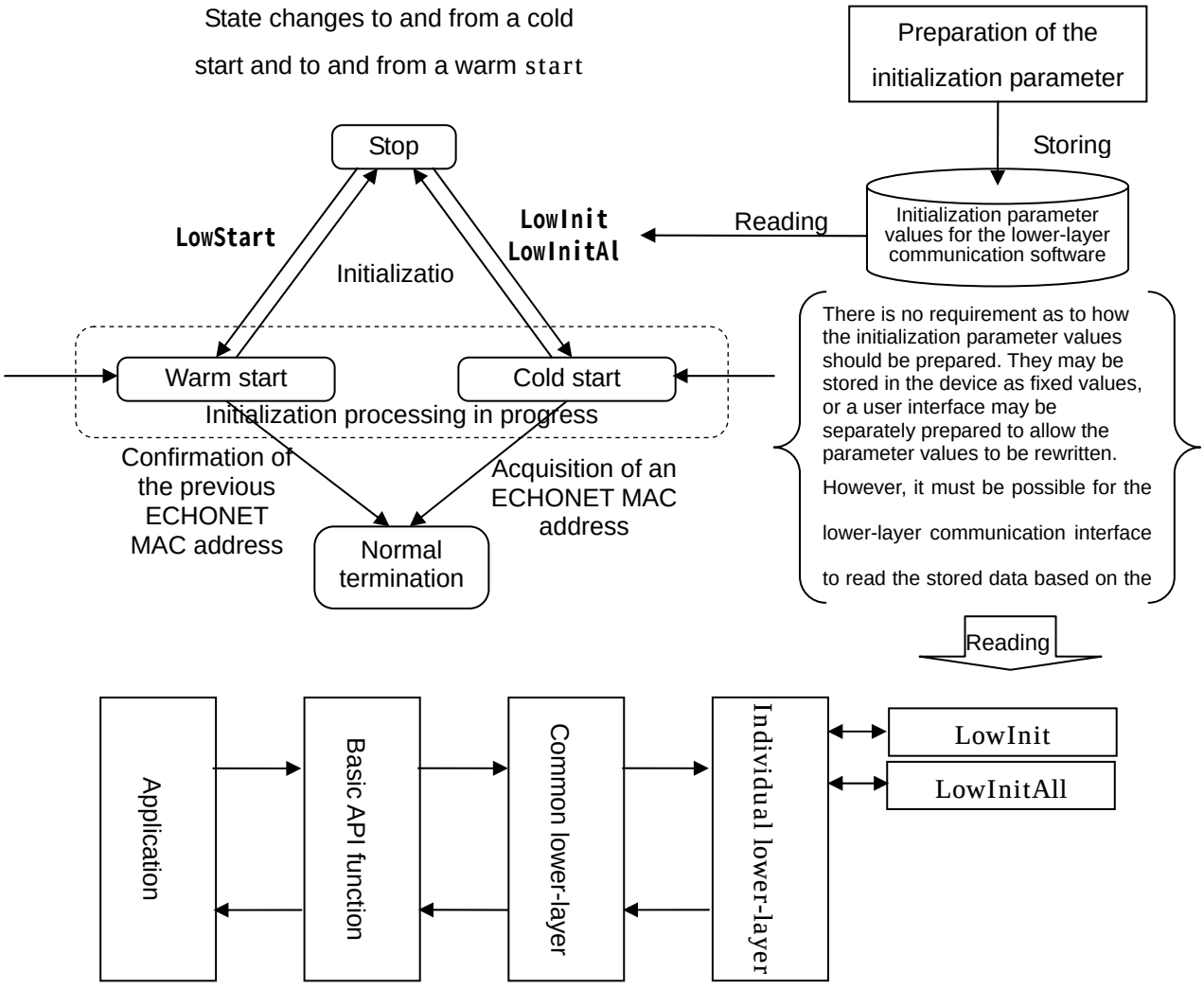


Fig. A9.1-1 Reading of the Initialization Parameter Values

As shown in the figure above, the settings unique to radio LAN are read into the communication driver during the initialization processing, to allow them to be used as part of the individual lower-layer communication software. Therefore, if, for example, the IEEE802.11/11b communication channel or authentication key is changed, the ECHONET device must call LowInit or LowInitAll and perform a cold start.

It will become necessary to call a cold start when any of the following occurs:

a. A communication mode change
b. An IP address change
c. An ECHONET MAC address change
d. A service set identifier change
e. An authentication method change
f. An authentication key change
g. A communication channel change

It will become necessary to call a warm start when, for example,

a. A device reenters a communication area after exiting it; or
b. A device in which settings are stored restarts.

From the standpoint of the IEEE802.11 standard, it will become necessary to call a warm start when a base station returns a re-association response (that gives a permission to connect) in response to a re-association request from a terminal station.

Brief explanations about the negotiation during a connection using management frames specified in the IEEE802.11 standard for the MAC layer are given below as reference information:

Management frames are classified into the following 3 types:

1) Beacon frames (information for notifications)
2) Authentication frames (authentication-related information)
3) Association frames (connection-related information)

① Notifications using beacon frames

In the infrastructure mode, base stations (AP's) transmit beacon frames periodically to notify terminal stations (STA's) of the presence of the radio cells. In the ad-hoc mode, terminal stations transmit beacon frames at random intervals for IBSS notification purposes.

The structure of beacon frames is as follows:

Fig. A9.1-2 Beacon Frames – Subtype Designation

The information element type is specified by the Subtype section of the frame control field. Beacon frames are assigned with the value "1000" (B4 to B7).

The Frame Body section is used to store the information elements shown in the table below.

Table A9.1-1 Information Elements of Beacon Frames

No.	Information element	Length [octets]]	Explanation
1	Time Stamp	8	Time synchronization timer value (μ sec.)
2	Beacon Interval	2	Beacon frame interval (K μ sec. [1024 μ sec.])
3	Capability Information	2	Information as to whether polling, encryption, etc. are to be performed
4	Service Set ID	2-34	ESS or IBSS identifier
5	Supported Rates	3-10	List of supported transmission rates
6	FH Parameter Set	7	Frequency hopping system-related parameters
7	DS Parameter Set	3	Direct frequency sequence spread system-related parameters (including channels)
8	CF Parameter Set	8	PCF (polling system)-related parameters
9	IBSS Parameter Set	4	ATIM window length (K μ sec.) used in the ad-hoc mode
10	TIM	6-256	Power management information

② Probe request and response frames

In addition to beacon frames, probe request and response frames can also be used. Probe request frames are frames used by terminal stations to ask base stations what radio cells are present, and probe response frames are frames used by base stations to provide responses to terminal stations' requests. Probe request frames and probe response frames are assigned with the Subtype values "0100" and "0101," respectively. The information elements of probe request frames and probe response frames are as shown in the tables below.

Table A9.1-2 Information Elements of Probe Request Frames

No.	Information element	Length [octets]]	Explanation
1	Capability Information	2	Information as to whether polling, encryption, etc. are to be performed
2	Service Set ID	2-34	ESS or IBSS identifier
3	Supported Rates	3-10	List of supported transmission rates

Table A9.1-3 Information Elements of Probe Response Frames

No.	Information element	Length [octets]]	Explanation
1	Time Stamp	8	Time synchronization timer value (μ sec.)
2	Beacon Interval	2	Beacon frame interval (K μ sec. [1024 μ sec.])
3	Capability Information	2	Information as to whether polling, encryption, etc. are to be performed
4	Service Set ID	2-34	ESS or IBSS identifier
5	Supported Rates	3-10	List of supported transmission rates
6	FH Parameter Set	7	Frequency hopping system-related parameters
7	DS Parameter Set	3	Direct frequency sequence spread system-related parameters (including channels)
8	CF Parameter Set	8	PCF (polling system)-related parameters
9	IBSS Parameter Set	4	ATIM window length (K μ sec.) used in the ad-hoc mode

③ Authentication frames

Once the target for communication is detected through the use of beacon, probe request and probe response frames, authentication between the radio stations is performed using an authentication frame.

Authentication frames and de-authentication (“cancellation of authentication”) frames are assigned with the Subtype values “1011” and “1010,” respectively (B4 to B7 of the Frame Control field).

The information elements of authentication frames are as follows:

Table A9.1-4 Information Elements of Authentication Frames

No.	Information element	Length [octets]]	Explanation
1	Authentication Algorithm No.	2	Authentication method 0：Open System 1:Shared Key
2	Sequence Number	2	Authentication sequence number
3	Status Code	2	Authentication status in terms of success or failure and reason in the case of failure
4	Challenge Text	2-34	Challenge text used in shared key authentication

Open system authentication is compulsory in both the ad-hoc and infrastructure modes, but shared key authentication is not compulsory in the ad-hoc mode. However, it is possible to perform shared key authentication in the ad-hoc mode through exchange of frames explained above between terminal stations (STA's).

In the ad-hoc mode, it is in principle possible to establish connection using frames explained above (as explained later). In the infrastructure mode, "association" frames for connecting terminal stations (STA's) with base stations (AP's) and "re-association" frames for re-connecting terminal stations with base stations are also used.

④ Association frames

"Association" is the processing performed when an STA connects to an AP. The STA sends an association request frame to the AP and then waits for an association response frame containing the status code value "success" from the AP. If the STA receives such an association response frame within the specified period of time, the STA deems that it has successfully connected.

Association request frames and association response frames are assigned with the Subtype values "0000" and "0001," respectively.

The information elements of association request frames and association response frames are as shown in the tables below.

No.	Information element	Length [octets]]	Explanation
1	Capability Information	2	Information as to whether polling, encryption, etc. are to be performed
2	Listen Interval	2	Beacon frame reception interval to be used when STA is in power-saving mode
3	Service Set ID	2-34	ESS or IBSS identifier
4	Supported Rates	3-10	List of supported transmission rates

Table A9.1-5 Information Elements of Association Request Frames

Table A9.1-6 Information Elements of Association Response Frames

No.	Information element	Length [octets]]	Explanation
1	Capability Information	2	Information as to whether polling, encryption, etc. are to be performed
2	Status Code	2	Connection status in terms of success or failure and reason in the case of failure
3	Association ID (AID)	2	Terminal identifier (1 to 2007) to be assigned to the recipient STA
4	Supported Rates	3-10	List of supported transmission rates

The STA starts the connection processing as soon as a connection request primitive is issued by the management function of the MAC sub layer.

The STA first sends an association request frame to the AP. If the AP judges after receiving the association request frame that it is possible to comply with the request for connection, the AP sets the status code to “success,” assigns an AID value and returns a response. If the request is one from an STA that has not completed authentication, the AP returns a de-authentication frame.

If the STA receives an association response frame containing the status code value “success” from the AP, it deems that it has successfully connected and sends an acknowledgement response frame (ACK frame) to the AP. Upon receipt of the ACK frame by the AP, a mode change to the normal communication mode takes place.

An ACK frame consists of a 2-octet Frame Control field containing the Subtype value “1101”, a 2-octet Duration field, a 6-octet reception station address field and a 4-octet FCS.

Upon receipt of the ACK frame from the STA, the AP deems that the connection has been established.

⑤ Re-association frames

“Re-association” is the processing to reconnect an STA to an AP. This reconnection processing uses re-association request and response frames. The information elements of re-association request frames and re-association response frames are as shown in the tables below.

Table A9.1-7 Information Elements of Re-association Request Frames

No.	Information element	Length [octets]]	Explanation
1	Capability Information	2	Information as to whether polling, encryption, etc. are to be performed
2	Listen Interval	2	Beacon frame reception interval to be used when STA is in power-saving mode
3	Current AP Address	6	MAC address of the base station from which handoff is to be performed
4	Service Set ID	2-34	ESS or IBSS identifier
5	Supported Rates	3-10	List of supported transmission rates

Table A9.1-8 Information Elements of Re-association Response Frames

No.	Information element	Length [octets]]	Explanation
1	Capability Information	2	Information as to whether polling, encryption, etc. are to be performed
2	Status Code	2	Connection status in terms of success or failure and reason in the case of failure
3	Association ID (AID)	2	Terminal identifier (1 to 2007) to be assigned to the recipient STA
4	Supported Rates	3-10	List of supported transmission rates

2. Startup and operation during ad-hoc mode connections
   a. Network searches

   In the ad-hoc mode, all STA's in the network transmit beacon frames. Each STA performs, using information obtained from received beacon frames, the time synchronization necessary to perform communication.

   The figure below shows how STA's transmit beacon frames in the ad-hoc mode.

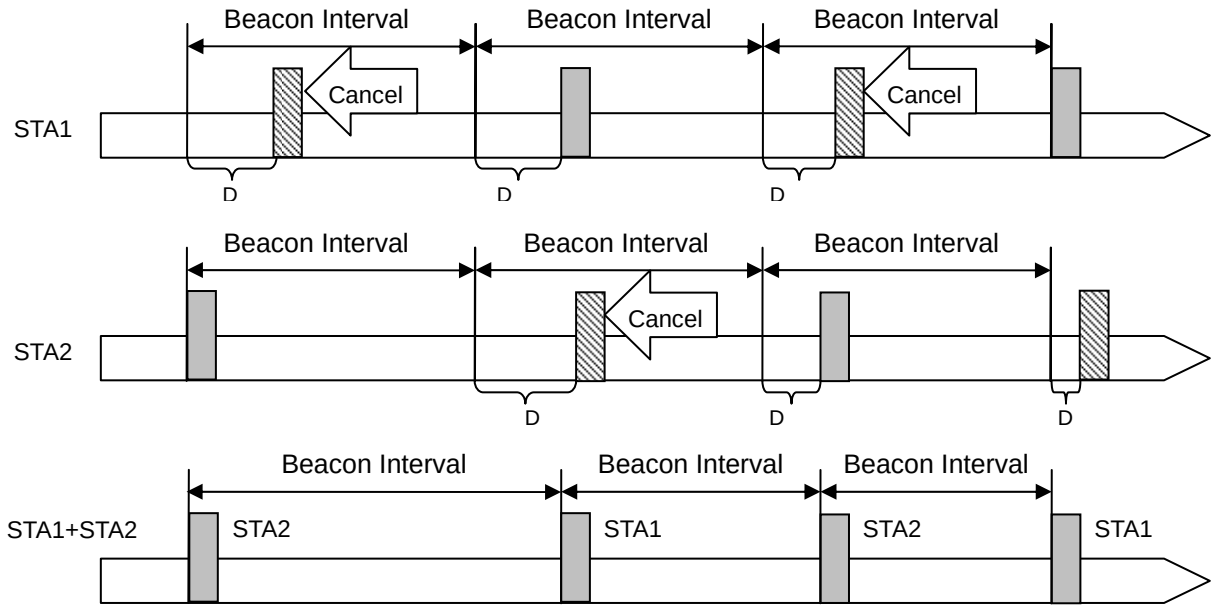


Fig. A9.1-3 Transmission of Beacon Frames in the Ad-hoc Mode

In the ad-hoc mode, all STA's transmit beacon frames in a distributed and autonomous manner. Because beacon frames contain information indicating the beacon frame interval, each STA joining the network determines the timing of the next beacon frame transmission (Target Beacon Transmitting Time (TBTT)) by referencing the beacon frame interval information contained in the received beacon frames, and, based on the TBTT, determines when to transmit its own beacon frame. The figure above conceptually shows the distributed and autonomous transmission of beacon frames by terminals (STA1 and STA2 in this case).

Each STA determines at random the time (D1) by which the beacon frame transmission timing is to be delayed and reduces the length of the D1 time every time a transmission opportunity comes.

If other beacon frame is received within D1 from a transmission opportunity, the transmission of the beacon frame at that opportunity will be canceled.

Compliance by each terminal with this rule results in a beacon frame transmission distribution like the one (STA1+STA2) shown in the figure.

In the ad-hoc mode, a technique called "passive scanning" is used to allow STA's to search for the networks to connect to.

The passive scanning method allows STA's to monitor beacon frames transmitted by other STA's and determine, using information contained in them, the networks to connect to.

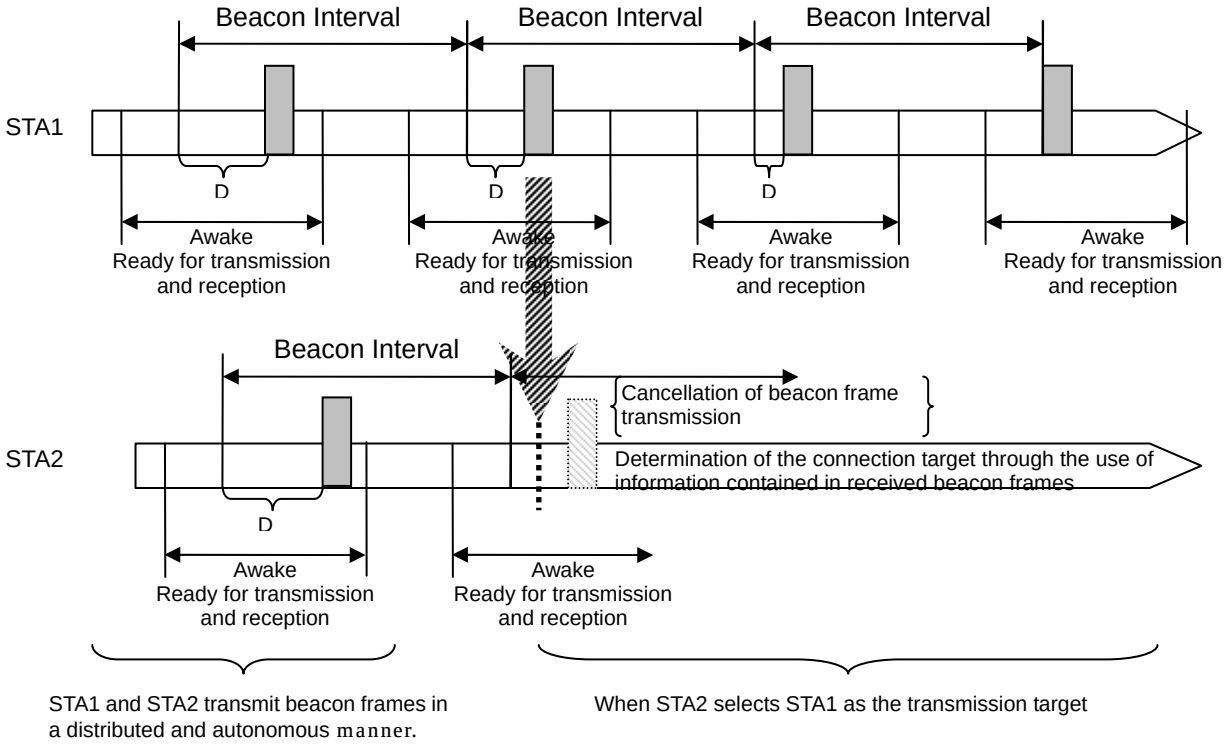


Fig. A9.1-4 Use of Beacon Frames in the Ad-hoc Mode

The example above shows the case where STA2 determines its transmission target (STA1) from an STA1's beacon frame.

b. IP address determination

After going through the above-mentioned process, STA's become eligible for processing in the network layer of the OSI basic reference model. It is recommended that a DHCP client be implemented in IP media in the ECHONET as the means of IP address setting.

After completion of the above-mentioned association, the individual lower-layer communication software broadcasts a DHCP client message that contains such information as the STA MAC address setting and determines the IP address from the DHCP server response. The DHCP protocol is explained in detail in RFC2131 (former RFC1541).

c. Reconnections in the ad-hoc mode

When a radio-based medium is used, connections and disconnections occur frequently regardless of whether or not the medium is an IEEE802.11/11b medium, because of the frequent exits of devices from communication areas and the frequent turning off of the power to save battery.

In the case of the ad-hoc mode, the processing from the beacon frame reception stage described above is repeated, because no association or re-association with an AP

takes place in the ad-hoc mode.

Because the IP address is not changed unless explicit individual lower-layer communication software rewriting occurs, disconnections and re-connections in the normal ECHONET stop state are possible. However, in the case where an interruption by another STA occurs while an STA is in the stop state, it is essential to reconfirm the IP address and hence the ECHONET MAC address because it is necessary to deem, to be on the safe side, that the DHCP assigns an address that is different from the previous IP address.

For this reason, it is recommended that the ECHONET MAC address acquisition sequence in the A-MODE described in "7.7.4 ECHONET MAC address acquisition sequence" be performed. When this sequence is performed, it is necessary to perform the initialization sequence again, receive device identification information and take the necessary steps by means of application software, because the ECHONET MAC address may be changed.

In the case where ECHONET MAC addresses must not be changed for application-related reasons, ECHONET MAC address server-based management becomes necessary.

The figure below shows a flowchart of connection- and reconnection-related events in the ad-hoc mode.

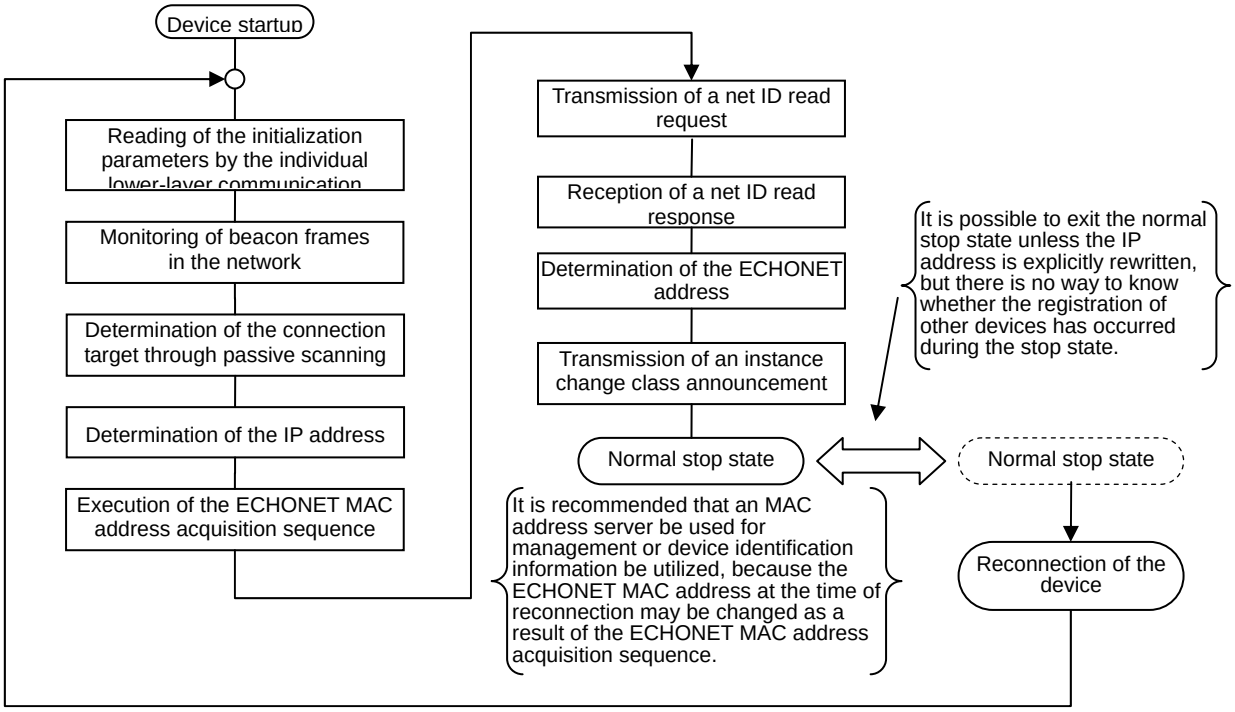


Fig. A9.1-5 Connection and Reconnection of Device in the Ad-hoc Mode

3. Startup and operation during infrastructure mode connections

a. Transmission of beacon frames

The pattern of transmission of beacon frames in the infrastructure mode is simpler than that in the ad-hoc mode, because the transmission of beacon frames in the infrastructure mode is managed by an AP.

The figure below shows the pattern of transmission of beacon frames by an AP. Beacon frames are set to the parameter value “dot1BeaconPeriod” and are in principle transmitted at fixed intervals. However, because the ordinary CSMA/CA procedure is used to transmit beacon frames, the transmission timing will be delayed when there is another frame that must be transmitted.

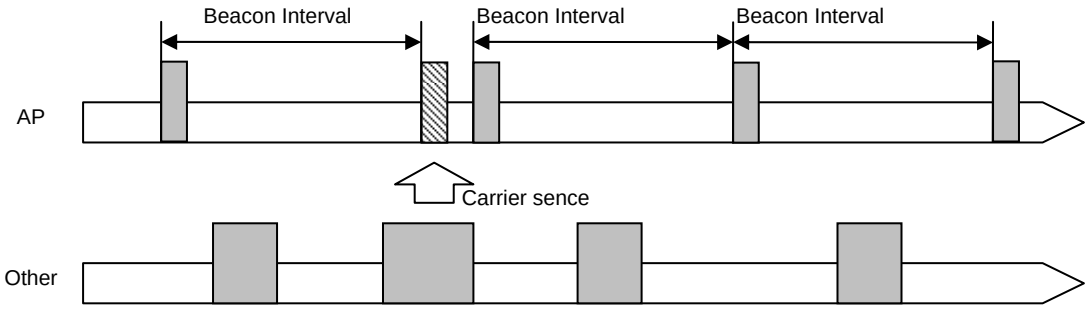


Fig. A9.1-6 Transmission of Beacon Frames in the Infrastructure Mode

The AP provides a notification to the STA by setting the beacon interval setting in a beacon frame or probe response frame and sending the frame. The STA sends a probe request frame to the AP using information contained in the received frame, and performs authentication and association to establish the network connection.

b. Network searches

Network searches in the infrastructure mode use a technique called “active scanning,” which involves exchange of probe request and response frames between the AP and STA.

The STA broadcasts a probe request frame to all available channels. In the case where the network to connect to has already been determined, the STA sets the SSID setting (i.e. the identifier for the network) in a probe request frame, sends the frame and then waits for a response for the specified period of time.

If the STA does not receive a probe response by the end of the minimum channel monitoring period, it will send the request to the next channel. If the STA receives a response, it will wait until the maximum channel monitoring period elapses and then processes the received probe response.

When the AP receives a probe request frame, it will return a probe response frame to the sender of the probe request frame, if the SSID value contained in the probe request frame is a broadcast SSID value or an SSID value that is to be managed by the AP.

c. Authentication

If the STA successfully completes the network search, it will perform the authentication sequence.

In communications based on the IEEE802.11 standard, two types of authentication are used; shared key authentication and open system authentication.

The figure below shows the sequences of shared key authentication and open system authentication.

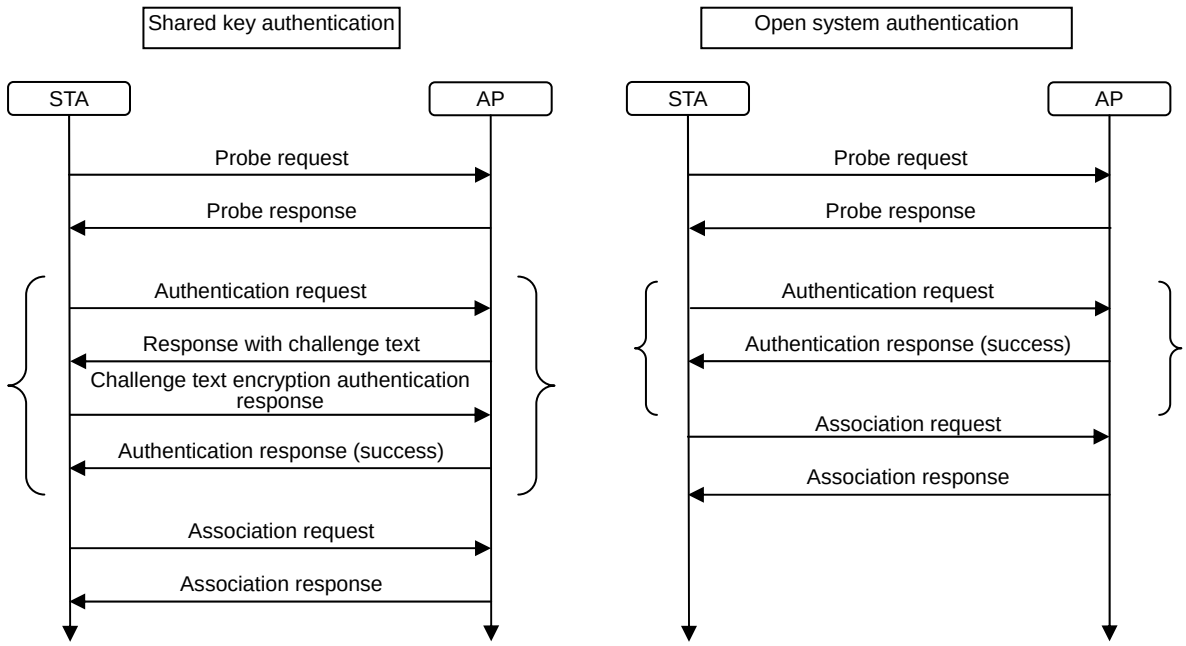


Fig. A9.1-7 Authentication Methods Used in Communications Based on the IEEE802.11 Standard

In shared key authentication, an AP that has received an authentication request from an STA sets the challenge text in an authentication frame and sends the frame to the STA. The STA encrypts the received challenge text and sends it back to the AP. The AP then decrypts the received encrypted challenge text using the shared key. If the decryption is successfully completed, the AP deems that the STA is the right party, sets the status code value to “authentication success” and sends the frame to the STA.

In open system authentication, an authentication response is returned immediately after an authentication request is received (the challenge text is sent, but the sender does not wait for a response) and all STA’s are authenticated.

d. Association

If the STA successfully completes the authentication, it will send an association

request frame to the AP. The figure below shows the association sequence for the STA and AP.

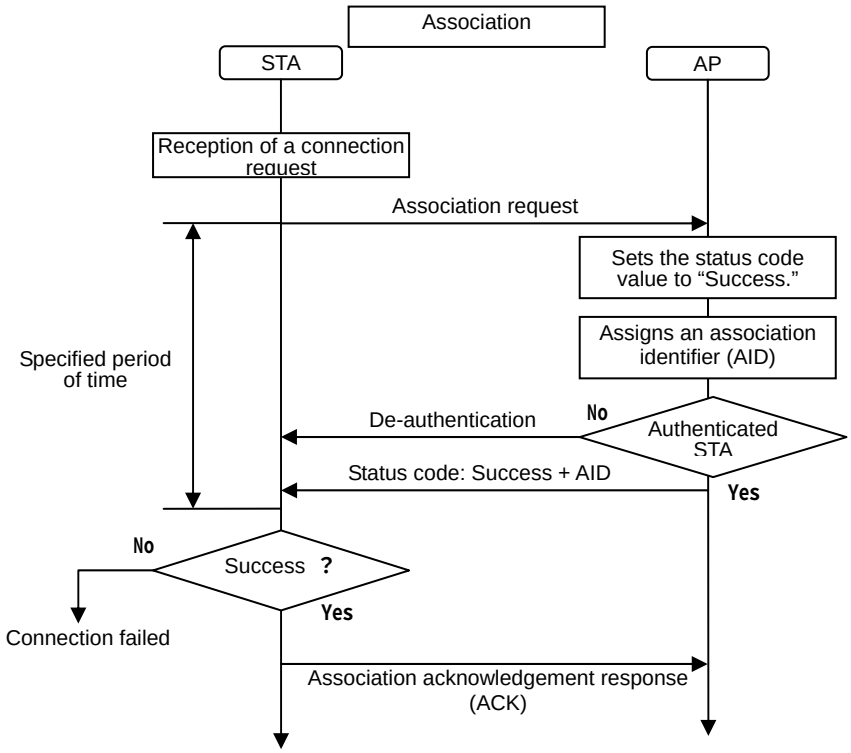


Fig. A9.1-8 Association Sequence

The STA sends an association request frame to the AP with which authentication has been performed. If the AP judges after receiving the association request frame that the sender of the association request frame is the STA with which authentication has been performed, the AP will set the status code value to Success and send it in an association response frame to the STA with an AID (Association ID) value assigned. If the sender of the association request frame is an STA that has not been authenticated, the AP will return a de-authentication frame to the STA.

If an STA that has sent an association request frame to an AP receives an association response frame from the AP which contains the status code value “Success,” the STA will deem that it has successfully connected and send an ACK frame to the AP as an acknowledgement response.

If the received response frame is a frame other than an association response frame or if the STA does not receive an association response frame within the specified period of time, the STA will deem that the connection attempt has failed and perform the appropriate alternative processing (e.g. re-authentication) according to the status code value.

After the connection to the network is established through the above-mentioned

process, the normal ECHONET node startup sequence is started (after the IP address determination process and the ECHONET MAC address acquisition sequence is completed).

e. Re-association

"Re-association" is the reconnection processing performed when an STA handover occurs or when an STA enters or exits a communication area.

The figure below shows the re-association sequence.

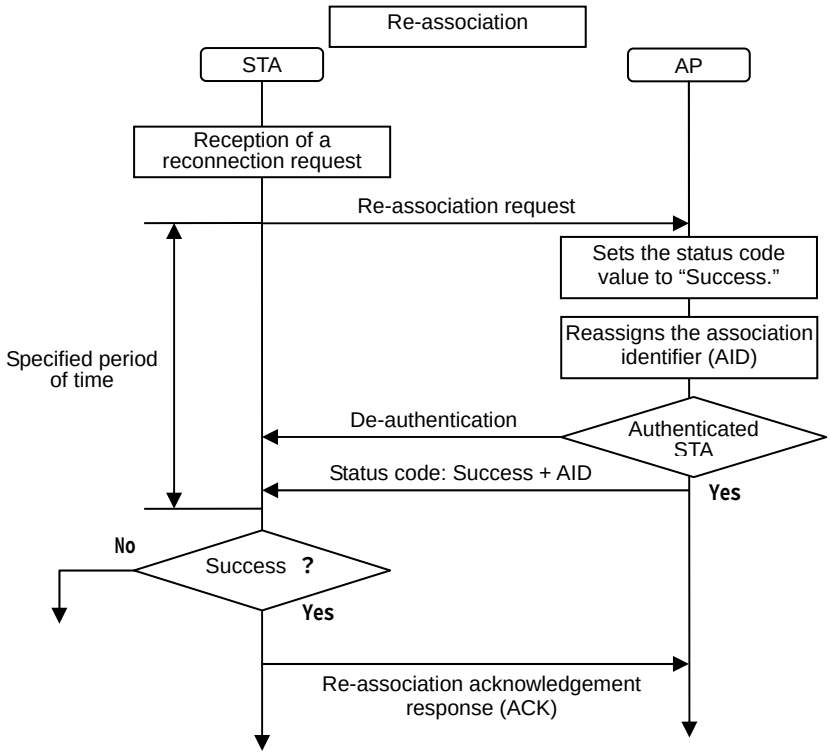


Fig.A9.1-9 Re-association Sequence

The sequence is basically the same as the association sequence. However, in the case of re-association, each re-association request frame contains the MAC address of the base station from which the handoff is made (or the MAC address of the base station at the time of the exit from the communication area), which means that each base station can check whether this MAC address is the same as its own MAC address.

Each AP can confirm the MAC address of the base station from which the handoff is made and assign the previous AID at the time of AID reassignment, but this is not defined in the IEEE802.11 standard.

If reconnection is completed as described above, the previous IP address will be used unless the IP address is explicitly rewritten (as in the case of the ad-hoc mode).

Therefore, the sequence of events is the same as that shown in "Fig. A9.1-5 Connection and Reconnection of Device in the Ad-hoc Mode."

4. ECHONET MAC address acquisition sequence

Acquisition of ECHONET MAC addresses for IEEE802.11/11b shall be as per "7.7.4 ECHONET MAC address acquisition startup sequence."

The UDP packet types defined in Chapter 7 are not intended exclusively for Bluetooth®/IP; they are commonly used for media that uses an intermediary IP network.

As explained in "4) Common accommodation requirements for all IP medium types" of "9.9.1 Communication model," an ECHONET subnet in an IP medium is defined as "a group of different ECHONET/IP media connected by means of Layer 2 bridges."

Therefore, it is possible, for example, that an ECHONET node implemented with Bluetooth® and an ECHONET node implemented with IEEE802.11/11b belong to the same ECHONET subnet.

For this reason, the "ECHONET MAC address acquisition startup sequence" requires a sequence that ensures the uniqueness of ECHONET MAC addresses between different IP media.

However, in the case where the two different media mentioned above are connected by means of an IP router, that is, in the case where they reside in different IP subnets, Item 2 of "4) Common accommodation requirements for all IP medium types" of "9.9.1 Communication model" applies and they belong to different ECHONET subnets, which means that there is no need to consider the possibility of overlapping of the ECHONET MAC addresses.

In either case, the use of Class C is the necessary and sufficient condition with regard to IP addresses because the upper limit for the number of ECHONET nodes applies.

Therefore, this scenario starts from the point where an ECHONET node starts acquiring an ECHONET MAC address using the A-MODE (see Section 7.7.4) as the startup mode after it connected to a Class C IP network and acquired a unique host address using DHCP.

a. Acquisition of an ECHONET MAC address by a sole device

First, a sole ECHONET device starts up from scratch. The IP address of the device at this point is an address that has been acquired freely from an IP network device having a DHCP server function which is not an ECHONET device (the result is the same if manual setting is used).

The ECHONET MAC address acquisition startup sequence in this case is as follows:

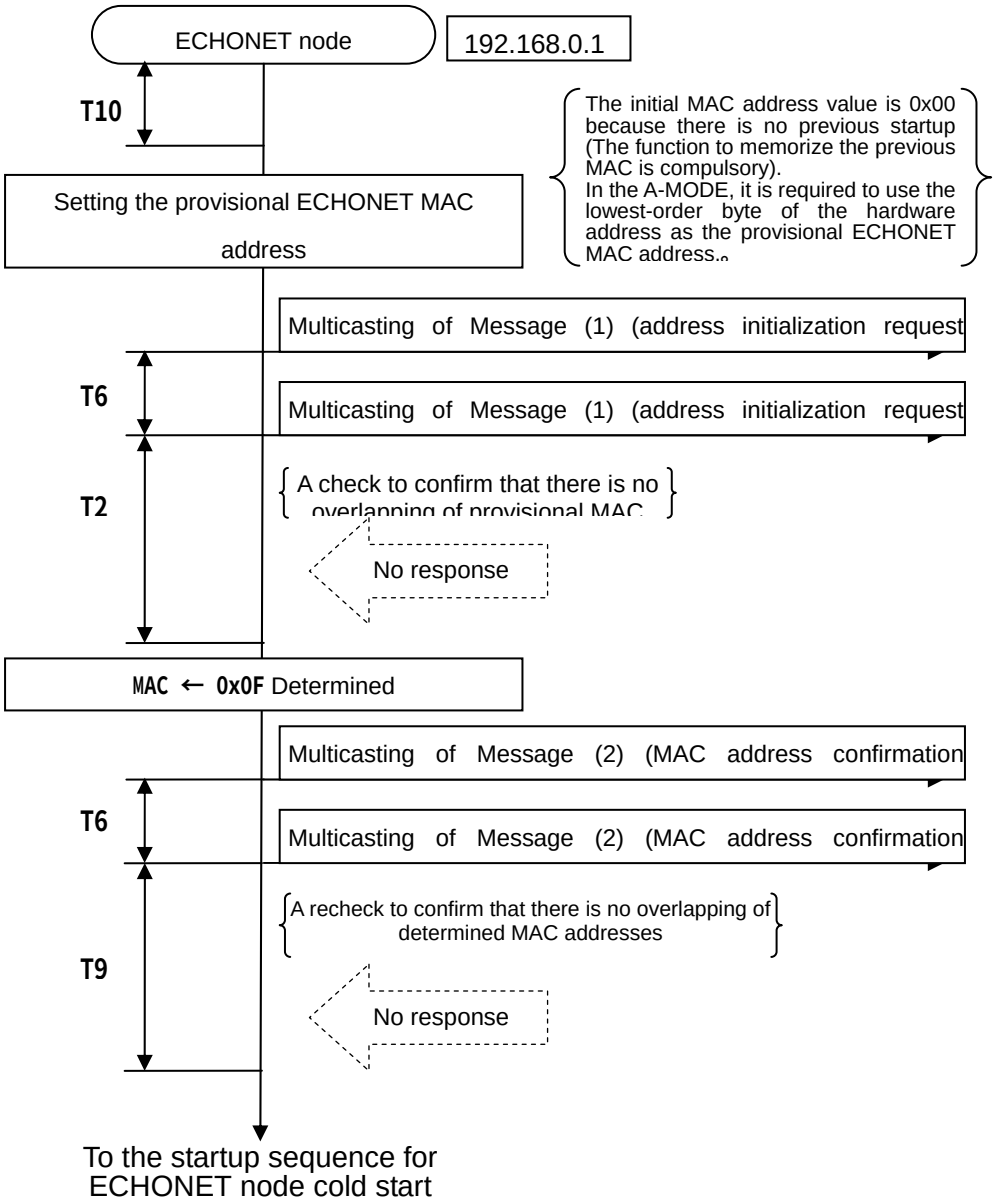


Fig. A9.1-10 ECHONET MAC Address Acquisition Sequence (1)

Table A9.1-9 ECHONET MAC Address Acquisition Sequence (1)

Message (1)	- Version is set to 0x01 - Type is set to 0x07 - Flag is set to 0x00 - A provisional ECHONET MAC address is entered as the RMAC value (0x0F in the case of the example shown in the figure) - RIPAddr is set to 0xC0A80001 (HEX notation of the IP address). - RHType is set to 0x06 (IEEE802.11-compliant hardware) - RHLen is set to 0x06 (MAC address size). - The MAC address of the communication medium (6 bytes) is entered as the RHAddr value. For details, refer to "7.7.2 Packet format."
Message (2)	- Version is set to 0x01 - Type is set to 0x0B - A provisional ECHONET MAC address is entered as the RMAC value (0x0F in the case of the example shown in the figure) - RIPAddr is set to 0xC0A80001 - RHType is set to 0x06 - RHLen is set to 0x06 - The MAC address of the communication medium (6 bytes) is entered as the RHAddr value For details, refer to "7.7.2 Packet format."
T2	- 3.0 sec. (Refer to Section 7.7.6.)
T6	- 100 msec. or less (Refer to Section 7.7.6.)
T9	- 3.0 sec. (Refer to Section 7.7.6.)
T10	- 0 to 100 msec. (random number) (Refer to Section 7.7.6.)

ECHONET nodes that start up for the first time shall wait for the T10 period (a random period between 0 and 100 msec.) after formal determination of their IP addresses and then determine their provisional ECHONET MAC addresses. In the A-MODE, it is required to use the lowest-order byte of the hardware address (MAC) as the initial provisional ECHONET MAC address value.

After determination of the provisional ECHONET MAC address, an ECHONET MAC address initialization request packet is multicast in accordance with the requirements specified in (c) 3. of Section 7.7.4.

Because there is no other ECHONET node in this setting, the specified period of time elapses with no ECHONET MAC address initialization response packet received, at which point the provisional ECHONET MAC address formally becomes the ECHONET MAC address.

After this point, an additional packet for confirmation is transmitted in accordance with (c) 8. of Section 7.7.4, because, depending on the timing, the same ECHONET MAC

address may have been assigned to another node around the same time. Since no response is sent back in this particular setting, the ECHONET MAC address is formally fixed after the passing of the T9 period and the startup sequence for a normal ECHONET node cold start starts.

b. Addition of devices

If another device newly joins under the same conditions that are described in a., an ECHONET MAC address initialization response packet will be returned by the mode that is already in operation. The figure below shows a typical sequence for this case. For the detailed sequences, refer to Section 7.7.4.

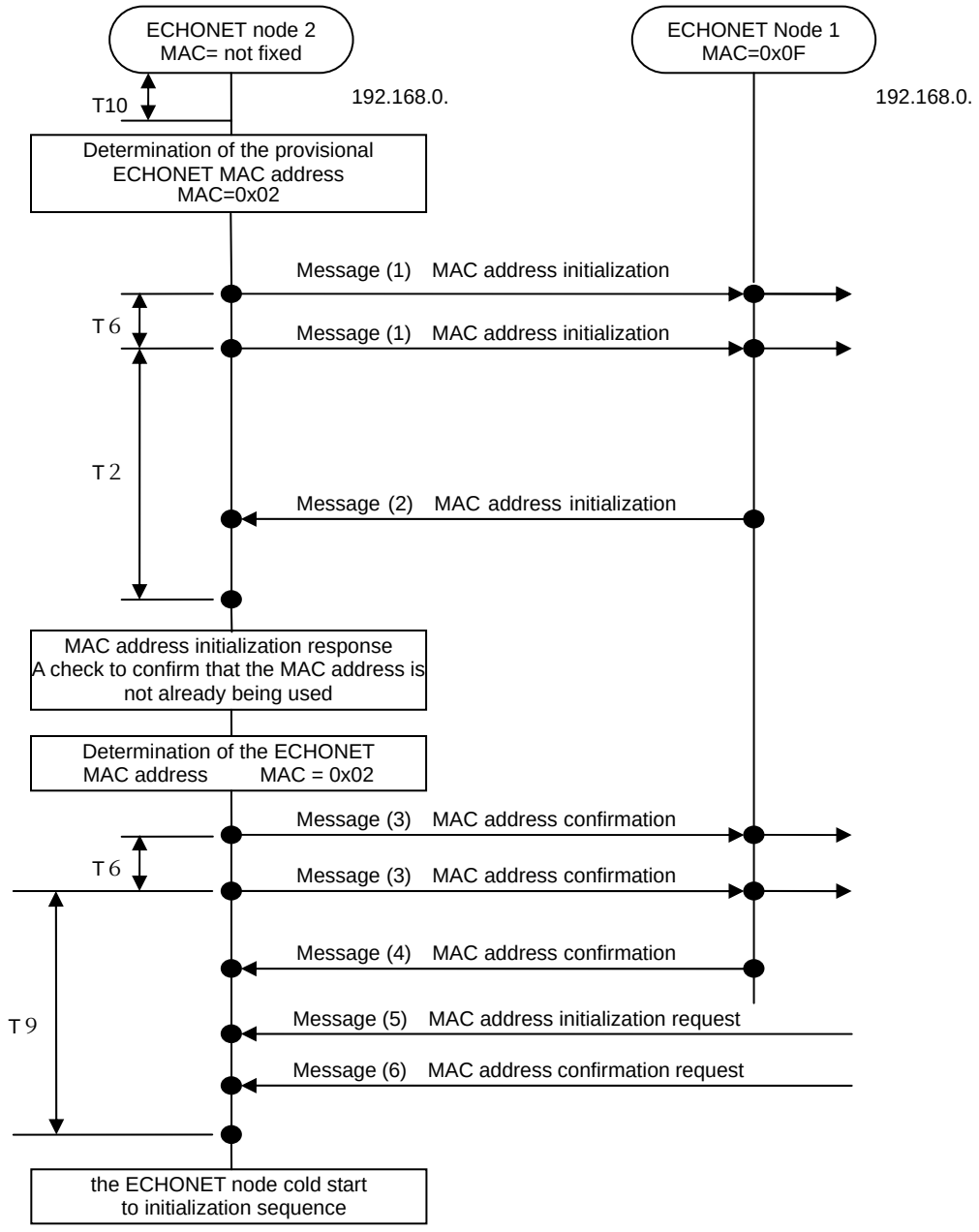


Fig. A9.1-11 ECHONET MAC Address Acquisition Sequence (2)

Table A9.1-10 ECHONET MAC Address Acquisition Sequence

Message (1)	- Version is set to 0x01 - Type is set to 0x07 - Flag is set to 0x00 - A provisional ECHONET MAC address is entered as the RMAC value (0x02 in the case of the example shown in the figure) - RIPAddr is set to 0xC0A80002 (HEX notation of the IP address) - RHType is set to 0x06 (IEEE802.11-compliant hardware) - RHLen is set to 0x06 (MAC address size) - The MAC address of the communication medium (6 bytes) is entered as the RHAddr value For details, refer to “7.7.2 Packet format.”
Message (2)	- Version is set to 0x01 - Type is set to 0x08 - Flag is set to 0x00 (0x01 in the case of a master router) - The ECHONET MAC address of the responder node is entered as the TMAC value (0x0F in the case of the example shown in the figure) - TIPAddr is set to 0xC0A80001 - THType is set to 0x06 - THLen is set to 0x06 - The MAC address of the responder side communication medium (6 bytes) is entered as the THAddr value For details, refer to “7.7.2 Packet format.”
Message (3)	- Version is set to 0x01. - Type is set to 0x0B - A provisional ECHONET MAC address is entered as the RMAC value (0x02 in the case of the example shown in the figure) - RIPAddr is set to 0xC0A80002 - RHType is set to 0x06 - RHLen is set to 0x06 - The MAC address of the communication medium (6 bytes) is entered as the RHAddr value For details, refer to “7.7.2 Packet format.”
Message (4)	- Version is set to 0x01 - Type is set to 0x0C - Flag is set to 0x00 (0x01 in the case of a master router) - The ECHONET MAC address of the responder node is entered as the TMAC value (0x0F in the case of the example shown in the figure) - TIPAddr is set to 0xC0A80001 - THType is set to 0x06 - THLen is set to 0x06 - The MAC address of the responder side communication medium (6 bytes) is entered as the THAddr value

	For details, refer to "7.7.2 Packet format."
Message (5)	Packet from other node which is in the same format as Message (1)
Message (6)	Packet from other node which is in the same format as Message (3)
T2	- 3.0 sec. (Refer to Section 7.7.6.)
T6	- 100 msec. or less (Refer to Section 7.7.6.)
T9	- 3.0 sec. (Refer to Section 7.7.6.)
T10	- 0 to 100 msec. (random number) (Refer to Section 7.7.6.)

If an ECHONET node is added when other nodes already exist in the domain, an ECHONET MAC address initialization response will be returned from each of those nodes after the multicasting of the ECHONET MAC address initialization request.

Any node that intends to join a domain must monitor all responses returned within the T2 period and check for an ECHONET MAC address overlap.

It is deemed that an overlap exists if any of the following conditions, which are specified in (ii) 7 of Section 7.7.4, is met:

a. The provisional ECHONET MAC address coincides with one of the ECHONET MAC address that are already being used in the subnet.
b. The provisional ECHONET MAC address contained in a received ECHONET MAC address initialization request packet coincides with the provisional home ECHONET MAC address.
c. The provisional ECHONET MAC address contained in a received ECHONET MAC address confirmation request packet coincides with the provisional home ECHONET MAC address.

Since none of the 3 conditions listed above is met in this setting, the provisional ECHONET MAC address formally becomes the home ECHONET MAC address after the passing of the T2 period.

After this point, an additional packet for confirmation is transmitted in accordance with (c) 8. of Section 7.7.4 and a check is made to see if there is any response or request packet that meets any of the 3 conditions listed above, because, depending on the timing, the same ECHONET MAC address may have been assigned to another node around the same time.

In this setting, an ECHONET MAC address confirmation response (Message (4)) is returned by ECHONET Node 1, but none of the 3 conditions listed above is met. As a result, the provisional ECHONET MAC address formally becomes the ECHONET MAC address of ECHONET Node 2 and the startup sequence for a normal ECHONET node cold start starts.

If a packet is received during the above-mentioned T2 or T9 period which meets any of the 3 conditions listed above, a different provisional ECHONET MAC address will be issued, an ECHONET MAC address confirmation request packet will be multicast and a

check for an ECHONET MAC address overlap will be made again. For details about the method of changing provisional ECHONET MAC address, refer to (ii) 8 of Section 7.7.4. With regard to ECHONET MAC address acquisition, the settings described in a. and b. above represent the simplest sequences. For more complex sequences such as sequences for the case where a larger number of devices are involved or for the case where 2 nodes start up around the same time, refer to Chapter 7.

Supplement 9.2 Basic Philosophy for the Use of IEEE802.11/11b Transmission Media Standard-compliant Devices

Because IEEE802.11/11b-compliant devices fall under the “Second-generation Low-power Data Communication Systems’ Radio Stations” category defined by ARIB STD-T66, they shall be used in accordance with the “Guidelines for the Use of Second-generation Low-power Data Communication Systems’ Radio Stations” (Reference Document 2 for ARIB STD-T66).

The Guidelines were defined to prevent malfunctions due to mutual interferences, etc., because devices that use the ISM Frequency Band, including IEEE802.11/11b- and Bluetooth®-compliant devices, microwave ovens and general medical devices, can in principle be used without a license.

The ECHONET Specification is a standard intended to promote the interconnectivity between control functions of various manufacturers’ and vendors’ equipment, devices and home appliances interconnected via networks as well as between control functions of different models of equipment, devices and home appliances interconnected via networks, and is not intended as a standard that covers faults, etc. due to mutual interferences with ISM Band devices.
Therefore, when using IEEE802.11/11b as a transmission medium, it is recommended that the above-mentioned standard be obtained from the Association of Radio Industries and Businesses and the “Guidelines for the Use of Second-generation Low-power Data Communication Systems’ Radio Stations” be strictly followed.

The ECHONET Specification does not cover deteriorations of communication functions caused by microwave ovens, IH equipment, general high-frequency treatment devices, etc. or by mutual interferences with Bluetooth transmission media. These are problems that concern the performance of individual produces, which should be avoided at the application level by the manufacturer.

Chapter 10 Specifications for the Power Line Communication Protocol C System

10.1 Physical Layer Specifications

The physical layer specifications are as per the IT800PHY system of YITRAN Communications Ltd., Israel. Contact the ECHONET Consortium Secretariat for the method to obtain the specifications.

10.2 Logical Specifications

This section provides the requirements relating to the Layer 2 packet format for the Power Line Communication Protocol C System.

10.2.1 Layer 2 packet format

The Layer 2 protocol packet format is as follows:

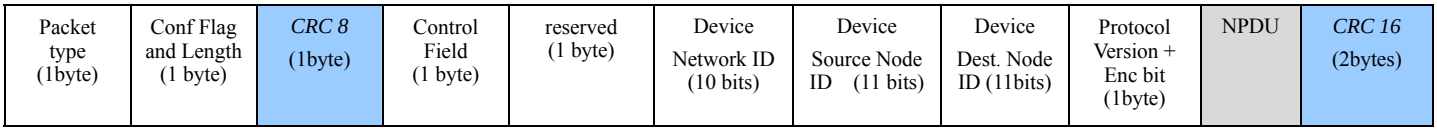


Fig. 10.1 Data Link Layer Packet Format

(1) NPDU (Network Protocol Data Unit)

NPDU represents the data field received from Layer 3.

The other fields (the fields shown in white in the figure above) are added and decoded by Layer 2. Layer 2 is also responsible for retransmissions, responses, broadcasts and retries.

About implementation: Layer 1 applies CRC8 (confirmation of the Layer 1 header of the packet) to the bit strings of Packet type and Conf Flag and Length and uses CRC16 (confirmation of the Layer 2 header and NPDU) at the end of the packet.

(2) “Packet type” field (1 byte)

The “Packet type” value is set in accordance with the transmission protocol.

0xFE (Standard mode)

0xFC (Robust mode)

0xFD (Acknowledgement Packet)

0xFB (Extremely Robust Mode)

(3) “Conf Flag and Length” field (1 byte)

The MSB of this byte shall always be 0, and the remaining 7 bits shall be used as the

data length field.

(4) CRC8 (1 byte)
This is the CRC for the packet header, which is used for the Packet type and Conf Flag and Length fields.

(5) Control Field (1 byte)
Control Field specifies the service type and packet priority. In addition, every time a new NPDU is transmitted, 2 bits of this byte are used as the sequence number (0 to 3)(Refer to Fig. 10.2)

(6) reserved (1 byte)

(7) Protocol Version + Enc (encryption) bit (6 bits + 1 bit)
Protocol Version indicates the Layer 2 transfer protocol. Layer 2 ignores other types of protocols.
The encryption bit indicates whether or not an encryption function is used.
The remaining 1 bit is not used.
The MSB corresponds to the encryption bit, and the LSB corresponds to the protocol version.

(8) Address fields (Device Network ID, Device Source Node ID and Device Dest. Node ID fields)
Each of these fields contains address information. Device Network ID is used as the sub network identifier. Device Source Node ID indicates the unique address of each node. Device Dest. Node ID indicates the destination address or broadcast address.

(9) CRC16
This is the CRC (2 bytes) for the packet data (header + NPDU).

The figure below shows an example Control Field structure

Sequence number	Priority	Service
		Reserved except for the following: 1001 – Repetitive Unacknowledged Request 1010 – Acknowledge Request 1011 – Fragmented Packet 1100 – Last Fragment

	00 – Low 01 – Ordinary 10 – High
0 to 3	

Fig. 10.2 Example Control Field Structure

Repetitive Unacknowledged Request : Request not to send the same response repeatedly
Acknowledge Request : Acknowledgement request
Fragmented Packet : Split packet
Last Fragment : Last split packet

10.2.2 Layer 2 packet delivery services

Layer 2 has a response confirmation function which allows Layer 2 to confirm whether packets have been received properly by the destination nodes on the connected media. Each destination node transmits a response packet using the response window after receiving a packet properly. Layer 2 can resend the message when no response packet is transmitted or when no response packet is received.

(1) Repetitive Unacknowledged Service

This is used when the destination address is a broadcast address or when no response is requested to a single node.

The maximum number of times Layer 2 transmits a packet is determined by the value stored in the Layer 2 software.

Layer 2 must control channel access conflicts for each transfer.

“Repetitive Unacknowledged Service” is specified by the Service subfield of the Control Field (“Repetitive Unacknowledged Request” (1001)).

A “Sequence number” value (0 to 3) is assigned to each of the messages sent to the same node. This is used when discarding redundant packets. During packet retransmissions, the same sequence number is used.

In the case of a packet whose priority must be specified, the “Priority” setting is set.

The source node ID is also used to avoid packet redundancy at receiving nodes.

This service ensures high reliability in relation to broadcast messages.

(2) Repetitive Unacknowledged sequence / address association

Because there may be cases where redundant packets are sent to a destination address, a function must be implemented in Layer 2 that allows packet redundancy to be avoided.

To detect redundant packets, source node ID and sequence number are used by the Layer 2 of the receiving nodes. Receiving node association is used to discard all redundant packets transmitted by the transmitting nodes during retransmissions. All

retransmission packets have the same source node ID and sequence number.

To discard the redundant packets, the Layer 2 of the each receiving node extracts the association and performs a check.

(3) Acknowledged Service

This is used in one-to-one communication to judge whether the transmitting node succeeded in transmitting the packet.

If CRC16 of the received packet does not indicate an error and the destination ID matches the device address of the destination, the destination node will send an acknowledgement response to the transmitting node.

An 11-Unit Symbol Time (UST) (“acknowledgement response window”) is used as the channel access protocol, to allow nodes to send an acknowledgement response without a channel access conflict. This window is only used for packet reception confirmation and is not used for other traffic.

The destination must transmit an acknowledgement response packet during the first UST of the acknowledgement response window (UST#4 in Fig. 10.3). Other nodes are not allowed to transmit a packet using the acknowledgement response window except in the case where they return a response after receiving a packet.

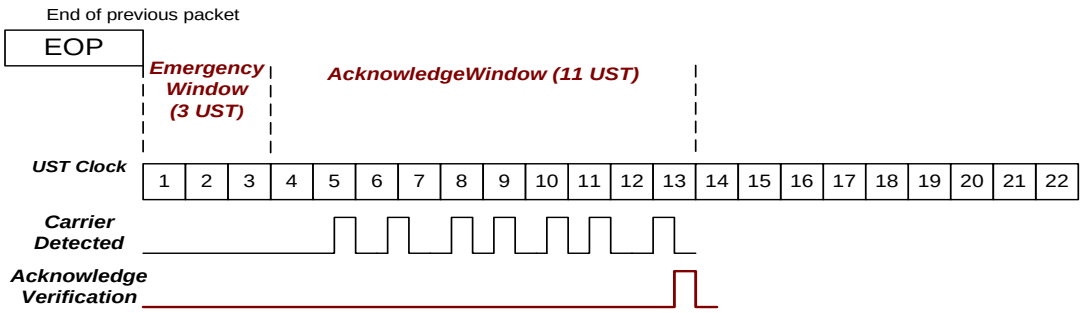


Fig. 10.3 Example Acknowledgement Response Timing

If no acknowledgement response is received, Layer 2 must have the transmission repeated until the value of the parameter for the number of retransmissions is reached.

“Acknowledged Service” is specified by the Service subfield of the Control Field (“Acknowledgement Request” (1010)). A “Sequence number” value (0 to 3) is assigned to each of the messages sent to the same node. During packet retransmissions, the same sequence number is used. In the case of a packet whose priority must be specified, the “Priority” setting is set.

10.2.3 Acknowledgement response packets

Layer 2 of the destination node will return an acknowledgement response when it receives a packet properly. This packet is different in format from ordinary message packets. In the case of response packets, no media conflict occurs. Layer 2 sends a single

byte that has a special response packet type (0xFD) to the PLC. The length is set to 0.

10.3 Layer 2 - 3 Interface Command Set

Power Line Communication Protocol C System devices achieve Layer 1 and Layer 2. Layer 3 interfaces with Layer 2 using a command set via UART.

The figure below shows a block diagram of a Power Line Communication Protocol C System PLCC (Power Line Carrier Communication) node.

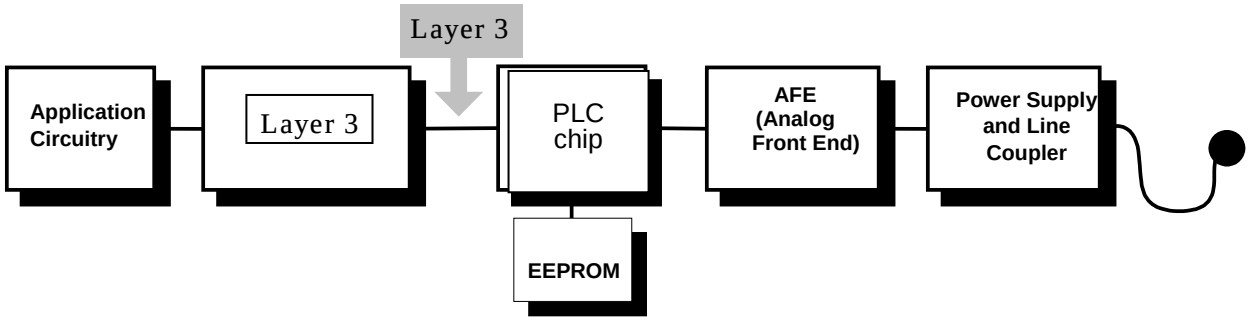


Fig. 10.4 Example Power Line Communication Protocol C System PLC Node Block Diagram

This section provides the requirements about the command set used for communication between Layer 3 and Layer 2.

Definitions of Terms

Acknowledge Service	Data is retransmitted until: (1) The destination node confirms the receipt of the data; or (2) The number of retransmissions reaches the parameter value that has been set.
Broadcast transmission	A transfer is made to all other nodes located in the same logical network. The DID field of the PTX command is set to 0x00.
DCSK	Differential Code Shift Keying
DID	Device Dest. Node ID Destination ID: DID is set to the PTX command to determine the target node.
DLL	Data Link Layer
Large packet	A large packet is a payload whose size is 120 bytes or more. Large packets are split into 2 or more split packets before transmission.
LSb	Least significant bit
LSB	Least significant byte
MSB	Most significant byte
NID	Device Network ID Network ID: Used as part of the PTX command. NID is assigned by the SDA command.
PLCC	Power Line Carrier Communication
Priority	Priority is set as part of PTX. When the Power Line Communication Protocol C System (automatically) detects a high-priority transfer on a channel, it will deem that the channel is occupied.
Repetitive Unacknowledged Service	Transmission is repeated until the parameter value that has been set is reached. No response is returned from the destination.
Small packet	A small packet is a payload whose size is less than 121 bytes.
SID	Device Source Node ID Source ID: Values up to 1023 (decimal numbers) are used. SID is assigned by the SDA command. The PTX command is used to identify the transmitting node.
Unicast Transmission	Data is transferred from a single source to a single destination.

10.3.1 Address management and protocol version

Layer 3 must set the addresses settings of nodes that join the network. In addition, the Protocol Version value is set to avoid collisions between products that have the same network ID and source ID.

Vendors must use the protocol version value specified below.

Implementation	Protocol Version value
ECHONET	0x01

10.3.2 Command and response formats

Each byte is transmitted in the LSb order.

(1) Command format

The format of commands that Layer 3 sends to Layer 2 is as follows:

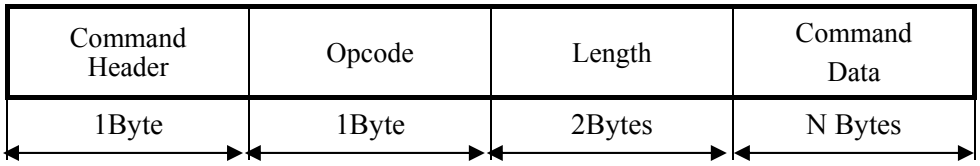


Fig. 10.5 Command Format

The specifications for the fields are as follows:

Table 10.1 Command Fields

Field	Number of bytes	Explanation	Setting
Command Header	1	Fixed in all commands	0xC9
Opcode	1	Command operation code	
Length	2	Number of bytes (hexadecimal notation) that follow this field	N The first byte is the LSB for length, and the second byte is the MSB
Command Data	**N**	Command data	

(2) Response format

The format of responses that Layer 2 returns in response to commands from Layer 3 is as follows:

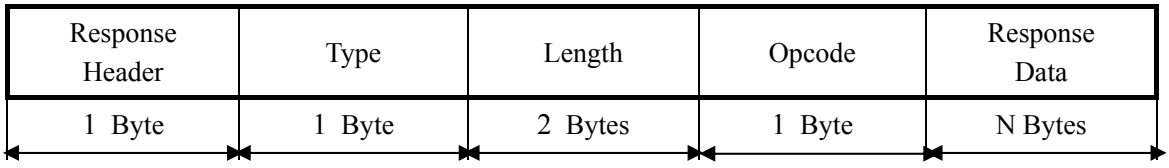


Fig. 10.6 Layer 2 Response Format

The specifications for the fields are as follows:

Table 10.2 Response Fields

Field	Number of bytes	Explanation	Setting
Response Header	1	Fixed in all responses	0xC9
Type	1	There are 2 types of responses. Refer to the explanations about the response types	0x04 or 0x0D
Length	2	Number of bytes that follow this field	K+1 The first byte is the LSB for length, and the second byte is the MSB
Opcode	1	Opcode of the command associated with the response	
Response Data	K	Response data as per the command response option settings	

Layer 3 must wait for a response from Layer 2 before transmitting another command.

10.3.3 Command set

The table below provides an overview of the commands.

Table 10.3 Overview of the Command Set

Command	Opcode	Name	Explanation
PTX	0x09	Packet Transmission Packet Transmission	Layer 3 packet transmission. The packet is transmitted immediately after the command is received.
SDA	0x0D	Set Device Address Set Device Address	Network and node address settings

(1) PTX (packet transmission)

This command is used when Layer 3 transmits data to a Power Line Communication Protocol system.

The options available at the transmission stage are as follows:

* Service type

(“Acknowledged Service” or “Repetitive Unacknowledged Service”)

* Large packet or small packet

Regardless of the content of the Service field of the header of the PTX command data, unicast transmissions of large packets always use “Acknowledged Service.” When a large packet is transmitted as a unicast packet, a response should be returned for each of the fragments. Large packets with the same NID and a different DID are not transmitted to Layer 3.

* Unicast / broadcast

Regardless of the service selection, transmissions of broadcast messages always use “Repetitive Unacknowledged Service.”

* Priority

(A) PTX command

The figure below shows the detailed structure of this command.

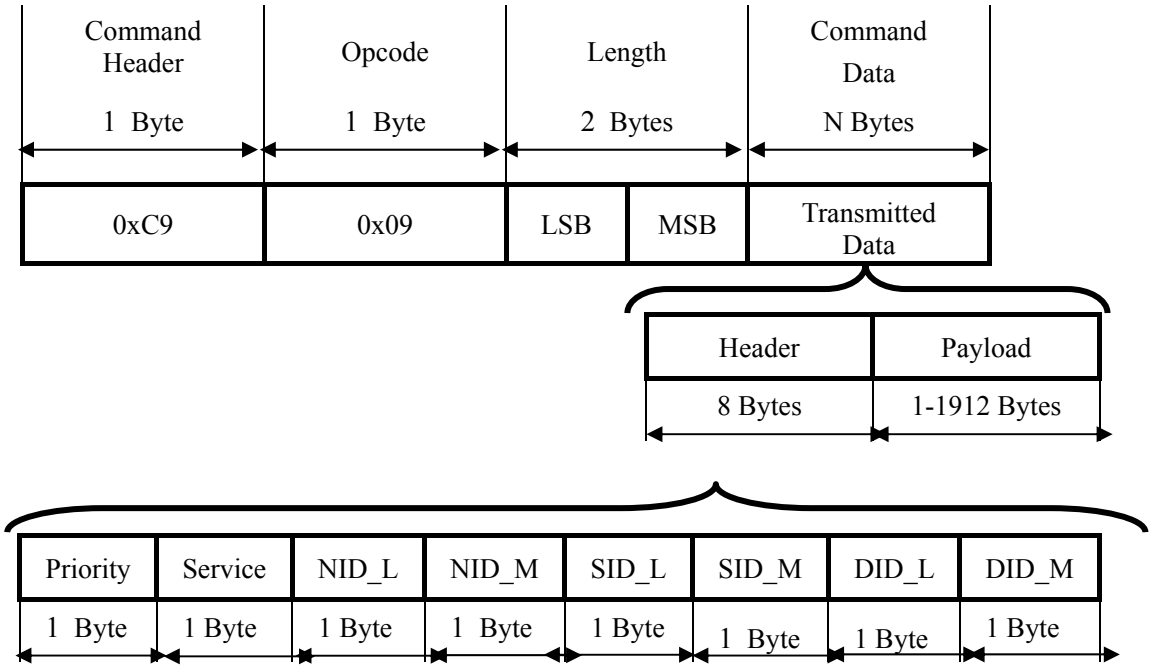


Fig. 10.7 PTX Command

The specifications for the fields are as follows:

Table 10.4 PTX Command

Command	Opcode	Name	Explanation
Command Header	1	Fixed	0xC9
Opcode	1	Fixed	0x09
Length	2	Number of bytes (hexadecimal notation) that follow this field	1st Byte: LSB 2nd Byte: MSB The value range is between 9 and 1920 inclusive (hexadecimal notation).
Command Data	9-1920	The data is transmitted onto the power line. The data contains "Header" (8 bytes) and the payload (1 to 1912 bytes). Large packets are split into split packets before transmission.	
		1st Byte (Host Header) – Priority: Transmission priority setting	0x00: Low 0x01: Ordinary 0x02: High
		2nd Byte(Host Header) – Service	0x01: Acknowledged Service 0x02: Repetitive Unacknowledged Service
		3rd Byte to 8th Byte – The NID, SID and DID values assigned by the SDA command	3rd Byte: NID_L – NID LSB
			4th Byte: NID_M – NID MSB (All "0")
			5th Byte: SID_L – SID LSB
			6th Byte: SID_M – SID MSB (All "0")
			7th Byte: DID_L – DID LSB (0x00 is for broadcast)
			8th Byte: DID_M – DID MSB (All "0")

(B) PTX response

The format of and specifications for responses to transmissions of the PTX command are as shown in the figure and table below. In the case where the received packet is a large packet made up of split packets, additional responses are issued.

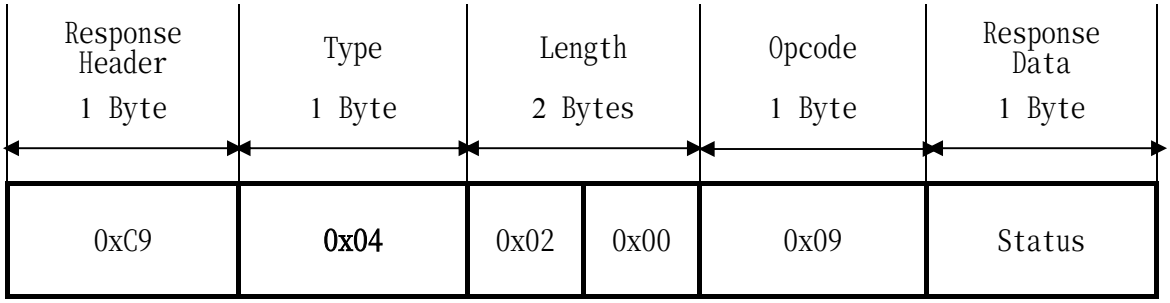


Fig. 10.8 PTX Response

Table 10.5 PTX Response

Command	Opcode	Name	Explanation
Response Header	1	Fixed	0xC9
Type	1	Fixed	0x04
Length	2	Fixed	1st Byte: 0x02 2nd Byte: 0x00
Opcode	1	Opcode of the PTX command	0x09
Response Data	1	Command execution "Status"	Large packet: 1. 0x01 – Packet Received (The additional responses follow.) 2. 0x02 – Packet Rejected Small packet: 1. 0x08 – Packet Blocked – In the case where a packet to be transmitted was not transmitted successfully and the retransmission attempts made up to the number of times permitted by the retransmission parameter failed, the packet must be discarded and the PTX command reissued. 2. 0x80 – ACK failed – —A packet was transmitted using "Acknowledged Service" and no ACK was received after the retransmission attempts made up to the number of times permitted by the retransmission parameter. 3. 0xC0 – Success – A packet was transmitted using "Acknowledged Service" and an ACK was received, or a packet was transmitted using "Repetitive Unacknowledged Service."

(C) Additional responses for large packets

In the case of split transfer (a large packet), additional responses are transmitted as described below after the successful transmission of the response described above.

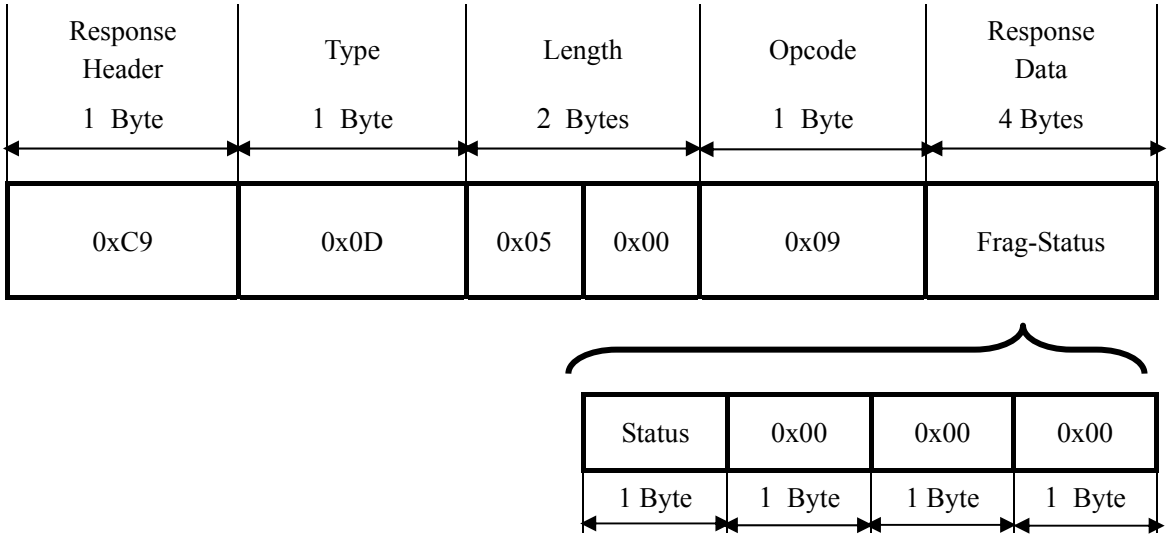


Fig. 10.9 Additional PTX Responses

Table 10.6 Additional PTX Responses

Command	Opcode	Name	Explanation
Response Header	1	Fixed	0xC9
Type	1	Fixed	0x0D
Length	2	Fixed	1st Byte: 0x05 2nd Byte: 0x00
Opcode	1	Opcode of the PTX command	0x09
Response Data	4	"Frag-Status" contains the command execution result followed by three "0x00"s.	"Status": 1. 0x00 – Command failed 2. 0x01 – Command execution successfully completed

(2) SDA (device address setting)

The network and source address settings of nodes are set from the Layer 3 side. Layer 3 must assign and manage addresses.

(A) SDA command

The format of and specifications for this command are as follows:

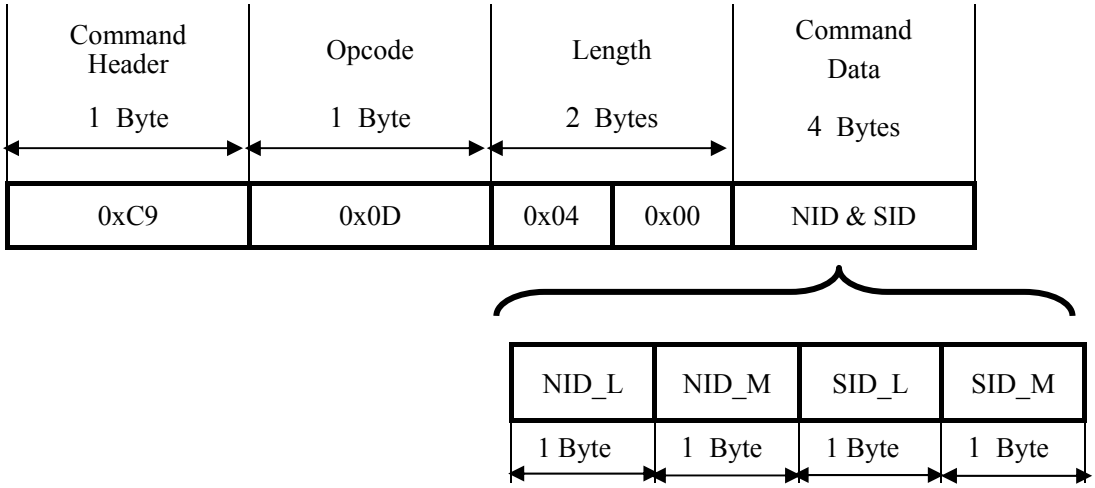


Fig. 10.10 SDA Command

Table 10.7 SDA Command

Command	Opcode	Name	Explanation
Command Header	1	Fixed	0xC9
Opcode	1	Fixed	0x0D
Length	2	Fixed	1st Byte: 0x04 2nd Byte: 0x00
Command Data	4	NID (Network ID) and SID (Source ID) are assigned.	1st Byte: NID_L – LSB
			2nd Byte: NID_M – NID MSB (All "0")
			3rd Byte: SID_L – SID LSB
			4th Byte: SID_M– SID MSB (All "0")

Notes:

1. "0x00" shall not be used for NID or SID. It may be used for SID only in the case of a broadcast transmission.

(B) SDA response

The format of and specifications for responses to transmissions of the SDA command are as shown in the figure and table below.

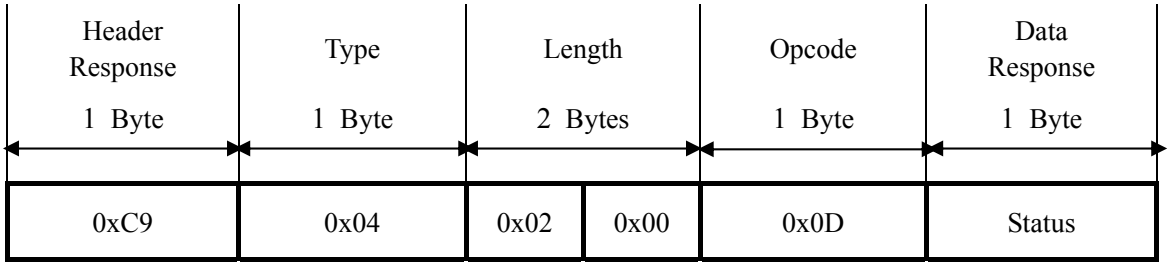


Fig. 10.11 SDA Response

Table 10.8 SDA Response

Command	Opcode	Name	Explanation
Response Header	1	Fixed	0xC9
Type	1	Fixed	0x04
Length	2	Fixed	1st Byte: 0x02 2nd Byte: 0x00
Opcode	1	Opcode of the SDA command	0x0D
Response Data	1	Command execution "Status"	"Status": 0x00 – Command failed 0x01 – Command execution successfully completed

10.3.4 Packet reception

(1) Format of received packets

When a packet is received from a power line, Layer 2 will send the received data to Layer 3 in the format shown in the figure below via a UART interface.

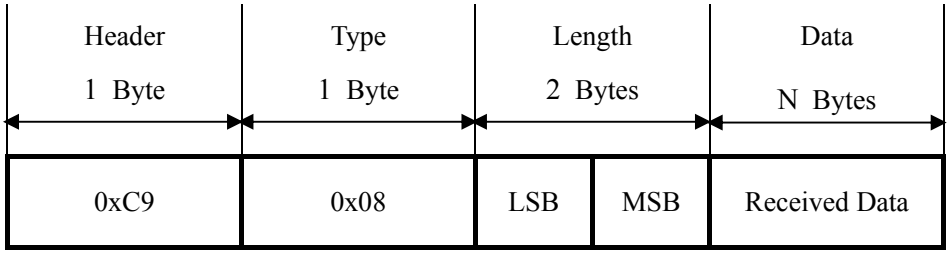


Fig. 10.12 Received Packet

The format of packets sent from Layer 2 to Layer 3 is as follows:

Table 10.9 Received Packet

Command	Opcode	Name	Explanation
Header	1	Fixed	0xC9
Type	1	Fixed	0x08
Length	2	Number of bytes that follow this field	1st Byte: LSB 2nd Byte: MSB The length value range is between 16 and 1927 inclusive.
Received Data	N	Data to be sent from Layer 2 to Layer 3	Refer to (2).

(2) "Received Data" field

This field of the received packet is sent from Layer 2 to Layer 3.

1st Byte	Reserved	ASCII format
2nd Byte	Reserved	
3rd Byte	Reserved	
4th Byte	Device Network ID Highest-order digit	
5th Byte	Device Network ID Second-highest-order digit	
6th Byte	Device Network ID Second-lowest-order digit	
7th Byte	Device Network ID Lowest-order digit	
8th Byte	Device Source Node ID Highest-order digit	
9th Byte	Device Source Node ID Second-highest-order digit	
10th Byte	Device Source Node ID Second-lowest-order digit	
11th Byte	Device Source Node ID Lowest-order digit	
12th Byte	Device Dest. Node ID Highest-order digit	
13th Byte	Device Dest. Node ID Second-highest-order digit	
14th Byte	Device Dest. Node ID Second-lowest-order digit	
15th Byte	Device Dest. Node ID Lowest-order digit	
16th Byte	Data	Binary format
...	...	
1927th Byte (max)	Data	

Fig. 10.13 "Received Data" Field

Layer 2 sends the part other than the data part to Layer 3 in the ASCII format. The data part is sent in the binary format. Therefore, each of the addresses is represented by 4

digits (with one byte used for each digit). The data part starts from the sixteenth byte of the packet to be sent to Layer 3. The range for the length of the data part is between 1 and 1912 bytes inclusive.

The figure below shows the structure of a packet whose data part contains “ABCDE” (command data). This packet will be received after being sent from Node 1 to Node 2 in Network 3.

Received data: 000000300010002ABCDE

Byte:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Content:	0	0	0	0	0	0	3	0	0	0	1	0	0	0	2	A	B	C	D	E

Fig. 10.14 Structure of Received Data (example)

10.4 P&P (Plug and Play) Protocol

10.4.1 Elements for achieving the P&P protocol

P&P is an automatic address assignment function. The system consists of a master node that is responsible for assigning and managing addresses and slave nodes to be assigned with addresses. The master node functions should in principle be achieved by incorporating them in home servers, etc., but it is permissible that a separate master node be constructed. Each node is a device implemented with the Power Line Communication Protocol C System.

Device Network ID and Device Node ID shall hereinafter be referred to as “DevNetID” and “DevNodeID,” respectively.

(1) Unique hardware address

Ten byte-long address information used to uniquely identify nodes.

(A) Home appliance manufacturer code: 2 bytes

0xFFFF and 0x0000 are reserved.

Code values shall be determined through consultations between manufacturers.

(B) Serial number: 8 bytes

(2) Address Settings

(A) Master node

DevNetID: Eight lowest-order bits of the unique hardware address

When the 8 lowest-order bits are 0, DevNetID shall be 1.

DevNodeID: 0xFF (fixed)

(B) Slave node (initial value)

DevNetID: 0xFF

DevNodeID: 0x01

(C) After slave node P&P

DevNetID: The DevNetID value assigned by the master node

DevNodeID: The DevNodeID value assigned by the master node

(3) Format for storing P&P-related unique information

The unique hardware address of the home node shall be stored in a memory. The following example assumes that an EEPROM is used as the memory.

The 2 highest-order bytes shall be the manufacturer code value, and the 8 lowest-order bytes shall be a manufacturer-defined value that allows the node to be uniquely identified.

(10 Bytes: The address area between 0x00 and 0x09 inclusive in the EEPROM shall be used.)

0x00	Unique hardware address of the home node （10bytes）	Reserved area
～0x80	Reserved area	

Fig. 10.15 Location of P&P-related Unique Information

The unique hardware address of the home node shall be stored before shipment of the product.

10.4.2 P&P Function Message

P&P function messages are messages that are used by the master node to set the network address (DevNetID) and node address (DevNodeID) settings of nodes whose DevNetID and DevNodeID settings have not been set.

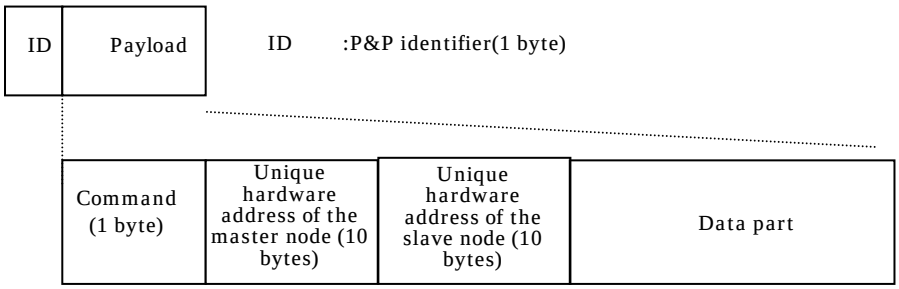


Fig. 10.16 P&P Function Message Format

(1) ID: P&P identifier (fixed at 0x55). This is used to distinguish P&P function messages from ordinary ECHONET protocol messages.

(2) Command part: Indicates the content of the message:

0x01: Slave node address setting request

0x11: Slave node address setting response

0x12: Address setting request

0x03: Slave node presence confirmation request (confirmation of whether the slave node is connected to the network)

0x13: Slave node presence confirmation response

(3) Unique hardware address of the master node

Manufacturer code (2 bytes) and serial number (8 bytes) of the master node.

When the unique hardware address of the master node is unknown, 0x0000, 0000, 0000, 0000 and 03FF shall be used.

(4) Unique hardware address of the slave node

When the node is a slave node, this is used to set the unique hardware address of that node.

When the node is a master node, this is used to set the unique hardware address of the target slave node for the P&P function.

(5) Data part

The composition of the data part differs depending on the command:

0x01: The 8 lowest-order bits of the network address field (2 bytes) and the 8 l owest-order bits of the node address field (2 bytes) are used.

0x11: The 8 lowest-order bits of the field (2 bytes) for the assigned network address and the 8 lowest-order bits of the field (2 bytes) for the assigned node address are used.

The values are the same as those set at the time of the slave node address setting request.

0x12: None

[Note] When the unique hardware address of the master node is unknown, 0x0000, 0000, 0000, 0000 and 03FF shall be used.

0x03: None

0x13: The 8 lowest-order bits of the network address field (2 bytes) and the 8 lowest-order bits of the node address field (2 bytes) are used.

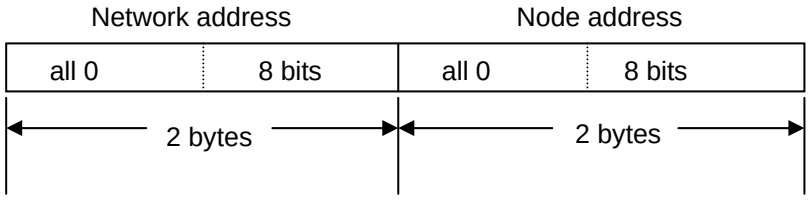


Fig. 10.17 Compositions of Addresses

10.4.3 P&P sequence

The P&P sequence is as follows:

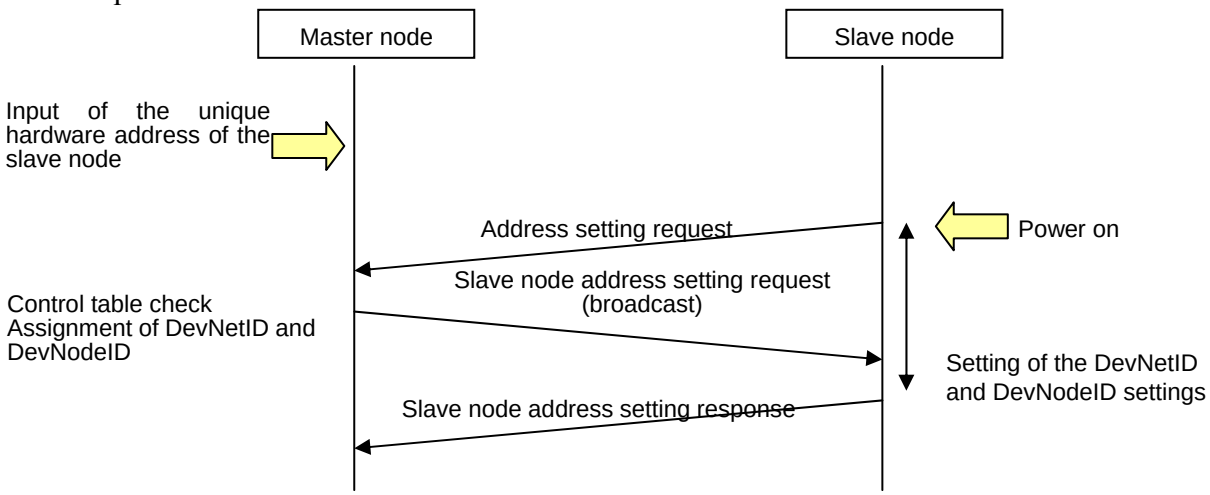


Fig. 10.18 P&P Sequence

(1) Sequence on the master node side

(A) The unique hardware address of the slave node is input.

(B) A P&P function message (address setting request) is received during execution of ECHONET protocol.

(C) A check is made to see whether or not the control table contains the unique hardware address of the slave node that has been input.

(D) If the control table contains the unique hardware address of the slave node that has been input, the previously assigned DevNetID and DevNodeID values are notified to the slave node by means of a slave node address setting request message.

(E) If the control table does not contain the unique hardware address of the slave node that has been input, a new address is assigned (The DevNodeID value range is between 0x01 and 0xFE inclusive).

(F) The master node registers the newly assigned address in the control table.

(G) The assigned DevNetID and DevNodeID values are transmitted by means of a slave node address setting request.

(H) The P&P sequence is completed with the reception of the slave node address setting response.

(2) Sequence on the slave node side

(A) Powers up after confirming that the unique hardware address of the slave node has been input by the master node (The LED for operation status confirmation comes on).

(B) Transmits an address setting request (The reception filter setting shall be “receive all” and the broadcasting mode shall be used for packet transmission).

(C) Waits for a slave node address setting request for the specified period of time.
If no request is received, a notification is given to the maintenance personnel by means of, for example, the turning off of an LED, and checks of the power supply etc. of the master node are made.

(D) After receiving the slave node address setting request, the slave node uses the DevNetID and DevNodeID values as its home address values.

(E) Completes the P&P sequence by sending (broadcast) a slave node address setting response.
Changes the reception filter setting to “only receive packets addressed to the home node.”

[Note 1] The reception filter is a PLC function to switch between the mode to receive all packets and the mode to only receive packets addressed to the home node.

[Note 2] The LED function is a recommended function.

10.4.4 Preconditions for performing P&P

P&P must not be performed unless the following preconditions have been satisfied:

(1) All unique hardware address information shall be stored in a memory such as an EEPROM before shipment by the device manufacturer.

(2) P&P-based address assignment shall be on a slave node-by-slave node basis.

(3) The master node shall have a function to input the unique hardware addresses of slave nodes and shall be in operation at all times.

(4) Each node shall have a function to switch its reception filter between the mode to receive all packets regardless of the address and the mode to only receive packets addressed to the home node.

The inputting of the unique hardware addresses of slave nodes by the master node also works as a kind of a home appliance password function to register devices, and only the registered devices can join the network.

Appendix 10.1 Specifications for Processing at the Protocol Difference Absorption Processing Section

The ECHONET protocol difference absorption processing section shall have the functions listed below. Messages shall be transmitted onto the power line after the setup in the payload part of the PTX command described in Chapter 10 is completed.

(1) Processing for the reception and recombining of messages
(2) Processing for the splitting and transmission of messages
(3) Address conversion processing
(4) Communication type conversion processing
(5) Common lower-layer communication interface processing

* Processing for the reception and recombining of messages and processing for the splitting and transmission of messages.

The processing for the splitting of messages and the processing for the recombining of messages shall in principle meet the specifications specified in “Part 2 ECHONET Communications Middleware Specifications,” “Chapter 7 Specifications for Processing at the Protocol Difference Absorption Processing Section” of the ECHONET Specification.

However, because the Power Line Communication Protocol C System has a processing function for the splitting and recombining of messages, there is in principle no need to consider processing for the splitting and recombining of messages.

In the Power Line Communication Protocol C System, each message that exceeds 120 bytes in length is automatically split into split messages upon reception, and the receiving side performs the processing to recombine the split messages. The Power Line Communication Protocol C System can handle messages with a message length of up to 1912 bytes.

In the case where it is not possible to provide a sufficient buffer size as specified in the ECHONET Specification, the existing ECHONET specifications (for the protocol difference absorption processing section) must be followed.

* Address conversion processing in the Power Line Communication Protocol C System

Because NodeID = MAC address, no conversion is required.

The lower-layer communication software section uses 11 bits of DevNodeID, but only the 8 lowest-order bits are notified to the overlying layer. In the ECHONET, managing NodeID’s as values up to 8 bits suffices for the purposes of address assignment.

However, in the Power Line Communication Protocol C System, “0x00” indicates a simultaneous broadcast address.

* Communication type conversion processing

Messages that has “0x00” as the MAC address value are processed as simultaneous broadcast messages.

Chapter 11 Specifications for the Power Line Communication Protocol D System

11.1 System Overview

The mechanical and physical specifications for the Power Line Communication Protocol D System are the same as the mechanical and physical specifications specified in "Chapter 2 Specifications for the Power Line Communication Protocol A System," and the electrical specifications for the Power Line Communication Protocol D System are basically same as the electrical specifications specified in Chapter 2 except that the transmission speed (The term "transmission speed" here means the effective speed; the physical speed is 9600 bps) must be changeable in 3 steps in the case of the Power Line Communication Protocol D System. Therefore, it is possible to implement both the Power Line Communication Protocol A System and Power Line Communication Protocol D System and switch between them as desired.

The major differences between the Power Line Communication Protocol D System and the Power Line Communication Protocol A System lie in the logical specifications. That is, the frame formats used in the Power Line Communication Protocol D System are different from those used in the Power Line Communication Protocol A System, and symbol synchronization and frame synchronization functions are more enhanced in the Power Line Communication Protocol D System. In addition, payload error correction encoding is also possible in the Power Line Communication Protocol D System. This allows more reliable communications than in the Power Line Communication Protocol A System to be achieved in terms of the resistance against degradations of transmission path characteristics (distortions, noises) of power lines.

The major differences between the Power Line Communication Protocol D System and the Power Line Communication Protocol A System lie in the logical specifications. That is, the frame formats used in the Power Line Communication Protocol D System are different from those used in the Power Line Communication Protocol A System, and symbol synchronization and frame synchronization functions are more enhanced in the Power Line Communication Protocol D System. In addition, payload error correction encoding is also possible in the Power Line Communication Protocol D System. This allows more reliable communications than in the Power Line Communication Protocol A System to be achieved in terms of the resistance against degradations of transmission path characteristics (distortions, noises) of power lines.

11.1.1 Scope of this chapter

These Specifications for the Power Line Communication Protocol D System consist of the mechanical and physical specifications, electrical specifications and logical specifications for Layer 1 and the logical specifications for Layer 2 and Layer 3. The

mechanical and physical specifications are the specifications for power lines to use and connectors, and the electrical specifications are the specifications for the modulation and demodulation sections. The logical specifications for Layer 1, Layer 2 and Layer 3 are the specifications for processing in Layer 1, Layer 2 and Layer 3 and signal interfaces between the 3 layers.

These Specifications for the Power Line Communication Protocol D System consist of the mechanical and physical specifications, electrical specifications and logical specifications for Layer 1 and the logical specifications for Layer 2 and Layer 3. The mechanical and physical specifications are the specifications for power lines to use and connectors, and the electrical specifications are the specifications for the modulation and demodulation sections. The logical specifications for Layer 1, Layer 2 and Layer 3 are the specifications for processing in Layer 1, Layer 2 and Layer 3 and signal interfaces between the 3 layers.

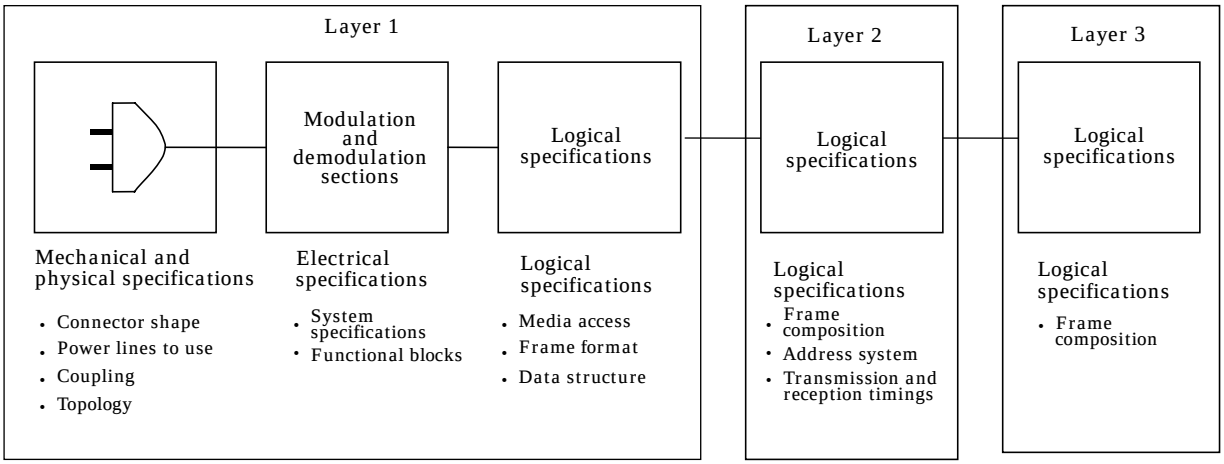


Fig. 11.1 Scope of the Specifications for the Power Line Communication Protocol D System

11.2 Mechanical and Physical Specifications

Refer to Chapter 2, “2.2 Mechanical and Physical Specifications.”

11.2.1 Connector shape

Refer to Chapter 2, “2.2.1 Connector shape.”

11.2.2 Power lines to use

Refer to Chapter 2, “2.2.2 Intended power line.”

11.2.3 Medium specifications

Refer to Chapter 2, “2.2.3 Medium specifications.”

11.2.4 Topology

Refer to Chapter 2, "2.2.4 Topology."

11.3 Electrical Specifications

With regard to the power line carrier system, the requirements specified in Article 46-2-6 ("Requirements for Special Carrier Type Digital Data Transmission Devices that Use a Spread Spectrum Method as the Carrier Modulation Method") of the Radio Law Implementation Rules (as of December 2000; the article number was changed on July 12, 1999 by Ministerial Ordinance No.60 of the Ministry of Posts and Telecommunications) shall be followed.

11.3.1 System specifications

(1)Spread spectrum method

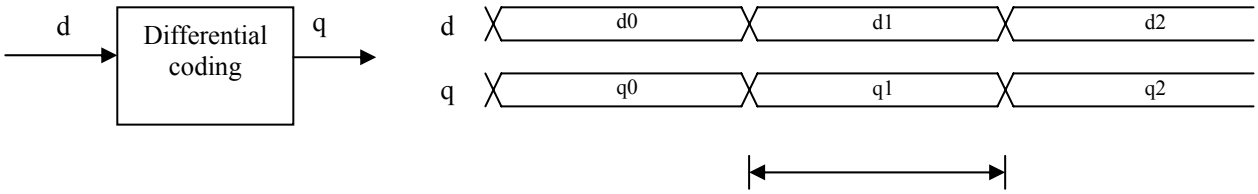
Direct spread spectrum method

Spread code: One-bit length for data shall match the spread code length.

The spread code series and chip length are not specified.

(2)Method of primary modulation

Differential coding



q(i) = q(i-1) xor d(i) 1/9600 sec

* xor: exclusive OR

(3)Transmission speed

9600bps±50ppm (no FEC)

4800bps±50ppm (FEC1)

2400bps±50ppm (FEC2)

(4)Carrier sense sensitivity

Input power: 0.1mW or less

(5)Transmission power

10mW/10kHz or less (The maximum allowable value is 120% of the rated value.)

(6)Spreading range

10kHz to 450kHz

(At the minimum, spreading in the 200 to 300kHz band must be achieved.)

(7)Spurious strength at the output terminal

5MHz or lower and higher than 450kHz: 56dB μ V or less

30MHz or lower and higher than 5MHz: 60 dB μ V or less

(8)Leakage electrical field (at a distance of 30m from the transmitter device)

(A) Spreading range frequency: 100 μ V/m

(B) 526.5kHz to 1606.5kHz: 30 μ V/m

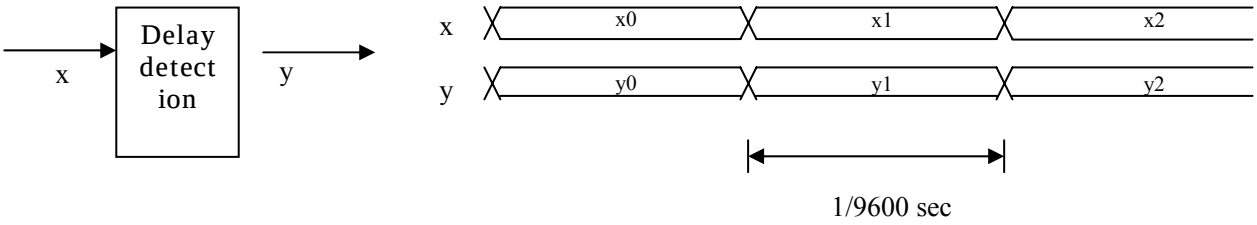
(C) Frequencies other than (A) and (B): 100 μ V/m

(9)Reception sensitivity

Input power: 0.1mW or less

(10)Detection method for the demodulation section

Delay detection



・x: Received spread signal or part of it — spread code reverse spreading and differential demodulation are performed simultaneously.

(A) When input x is a binary signal:

$y(i) = x(i)$ xor $x(i-1)$

(B) (B) When input x is a multi-value digital signal or an analog signal:

$$y(i) = \begin{cases} 0 \text{ (the phase of x (i) and the phase of x(i−1) are the same)} \\ 1 \text{ (the phase of x (i) and the phase of x(i−1) are opposite)} \end{cases}$$

・The term “multi-value digital signal” refers to a digital signal that has more voltage levels than binary signal or to a 2^k-value digital signal (k denotes an integer equal to or larger than 2) transmitted via a bus that consists of two or more binary signal lines.

Supplement 1 Example Compositions of the Modulation and Demodulation Sections

Power lines are not inherently designed to carry high-frequency signals that are used in communications. For this reason, noises, attenuations and impedance fluctuations due to the operation of home appliances occur.

Because the characteristics of power lines as transmission paths vary widely depending on where they are used, this ECHONET Specification provides latitude in selecting the method of demodulation by not specifying requirements for the method of demodulation. In addition, differences in the method of demodulation do not pose any problem from the standpoint of interconnection as well.

Supplement 1.1 Example Composition of the Modulation Section

Fig. 11.5 shows an example composition of the modulation section. The modulation section consists of a differential data coding block, a spread code generation block and a multiplication block for multiplying the differential coding data by the spread code.

As this ECHONET Specification does not specify spread code requirements, any desired spread code may be used.

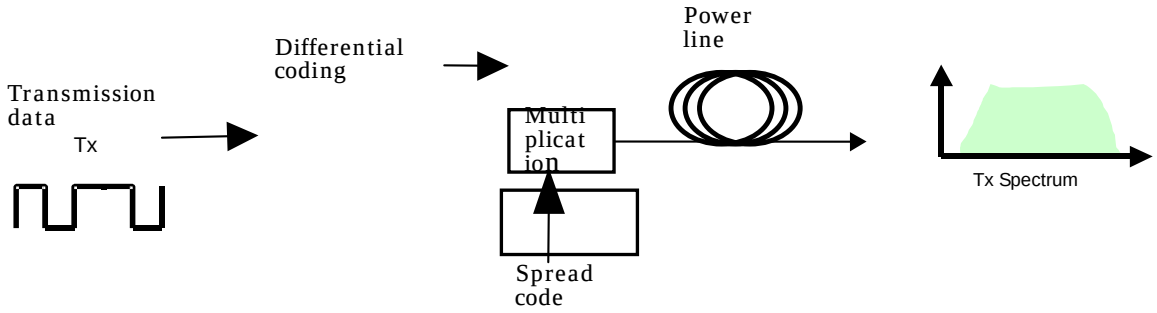


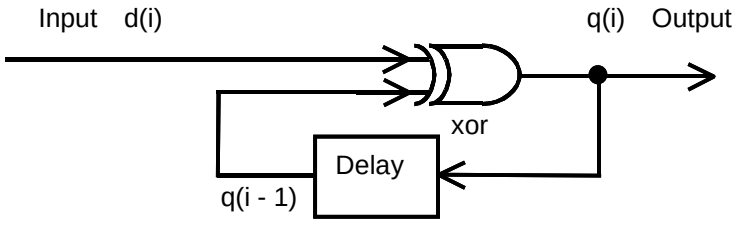
Fig. 11.5 Modulator Unit (Direct Spread Spectrum) Configuration Example

Supplement 1.2 Example of Differential Coding Block's Input and Output Data

Input data		1	1	0	1	0	1	1	0	0	1	0
Output data	(0)	1	0	0	1	1	0	1	1	1	0	0

* When the input data is "0", the immediately preceding output data is output as-is.
When the input data is "1", the immediately preceding data is inverted and then output.

A differential coding block configuration example is shown below:



Supplement 1.3 Demodulator Unit Configuration Example

Figure 11.6 shows a sub-band delay detection system as an example of a demodulator. This system uses the frequency diversity effect to obtain excellent receiving characteristics even in places with poor transmission characteristics.

As shown in Fig. 2.6, a received spectrum spread signal is frequency-divided using BPF 1 to n. The sub-band width and number of sub-bands are optional.

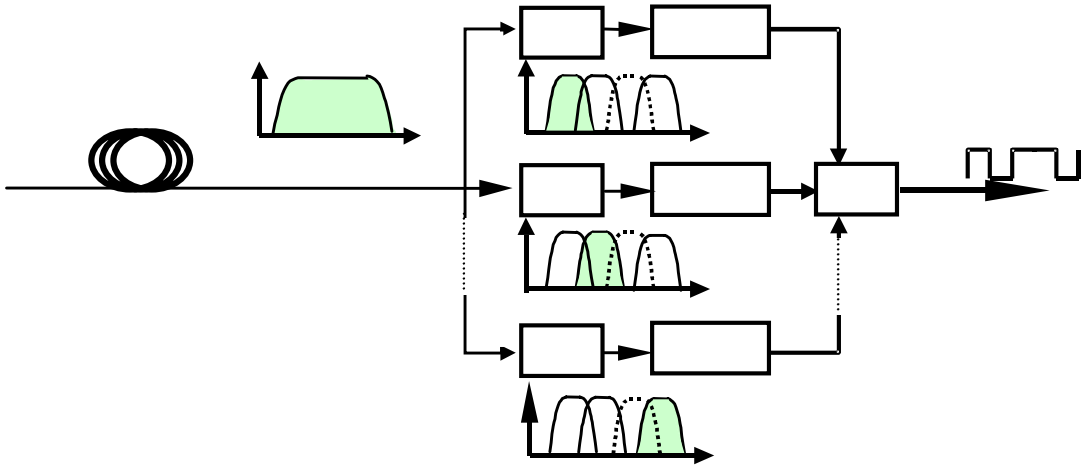


Fig. 11.6 Demodulator (Sub-band Delay Detection System) Configuration Example

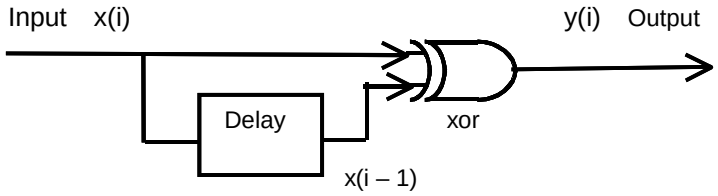
Supplement 1.4 Delay Detection Block Input/Output Data (Example 1)

(When the input is a binary signal and the spread code is 101)

Input signal	010	101	010	010	101	101	010	101	101	101	010	010
Delay signal		010	101	010	010	101	101	010	101	101	101	010
Output data		1	1	0	1	0	1	1	0	0	1	0

* The input signal and the immediately preceding input signal (delay signal) are XORred (exclusive-ORed) and then used as the output signal.

A delay detection block configuration example is shown below:



Supplement 1.5 Delay Detection Block Input/Output Data (Example 2)

(When the input is a multivalue digital signal*)

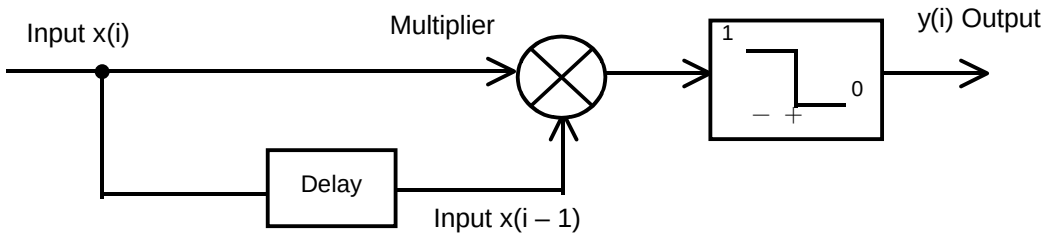
* Multivalue digital signal: Either a digital signal having a larger number of voltage

levels than a binary one or a 2^k-value digital signal (k = 2 or greater integer) transmitted by a bus comprising two or more binary signal lines.

Input signal	-1, 2, -1	1, -2, 1	-1, 2, -1	-1, 2, -1	1, -2, 1	1, -2, 1	-1, 2, -1
Delay signal		-1, 2, -1	1, -2, 1	-1, 2, -1	-1, 2, -1	1, -2, 1	1, -2, 1
Multiplication result		-	-	+	-	+	-
Output signal		1	1	0	1	0	1

* The input signal is multiplied by the immediately preceding input signal (delay signal). The obtained multiplication result is converted to a binary equivalent and then output.
- 1 when a minus sign is used (when the input signal is in opposite phase with the delay signal)
- 0 when a plus sign is used (when the input signal is in phase with the delay signal)

A delay detection block configuration example is shown below (implementation is also achievable for an analog input by using an analog circuit in the same configuration):



11.4 Logical Specifications

11.4.1 Layer 1

Frames used in the Power Line Communication Protocol D System are no different from frames used in the Power Line Communication Protocol A System in terms of the transmission control method and carrier sense function. In addition, frames used in the Power Line Communication Protocol D System are compatible with frames used in the Power Line Communication Protocol A System with regard to the dormancy period and Layer 1 frame composition. For these reasons, nodes equipped with lower-layer communication software that only meets the Specifications for the Power Line Communication Protocol A System and nodes equipped with lower-layer communication software that meets the Specifications for the Power Line Communication Protocol D System can coexist.

When a node transmits a Power Line Communication Protocol D System frame, nodes equipped with lower-layer communication software that only meets the Specifications for the Power Line Communication Protocol A System will not inhibit the Power Line Communication Protocol D System frame-based communications between other nodes, because nodes equipped with lower-layer communication software that only meets the Specifications for the Power Line Communication Protocol A System can recognize Power Line Communication Protocol D System frames through carrier sensing.

(1)Transmission control method

CSMA method

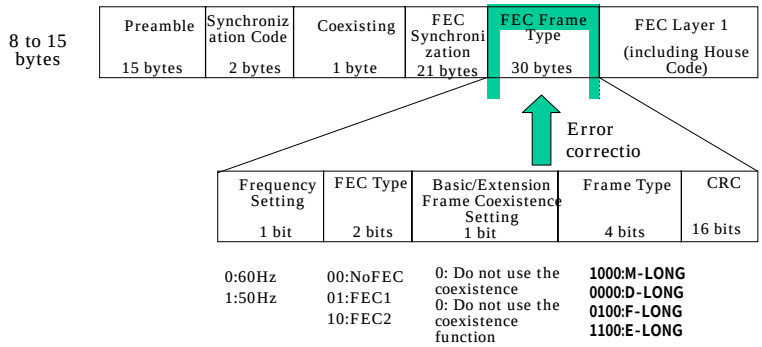
(2)Carrier sensing

Carrier sensing function is provided. Substitution is also possible.

(3)Dormancy period

Dormancy period for ordinary frames (excluding response signals and automatic retransmissions): 80ms or more.

(4)Layer 1 frame composition



(A)Preamble: symbol synchronization code

Preamble is used to synchronize the reception timing of the receiver device with the transmission timing of the transmitter device. In the Power Line Communication Protocol D System, Preamble lengths up to 15 bytes are permitted, so that a longer preamble than in the Power Line Communication Protocol A System can be provided to improve the reliability of communications.

Preamble: 010101….. 0101 (8 to 15 bytes)

(B)Synchronization Code: frame synchronization code

Synchronization Code is inserted between Preamble and the Frame Type field in order to indicate the beginning of the data. The Synchronization Code value shall be a fixed value. The synchronization code is modulated before transmission using the bit modulation method specified by the signaling method.

Synchronization Code: 1111010110010000

(C)Coexisting Frame Type:

Coexisting Frame Type is the same as the DOUBLE LONG Frame Type defined in the Specifications for the Power Line Communication Protocol A System. This allows nodes equipped with lower-layer communication software that only meets the Specifications for the Power Line Communication Protocol A System to recognize (through carrier detection) Power Line Communication Protocol D System frames as DOUBLE LONG frames for the Power Line Communication Protocol A System, thereby allowing transmissions to be prevented while DOUBLE LONG frame-based communications are being performed. Therefore, nodes equipped with lower-layer communication software that only meets the Specifications for the Power Line Communication Protocol A System do not inhibit the Power Line Communication Protocol D System frame-based communications between other nodes.

Coexisting frame type: 01100011 (1 byte)

(D)FEC Synchronization Code:

FEC Synchronization Code is a synchronization code that supports the FEC (forward error correction) technique. FEC Synchronization Code allows the beginning of the frame to be detected and frame synchronization to be achieved.

(E)FEC Frame Type: frame length/type definition code

This is a code generated by applying error correction coding to the 3-byte data that consists of “CRC,” “Frame Type,” “Frame Coexistence Setting,” “FEC Type ”and “Frequency Setting.” (30 bytes)

- Frequency Setting

  Frequency Setting specifies whether the power supply frequency is 50Hz or 60Hz. This allows error correction coding and coexistence pseudo synchronization code processing (which is described later) to be performed in a manner that is suitable for the frequency setting.

- FEC Type

  FEC Type allows the error correction coding method for the FEC Layer 1 payload to

be selected from between “No FEC,” “FEC1” and “FEC2.” If “No FEC” is specified, no error correction coding will be performed. “FEC1” is an error correction coding method with twice the degree of redundancy of “No FEC.” “FEC2” is an error correction coding method with four times the degree of redundancy of “No FEC.”

- Frame Coexistence Setting (Coexistence of Power Line Communication Protocol A System frames and Power Line Communication Protocol D System frames)

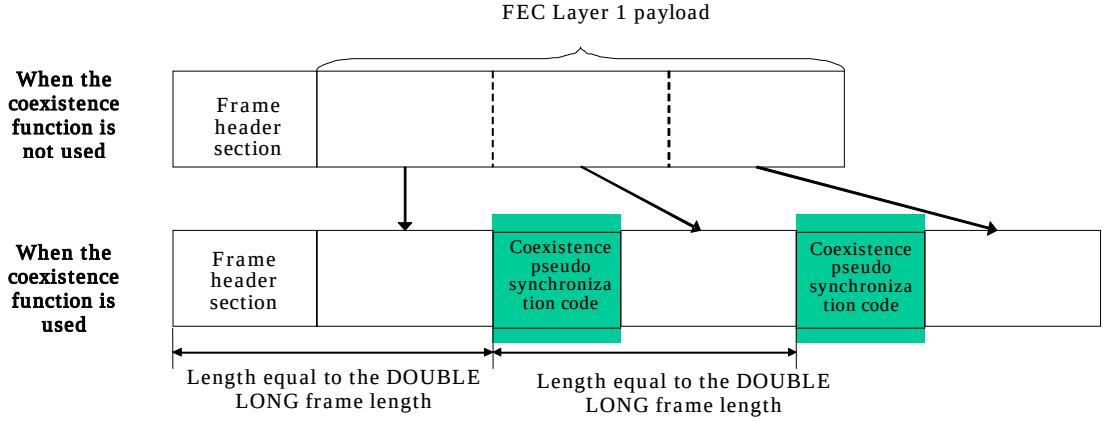
Frame Coexistence Setting specifies whether to use the function that allows Power Line Communication Protocol D System frames to coexist in the system with Power Line Communication Protocol A System frames. When a node equipped with lower-layer communication software that only meets the Specifications for the Power Line Communication Protocol A System is present, Frame Coexistence Setting must be set to “Use coexistence function” so that communications based on the Power Line Communication Protocol D System will not be inhibited by communications based on the Power Line Communication Protocol A System. The “Do not use the coexistence function” option must not be used unless all nodes in the system are nodes equipped with lower-layer communication software that meets the Specifications for the Power Line Communication Protocol D System.

Preamble	Synchronization Code	Coexisting Frame Type	Coexistence Dummy
8 bytes	2 bytes	1 byte	

Coexistence pseudo synchronization code

The coexistence pseudo synchronization code consists of “Preamble,” “Synchronization Code,” “Coexisting Frame Type” and “Coexistence Dummy.” Preamble is identical to the Preamble of the basic frame. The data size of Preamble is 8 bytes. Coexistence Dummy is a dummy code whose size is 1 byte when the Frequency setting is 50Hz and 9 bytes when the Frequency setting is 60Hz. The data content is not specified.

The figure below shows the frame composition for the case where the coexistence function is not used and the frame composition for the case where the coexistence function is used.



In the case where the coexistence function is not used, each frame consists of a frame header section and an FEC Layer 1 payload. In the case where the coexistence function is used, the FEC Layer 1 payload of each frame whose size (frame header section plus FEC Layer 1 payload) is larger than the Power Line Communication Protocol A System's DOUBLE LONG frame size is split into 2 split FEC Layer 1 payloads so that the size including the frame header section becomes equal to the Power Line Communication Protocol A System's DOUBLE LONG frame size, and a coexistence pseudo synchronization code is inserted between the 2 split FEC Layer 1 payloads. If the sum of the size of the inserted coexistence pseudo synchronization code and the size of the second split FEC Layer 1 payload is larger than the Power Line Communication Protocol A System's DOUBLE LONG frame size, the second split FEC Layer 1 payload is further split into 2 split FEC Layer 1 payloads so that the size including the coexistence pseudo synchronization code becomes equal to the Power Line Communication Protocol A System's DOUBLE LONG frame size, and a coexistence pseudo synchronization code is inserted between the 2 split FEC Layer 1 payloads. This is repeated until the sum of the size of the last split FEC Layer 1 payload and the size of the inserted coexistence pseudo synchronization code becomes equal to or less than the Power Line Communication Protocol A System's DOUBLE LONG frame size.

● Frame Type

Frame Type is used to select the frame type from between M-LONG, D-LONG, F-LONG and E-LONG.

● CRC

This is a CRC code that is given by calculating the one byte that contains the "Frequency Setting" to "Frame Type" bits.

CRC calculation formula:

Generator polynomial: $G(x) = X^{16} + X^{12} + X^5 + 1$ (CRC-CCITT Recommendations)

Twelve different frame compositions can be achieved by combining "Frame Type," "Frame Coexistence Setting" and "Frequency Setting" values of FEC Frame Type:

M-LONG frame (Use the coexistence function, 60Hz)

Preamble	Synchronization Code	Coexisting Frame Type	FEC Synchronization Code	FEC Frame Type	FEC Layer 1 payload
8 to 15 bytes	2 bytes	1 byte	21 bytes	30 bytes	120 bytes

M-LONG frame (Use the coexistence function, 50Hz)

Preamble	Synchronization Code	Coexisting Frame Type	FEC Synchronization Code	FEC Frame Type	FEC Layer 1 payload
8 to 15 bytes	2 bytes	1 byte	21 bytes	30 bytes	120 bytes

M-LONG frame (Do not use the coexistence function)

Preamble	Synchronization Code	Coexisting Frame Type	FEC Synchronization Code	FEC Frame Type	FEC Layer 1 payload
8 to 15 bytes	2 bytes	1 byte	21 bytes	30 bytes	120 bytes

D-LONG frame (Use the coexistence function, 60Hz)

Preamble	Synchronization Code	Coexisting Frame Type	FEC Synchronization Code	FEC Frame Type	FEC Layer 1 payload
8 to 15 bytes	2 bytes	1 byte	21 bytes	30 bytes	170 bytes

Preamble	Synchronization Code	Coexisting Frame Type	Coexistence Dummy	FEC Layer 1 payload
8 byte	2 byte	1 byte	9 byte	50 byte

Coexistence pseudo synchronization code

D-LONG frame (Use the coexistence function, 50Hz)

Preamble	Synchronization Code	Coexisting Frame Type	FEC Synchronization Code	FEC Frame Type	FEC Layer 1 payload
8 to 15 bytes	2 bytes	1 byte	21 bytes	30 bytes	168 bytes

Preamble	Synchronization Code	Coexisting Frame Type	Coexistence Dummy	FEC Layer 1 payload
8 byte	2 byte	1 byte	1 byte	60 byte

Coexistence pseudo synchronization code

D-LONG frame (Do not use the coexistence function)

Preamble	Synchronization Code	Coexisting Frame Type	FEC Synchronization Code	FEC Frame Type	FEC Layer 1 payload
8 to 15 bytes	2 bytes	1 byte	21 bytes	30 bytes	240 bytes

F-LONG frame (Use the coexistence function, 60Hz)

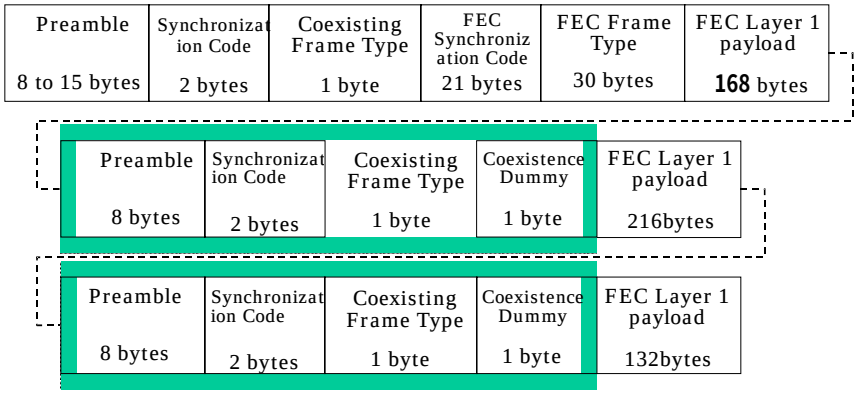
Preamble	Synchronization Code	Coexisting Frame Type	FEC Synchronization Code	FEC Frame Type	FEC Layer 1 payload
8 to 15 bytes	2 bytes	1 byte	21 bytes	30 bytes	**170** bytes

Preamble	Synchronization Code	Coexisting Frame Type	Coexistence Dummy	FEC Layer 1 payload
8 bytes	2 bytes	1 byte	9 bytes	210bytes

Preamble	Synchronization Code	Coexisting Frame Type	Coexistence Dummy	FEC Layer 1 payload
8 bytes	2 bytes	1 byte	9 bytes	120bytes

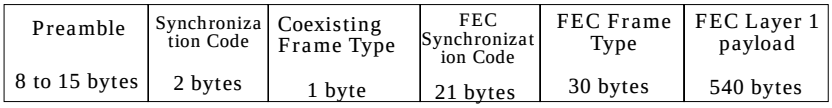
Coexistence pseudo synchronization code

F-LONG frame (Use the coexistence function, 50Hz)

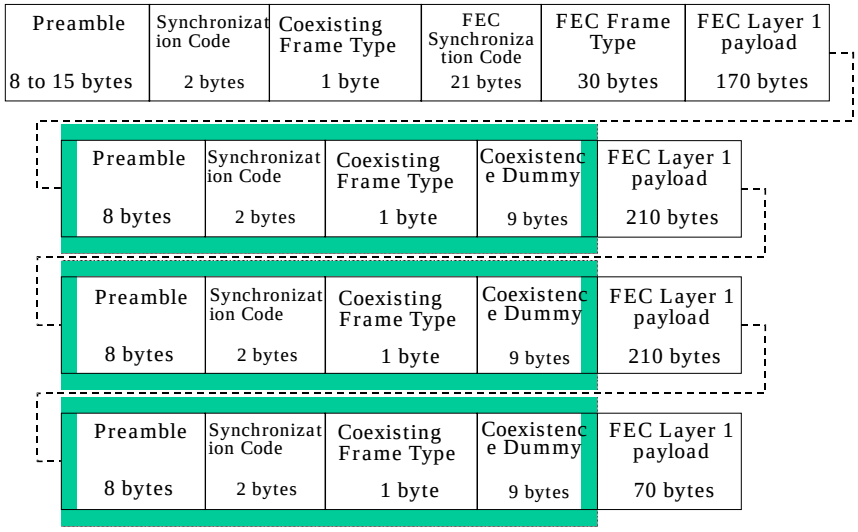


Coexistence pseudo synchronization code

F-LONG frame (Do not use the coexistence function)

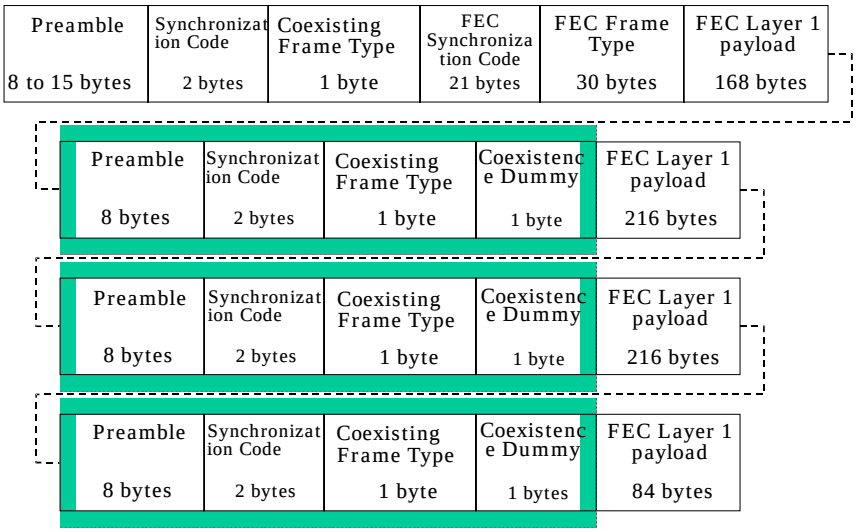


E-LONG frame (Use the coexistence function, 60Hz)



Coexistence pseudo synchronization code

E-LONG frame (Use the coexistence function, 50Hz)

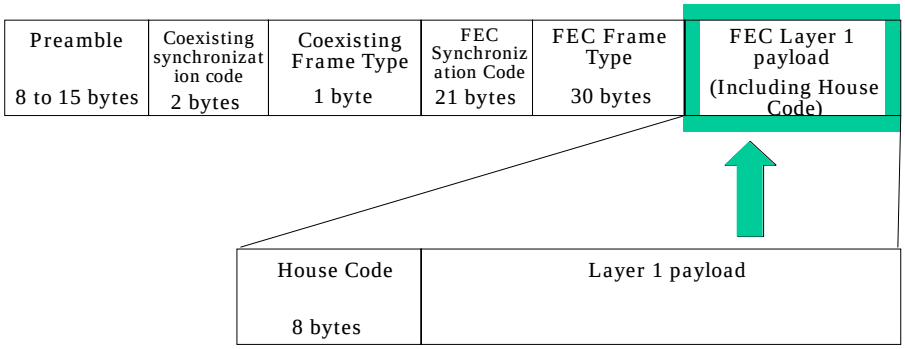


Coexistence pseudo synchronization code

E-LONG frame (Do not use the coexistence function)

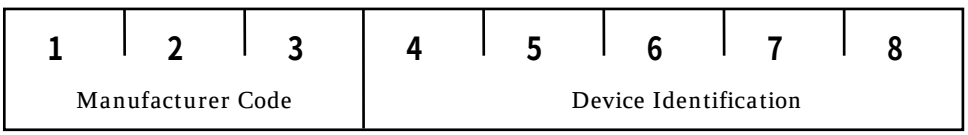
Preamble	Synchronization Code	Coexisting Frame Type	FEC Synchronization Code	FEC Frame Type	FEC Layer 1 payload
8 to 15 bytes	2 bytes	1 byte	21 bytes	30 bytes	720 bytes

(F)FEC Layer 1 payload:



The FEC Layer 1 payload consists of “House Code” and the “Layer 1 payload” when the “FEC Type” value is “No FEC.” When the “FEC Type” value is “FEC1” or “FEC2,” the FEC Layer 1 payload also consists of “House Code” and the “Layer 1 payload,” but error correction coding is applied to the message.

(G)House Code: household identifier



(a) Manufacturer Code
- The 3 highest-order bytes of the House Code value shall be used as the Manufacturer Code value.

(b) Device Identification Code
- The 5 lowest-order bytes of the House Code value shall be used as the Device Identification Code value.
- Each company owning a Manufacturer Code value shall manage Device Identification Code values associated with the Manufacturer Code value. Device Identification Code values shall be managed separately for individual Manufacturer Code values.
- Unique numbers (e.g. serial numbers) shall be assigned.

(c) P&P setup reservation code
- The House Code value specified below is reserved for House Code P&P setup as the common code value for all nodes.

  The value specified below must not be used for other than the transmission and reception of the announcement address 0 for P&P setup.
- Reservation code value: 0x0000000000000000

● Layer 1 payload

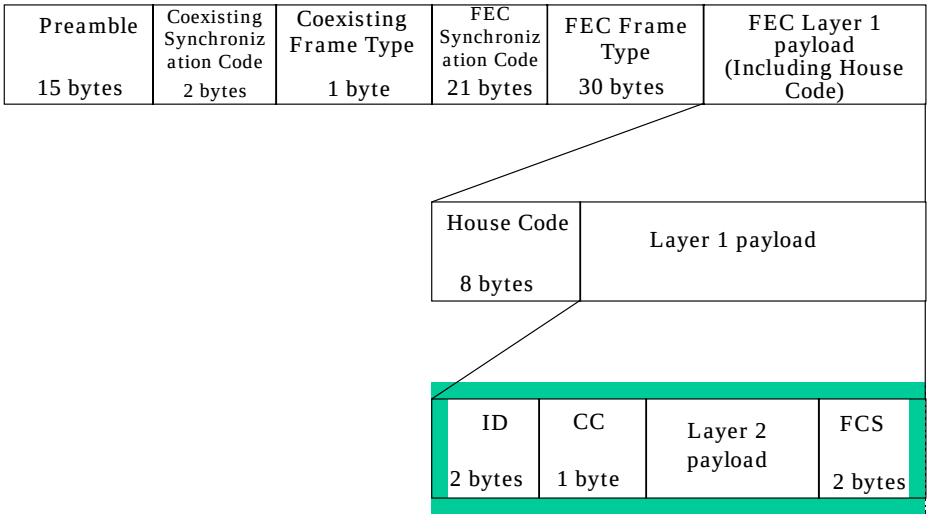
Refer to “11.4.2 Layer 2,” “(1) Layer 2 frame composition.”

* About details of FEC

This ECHONET Specification does not provide details of FEC, for confidentiality reasons. If it is necessary to confirm details of FEC to manufacture devices, a request shall be made to the ECHONET Consortium and a confidentiality agreement for the provision of the detailed information shall be individually concluded with the ECHONET Consortium.

11.4.2 Layer 2

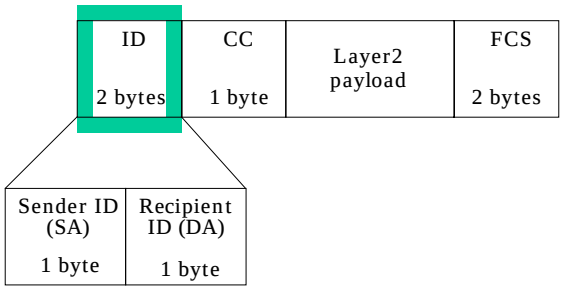
(1)Layer 2 frame composition



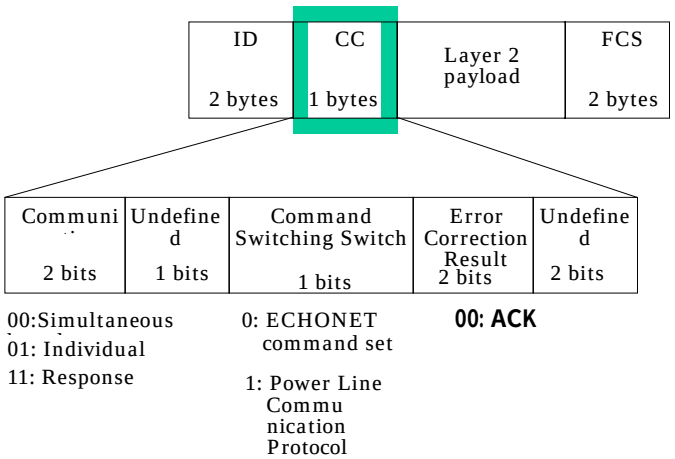
Layer 2 frame

(A) ID:

ID consists of the sender's terminal ID and the recipient's terminal ID (physical addresses). The lowest-order byte (NodeID) of the MAC address specified in "2.4.2 Layer 2," "(2) Layer 2 address system" shall be used as the terminal ID.



(B) CC: Control code



- Communication Type

  Communication Type specifies whether the frame to transmit is a simultaneous broadcast communication frame, an individual communication frame or a response to an individual communication frame.

  When "Response" is specified, an M-LONG frame that meets the Specifications for the Power Line Communication Protocol A System shall be used and "FEC Type" shall be set to "FEC2." Such a frame is called a response frame.
- Command Switching Switch

  Refer to "2.4.3 Layer 3," "(1) Layer 3 frame composition."
- Error Correction Result

  Error Correction Result indicates the result of the error correction of the received individual communication frame.

  In the case where the received individual communication frame has been corrected properly, a response shall be sent to the sender ID with Error Correction Result set to ACK. The response must be a response frame.

  In the case where the received individual communication frame has not been corrected properly, no response frame shall be returned and the received frame shall be discarded (because the sender ID value of the received individual communication frame is not guaranteed).

(C) Layer 2 payload:

Refer to "11.4.3 Layer 3," "(1) Layer 3 frame composition."

(D) FCS: frame check sequence

This is a CRC code that is given by calculating the "House Code" to "Layer 2 payload" bytes.

FCS calculation formula:

Generator polynomial: $G(x) = X^{16} + X^{12} + X^5 + 1$ (CRC-CCITT Recommendations)

(2)Layer 2 address system

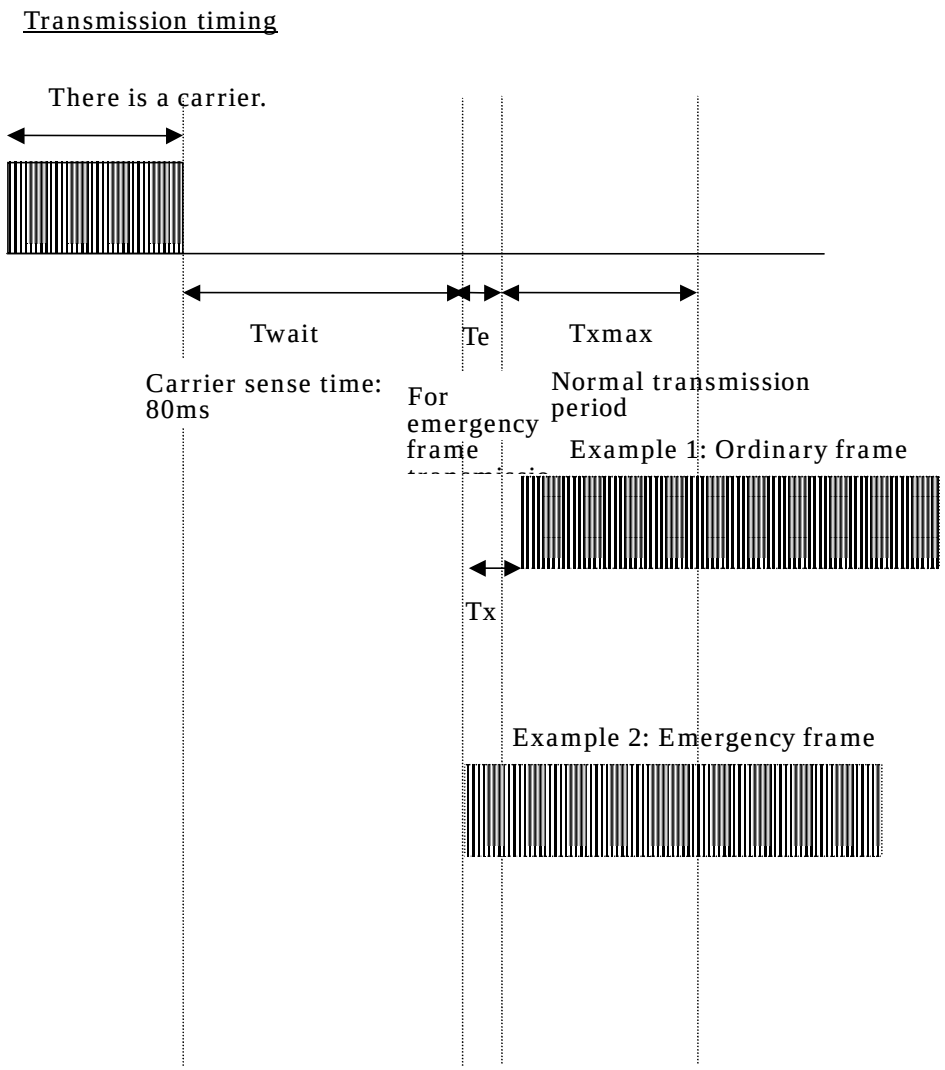
No.	Target	MAC address (HEX)	
1	Plug and play manager address	40	00
2	Individual communication address	40	01 to EF
3	Simultaneous broadcast address	40	F0
4	For future reserved	40	F1 to FE
5	Reserved for P&P	40	FF

The 8 highest-order bits of the terminal ID are fixed at 0x40 for the time being (To be expanded in the future).

The 8 lowest-order bits constitute the Node ID, which is part of the ECHONET address.

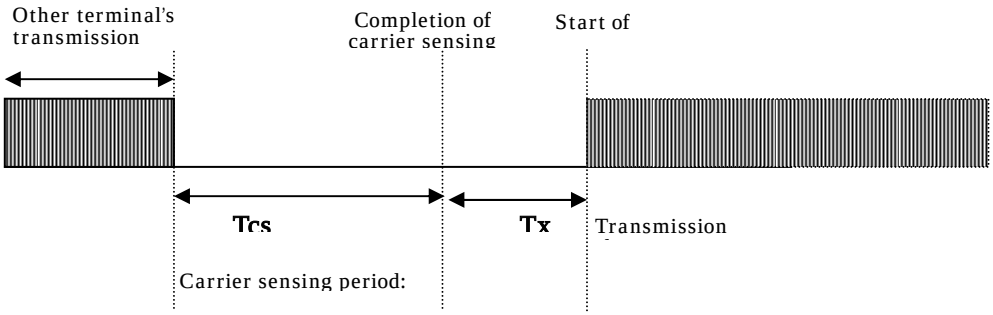
This ECHONET Specification does not permit the use of "For future reserved" addresses, but a means must be provided that allows "SA = For future reserved address" messages to be received in consideration of future use.

(3)Transmission timing



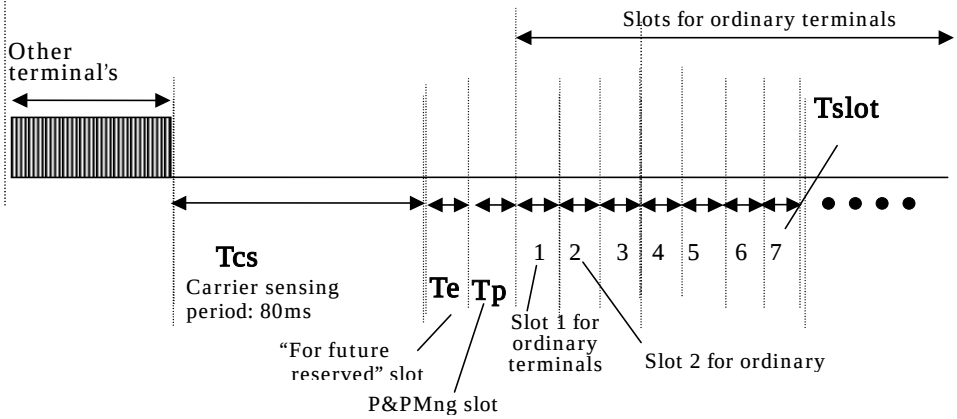
1. Transmission timing on the power line: Carrier sensing shall be performed for the Tcs = 80ms period in accordance with the Radio Law Implementation Rules.

   Because the transmission is made using the relevant transmission slot after the Tx slot waiting period elapsed following completion of the carrier sensing, the transmission is started "Tcs＋Tx" ms after the power line became clear of the carrier.



2. Transmission slot

   The transmission slots available are a) one “For future reserved” slot (This must not be used in communications based on this ECHONET Specification), b) one P&PMng slot and c) 100 slots for ordinary terminals. The timing requirements are as follows:



   The P&PMng slot shall only be used for P&P processing. For normal communications that do not involve P&P processing, slots for ordinary terminals shall be used.

3. Ordinary terminals determine which slot to use for transmission through the use of the magic number “Nmagic.”

   Nmagic is generated from terminal-specific information and non-terminal-specific information. The requirements for Nmagic are as follows:

   a) A different value shall be generated for each transmission.
   b) It shall be possible to output different values between terminals of the same type.

4. If different terminals try to use the same slot at the same time, a collision will occur. If each of the terminals does not have a collision detection function, 2 types of transmissions will be made.

If an error occurs on the reception side, the reception side's error processing procedure shall be followed.

Transmission Timing

The transmission timing on the power line is as shown in Fig. 11.7.

If the transmission side transmits an M-LONG, D-LONG, F-LONG or E-LONG frame, the reception side will receive it, and, if the received frame satisfies the specified conditions for Layer 1 and Layer 2 and the recipient ID (DA) value matches the lowest-order byte of the reception side's MAC address, the reception side shall start the transmission of a response frame onto the power line within the response frame transmission period "Tout1." The "Tout1" value shall be 72ms.

As shown in Fig. 11.8, if the above-mentioned conditions are not satisfied on the reception side, no response frame shall be returned. If the transmission side does not receive a response frame within the response frame reception period "Tout2" following the completion of the transmission of its frame, the transmission side shall retransmit the frame. In the case where retransmissions occur as a result of the loss of a frame due to a collision (messages transmitted by other terminals), the retransmissions shall use slots for ordinary terminals so that no further collision occurs, and each retransmission shall be made at the timing determined at random.

Up to 2 retransmissions may be made. If no response is received after the second retransmission, the transmission attempt shall be terminated. The "Tout2" value shall be 230ms ("extension response frame communication time (155msec.) ＋ Tout1").

[Exception for transmission timing] As shown in Fig. 2.9 of Chapter 2, in the case of a simultaneous broadcast or a transmission to a terminal with a provisional address, the reception side shall not return a response frame even if the received frame satisfies the specified conditions for Layer 1 and Layer 2.

Transmission timing

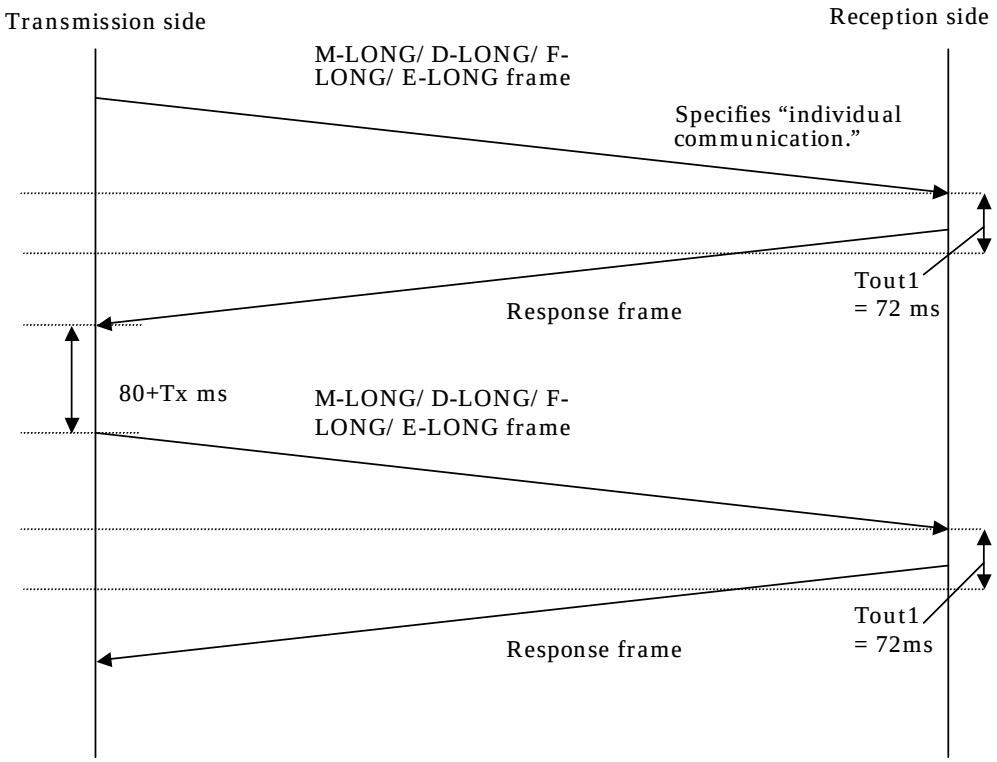


Fig. 11.7 Transmission Timing

Transmission timing (retransmissions)

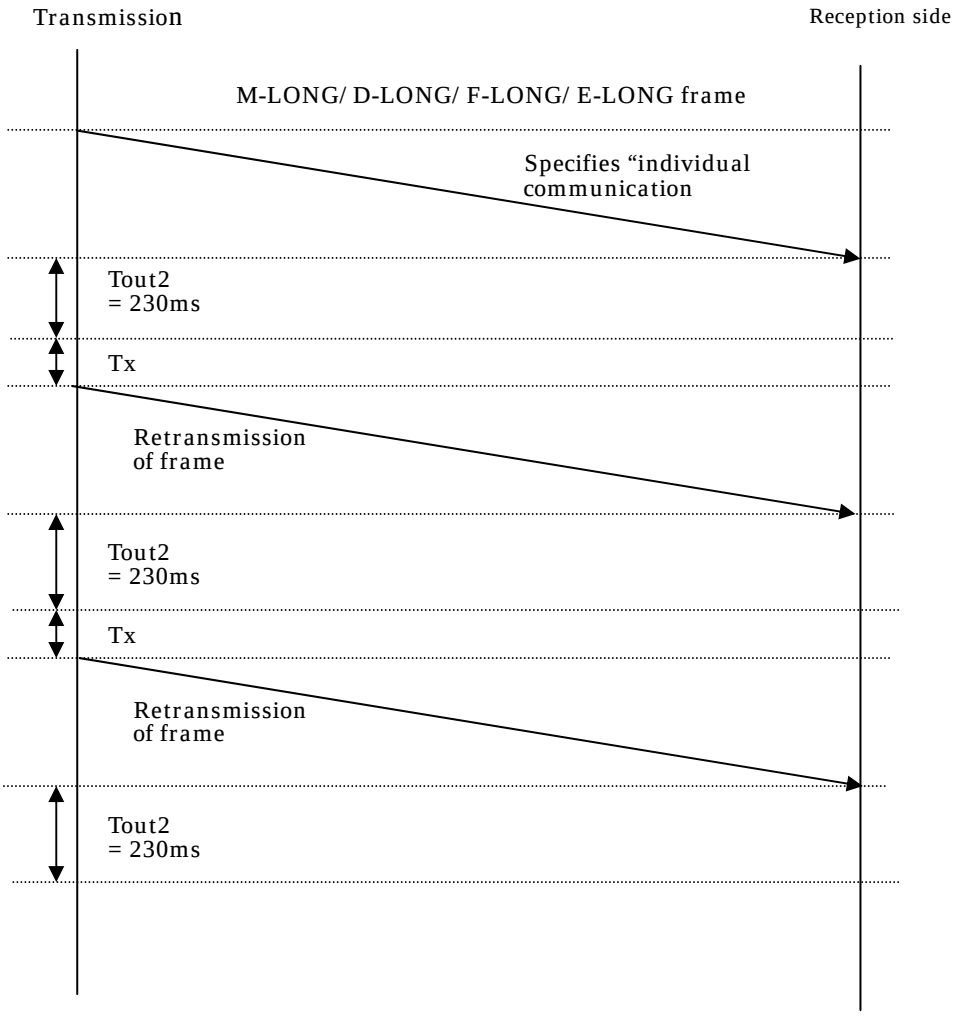


Fig. 11.8 Transmission Timing (Retransmissions)

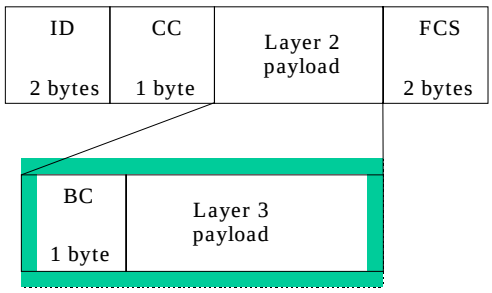
If the transmission side does not receive a response frame within the response frame reception period “Tout2” following the completion of the outputting of its frame, the transmission side shall retransmit the frame.

Retransmissions shall use slots for ordinary terminals, and each retransmission shall be made at the timing (Tx) determined at random.

The action to be taken when no response is received after the second retransmission shall be determined by the application.

11.4.3 Layer 3

(1) Layer 3 frame composition



(A)BC: effective byte counter

Indicates the number of effective bytes of the Layer 3 payload.

(B)Layer 3 payload:

The Layer 3 payload consists of ECHONET Message Counter (EDC)(1 byte) and ECHONET frames (EHD to EDATA).

When the “FEC Type” value is “FEC2,” error correction coding is performed at 4 times the degree of redundancy for “No FEC.” In the case where the size of the FEC Layer 1 payload after error correction coding exceeds the maximum FEC Layer 1 payload storage size for E-LONG frames, split ECHONET frames are created by splitting the ECHONET frame and prefixing ECHONET Message Counter (EDC). The FEC Layer 1 payload is created by applying error correction coding to these split ECHONET frames.

In the case where the size of the FEC Layer 1 payload after error correction coding does not exceed the maximum FEC Layer 1 payload storage size for E-LONG frames, the EDC value “0x88” (no splitting) is prefixed and error correction coding is performed without splitting the ECHONET frame.

For details of the processing for the splitting and transmission of messages using EDC and the processing for the reception and recombining of messages using EDC, refer to Part 2, “Chapter 7 Specifications for Processing at the Protocol Difference Absorption Processing Section.”

11.5 Basic Sequences

This section provides the following:

- State change diagram
- Explanations about the sequences for the states shown in the state change diagram

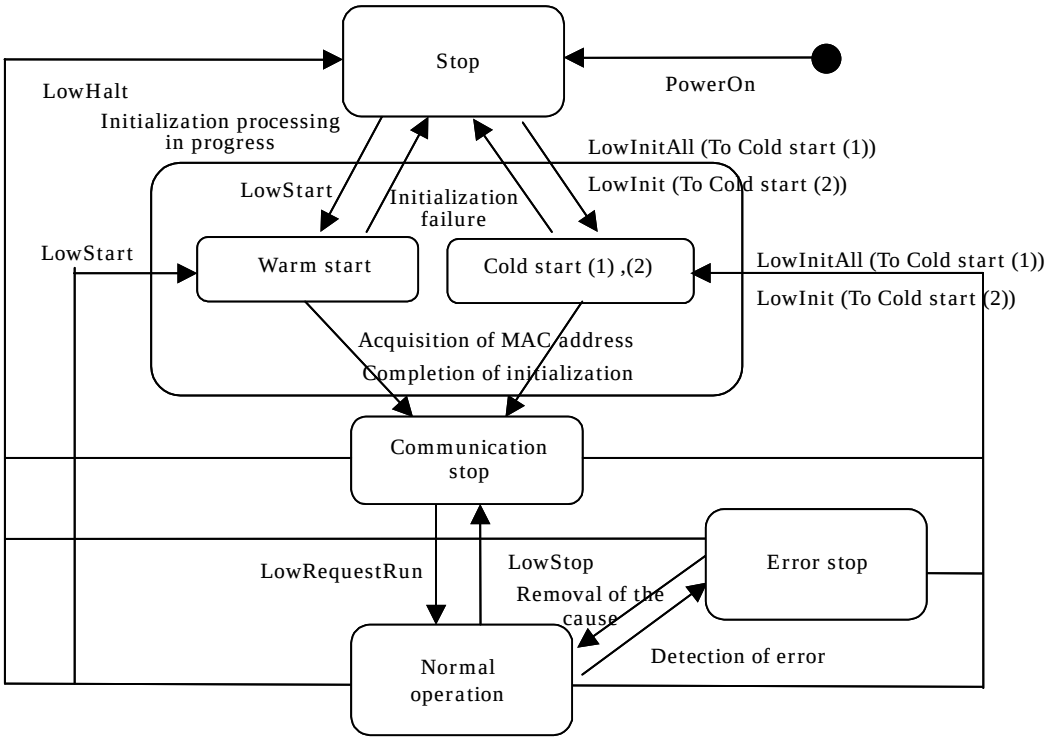
11.5.1 Basic concept

This subsection outlines the sequence for each of the following states of the individual

lower-layer communication software:

(1) Stop
(2) Initialization Processing in Progress
(3) Communication Stop
(4) Normal Operation
(5) Error Stop

The following diagram illustrates state changes between the above-mentioned states.



11.5.2 “Stop” status

The “Stop” status is a state in which the lower-layer communication software has stopped operating except for the P&P setup for functions unique to the lower-layer communication software which is started by, for example, the turning on of the power by the installer. The status immediately after power on is the “Stop” status. An overview of the processing to be performed immediately after the state change is given below together with a list of the individual lower-layer communication interface services that can be accepted during the “stop” state with brief explanations about the processing relating to the services.

(1) Trigger and the response behavior

Waits for an individual lower-layer communication interface service.

Transceiver initialization: The reset processing is performed immediately after power on.

(2) Status acquisition service (LowGetStatus)

Returns LOW_STS_STOP as the status.

(3) Lower-layer communication software type acquisition service (LowGetDevID)

Returns the type of lower-layer communication software.

The state change triggers are as follows:

(1) Trigger for a state change to the "initialization processing in progress" state:

Initialization service (LowStart, LowInit, LowInitAll)

11.5.3 "Initialization Processing in Progress" status

The "Initialization Processing in Progress" status is a state in which lower-layer communication software initialization is being performed. This is classified into Warm start, Cold start (1) and Cold start (2).

An overview of the processing to be performed immediately after the state change is given below together with a list of the individual lower-layer communication interface services that can be accepted during the "initialization processing in progress" state with brief explanations about the processing relating to the services.

(1) Trigger and the response behavior

Waits for an individual lower-layer communication interface service.

(2) LowStart (warm start)

If a MAC address value and a House Code value are stored, the initialization will be completed and a state change will be made to the "communication stop" state.

If a MAC address value and a House Code value are not stored, the initialization will fail and a state change will be made to the "stop" state.

(3) LowInitAll (Cold start (1))

If a MAC address value and a House Code value are stored, both will be discarded. Then, the steps described below for newly acquiring a House Code value and a MAC address value will be performed. If a MAC address value and a House Code value are acquired successfully, the initialization will be completed and a state change will be made to the "communication stop" state. If the attempt to acquire a MAC address value and a House Code value fails, the initialization will fail and a state change will be made to the "stop" state.

(a) Acquisition of a House Code value

A House Code value that uniquely identifies the power line communication protocol domain is acquired. This is achieved by means of manual setting by the installer using a dip switch, etc. or by means of acquisition from a sole plug and play manager (hereinafter referred to as "P&PMng") in the power line domain using the Register_ID function of the lower-layer communication software (which is explained in "2.6　P&P Setup of House Code and MAC Address").

(b) Acquisition of a MAC address value that is unique in the subnet

A MAC address value that is unique in the power line domain is acquired. As in the case of "① Acquisition of a House Code value," this is achieved by means

of manual setting by the installer using a dip switch, etc. or by means of acquisition from a sole plug and play manager (hereinafter referred to as "P&PMng") in the power line domain using the Register_ID function of the lower-layer communication software (which is explained in "2.6 P&P Setup of House Code and MAC Address").

(4) LowInit (Cold start (2))

If a MAC address value and a House Code value are stored, only the MAC address value will be discarded and a MAC address value will be newly acquired. The method is the same as that specified in "(3) LowInitAll (Cold start (1))," "② Acquisition of a MAC address value that is unique in the subnet." If a MAC address value is acquired successfully, the initialization will be completed and a state change will be made to the "communication stop" state. If the attempt to acquire a MAC address value fails, the initialization will fail and a state change will be made to the "stop" state. If a MAC address value and a House Code value are not stored, the initialization will fail and a state change will be made to the "stop" state.

(5) Status acquisition service (LowGetStatus)

Returns, as the status, LOW_STS_RST in the case of Warm start and LOW_STS_INIT in the case of Cold start (1) or Cold start (2).

(6) Lower-layer communication software type acquisition service (LowGetDevID)

Returns the type of lower-layer communication software.

The state change triggers are as follows:

(1) Trigger for a state change to the "communication stop" state.

Completion of initialization.

(2) Trigger for a state change to the "stop" state.

Initialization failure, power on, or an abnormal condition.

11.5.4 "Communication Stop" status

The "Communication Stop" status is a state in which an operation start request from the Communication Middleware is being waited for after completion of the initialization of the lower-layer communication software. An overview of the processing to be performed immediately after the state change is given below together with a list of the individual lower-layer communication interface services that can be accepted during the "communication stop" state with brief explanations of the processing relating to the services.

(1) Trigger and the response behavior

Waits for an individual lower-layer communication interface service.

(2) Status acquisition service (LowGetStatus)

Returns LOW_STS_CSTOP as the status.

(3) Lower-layer communication software type acquisition service (LowGetDevID)

Returns the type of lower-layer communication software.

(4) Physical address acquisition service (LowGetAddress)

Returns the MAC address.

(5) Profile data acquisition service (LowGetProData)

Returns the profile data.

The state change triggers are as follows:

(1) Trigger for a state change to the “normal operation” state

Operation start instruction service (LowRequestRun)

(2) Trigger for a state change to the “initialization processing in progress” state

Initialization service (LowStart, LowInit, LowInitAll)

(3) Trigger for a state change to the “stop” state

Stop service (LowHalt)

11.5.5 “Normal Operation” status

The “Normal Operation” status is a state in which a message is being transmitted to or received from the transmission medium (i.e. a state in which the primary function of the lower-layer communication software is being performed). An overview of the processing to be performed immediately after the state change is given below together with a list of the individual lower-layer communication interface services that can be accepted during the “normal operation” state with brief explanations about the processing relating to the services.

(1) Trigger and the response behavior

Waits for an individual lower-layer communication interface service.

(2) Status acquisition service (LowGetStatus)

Returns LOW_STS_RUN as the status.

(3) Message transmission service (LowSendData)

Converts the provided protocol difference absorption processing section message into a lower-layer communication software message and outputs it to the transmission medium (The message will be split into two more split messages before the conversion when the original message size requires splitting).

(4) Message reception service (LowReceiveData)

Converts the lower-layer communication software message received from the transmission medium into a protocol difference absorption processing section message and outputs it to the protocol difference absorption processing section.

(5) Lower-layer communication software type acquisition service (LowGetDevID)

Returns the type of lower-layer communication software.

(6) Physical address acquisition service (LowGetAddress)

Returns the MAC address.

(7) Profile data acquisition service (LowGetProData)

Returns the profile data.

The state change triggers are as follows:

(1) Trigger for a state change to the “initialization processing in progress” state

Initialization service (LowStart, LowInit, LowInitAll)

(2) Trigger for a state change to the “error stop” state

An error

(3) Trigger for a state change to the “communication stop” state

Stop service (LowStop)

(4) Trigger for a state change to the “stop” state

Stop service (LowHalt)

11.5.6 “Error Stop” status

The “Error Stop” status is a state in which the operation has been stopped as a result of an error. An overview of the processing to be performed immediately after the state change is given below together with a list of the individual lower-layer communication interface services that can be accepted during the “error stop” state with brief explanations about the processing relating to the services.

(1) Trigger and the response behavior

A state change to the “error stop” state will be made upon detection of an error, and error processing will be performed.

(2) Status acquisition service (LowGetStatus)

Returns LOW_STS_ESTOP as the status.

(3) Lower-layer communication software type acquisition service (LowGetDevID)

Returns the type of lower-layer communication software.

The state change triggers are as follows:

(1) Trigger for a state change to the “initialization processing in progress” state

Initialization service (LowStart, LowInit, LowInitAll)

(2) Trigger for a state change to the “normal operation” state

Removal of the cause of the error.

Causes of errors relating to redundant receptions of a House Code value shall be removed by means of manual House Code re-setup.

(3) Trigger for a state change to the “stop” state

Stop service (LowHalt)

11.6 P&P Setup of House Code and MAC Address

Relationship with the Power Line Communication Protocol A System

In the case where a Power Line Communication Protocol A System Specifications-compliant system and a Power Line Communication Protocol D System Specifications-compliant system coexist, the following requirements for plug and play managers (P&PMngs) and ECHONET nodes shall be satisfied:

(A) When the Power Line Communication Protocol A System Specifications and Power Line Communication Protocol D System Specifications coexist within nodes:

- Method of assignment from the plug and play manager (P&PMng)

  A plug and play manager (P&PMng) that is compliant with both the Power Line Communication Protocol A System Specifications and Power Line Communication Protocol D System Specifications shall sequentially assign the same House Code value and MAC address value to the ECHONET nodes of each system.

- ECHONET nodes

  ECHONET nodes compliant with both the Power Line Communication Protocol A System Specifications and Power Line Communication Protocol D System Specifications shall operate as ECHONET nodes having a House Code value and a MAC address value that are common to the Power Line Communication Protocol A System Specifications and Power Line Communication Protocol D System Specifications.

(B) When the Power Line Communication Protocol A System Specifications and Power Line Communication Protocol D System Specifications do not coexist within nodes (i.e. when systems made up of nodes compliant with the Power Line Communication Protocol A System Specifications operate independently of systems made up of nodes compliant with the Power Line Communication Protocol D System Specifications):

- Method of assignment from the plug and play manager (P&PMng)

  An independent plug and play manager (P&PMng) shall be provided for each of the Power Line Communication Protocol A System Specifications-based systems and each of the Power Line Communication Protocol D System Specifications-based systems, and each plug and play manager (P&PMng) shall assign its House Code and MAC address values to each ECHONET node.

- ECHONET nodes

  Each ECHONET node shall be set up in advance so that it will operate either as a Power Line Communication Protocol A System Specifications-compliant node or as a Power Line Communication Protocol D System Specifications-compliant node, and shall be assigned with a House Code value and a MAC address value by a plug and play manager (P&PMng) for a system which is based on the applicable Power Line Communication Protocol System Specifications.

The P&P setup procedure for assigning House Code and MAC address values to

ECHONET nodes that newly connected to a power line domain by means of the Register_ID function of the lower-layer communication software and a plug and play manager (P&PMng) is as follows.

This ECHONET Specification does not specify requirements for manual setup methods. However, the part of the Register_ID function that is associated with the processing for the side of ordinary nodes (i.e. nodes that are not equipped with the P&PMng function and nodes in which the P&PMng function is not effective) must be implemented, and it shall be possible to re-setup the House Code and MAC address using P&PMng even in the case where a House Code value and an MAC address value have already been set manually. However, ECHONET nodes whose House Code and MAC address have been set using P&PMng and ECHONET nodes whose House Code and MAC address have been set manually shall not coexist in the power line domain with the same House Code value.

(A) Man-machine requirements

- User operation requirements:

  A means for switching to the setting mode shall be provided.

- Supplementary explanations about user operation:

Below is an example of switching to the setting mode.

P&PMng: Pressing and holding a setting mode switch for several seconds

Ordinary node: Use of a reset switch, turning on the power when MAC address has not been set

- Indicator requirements:

  A node type indicator and an operation mode indicator shall be provided.

  However, the provision of a node type indicator is only compulsory for nodes equipped with the P&PMng function.

  This ECHONET Specification does not specify LED color requirements for LED's used as indicators. However, means shall be provided that allow users to distinguish between the node type indicator, the operation mode indicator and the LED indicator specified in "Part 7 ECHONET Communications Device Specifications," "Chapter 3 ECHONET Device Adapters," "3.3.2 Display section" (e.g. color coding, nameplates, etc.).

  The LED indication requirements are as follows:

	Lit	Unlit	Blinking
Node type	P&PMng	Ordinary node	
Operation mode	Setting mode	Normal operation	Setting error

In the case where non-LED indicators are used, they shall be capable of indicating the node types and operation modes shown in the table above in such a way that the user can clearly distinguish between them.

(B) House Code and MAC address in the setting mode

- House Code (HC) requirements

(C) Refer to “Operation of terminals.”

- MAC address requirements

P&PMng: “0x4000” shall be used.

However, ordinary nodes shall also accept the announcement address “0” of addresses other than “0x4000.”

The P&PMng shall have the correct House Code setting which was set manually by the installer at the time of deployment in the home domain or which was preset before shipment.

In addition, the P&PMng shall have MAC address values in advance in a number equal to the maximum number of eligible nodes.

Ordinary node: The value “0x40FF,” which is reserved for P&P, shall be used as the provisional address when MAC address has not been set.

(D) Operation of terminals

(1) Operation of P&PMng

- When a mode change to the setting mode is made through user operation, the P&PMng shall transmit the announcement address “0” at the T0 intervals. The House Code value for the announcement address “0”shall be 0x0000000000000000.
- A T0 value of around 700msec shall be used so that the traffic is not occupied by announcement address “0” frames. The operation mode indication shall be “setting mode.”
- The upper limit for the time the P&PMng operates in the setting mode shall be 5 minutes. When the 5 minute-period elapses, the P&PMng shall exit the setting mode and the operation mode indication shall change to “normal operation.”
- The P&PMng shall assign to the received Request_ID frame an unused MAC address value it selected from the MAC address table it manages (individual addresses: 0x4001 to 0x40EF) and send an ID notification to notify the sender of the Request_ID frame of the assigned MAC address value.

  The ID notification shall contain the terminal identifier uniquely identifying the terminal which was contained in the received Request_ID frame, and shall be sent to the provisional address (0x40FF). In the case where the recipient’s address (DA) is the provisional address, whether or not to take in the received frame shall be determined based on the terminal identifier.

(2) Ordinary node

- When a mode change is made to the setting mode, the node shall start waiting for the announcement address “0.” The operation mode indication shall be “setting mode.”

- When the predetermined period of time elapses, the node shall exit the setting mode and start operating in the normal operation mode. The operation mode indication for this state shall be “setting error.” The length of the “predetermined period of time” is not specified, but it is recommended that a period of 3 minutes or less be used so that incorrect settings by the P&PMng of a neighbor is prevented.
- The announcement address “0” shall be received for the $10T_0$ period or longer (This is called “overhear”).

  If the announcement address “0” from a P&PMng having the same House Code value is received 10 times consecutively, the House Code value contained in the announcement address “0” frame shall be acquired as the formal House Code value. Then, a Request_ID containing the terminal identifier shall be transmitted using this formal House Code value to request the P&PMng to assign an MAC address value.
- The node shall acquire a formal MAC address by receiving the ID notification, validate the already acquired formal House Code value and the formal MAC address value as its code values and then switch to the normal mode. The operation mode indication shall be “normal operation.” After completion of the communication sequence for the P&P setup of the House Code and MAC address, the formal House Code and MAC address values shall be stored in a non-volatile memory.
- If a MAC address value that falls outside the 0x4001 to 0x40EF range is presented by an ID notification, the address value shall not be received.
- If the node cannot receive an ID notification after transmitting a Request_ID, it shall retransmit the Request_ID.
- If the node receives the announcement address “0” from two or more P&PMngs before receiving an ID notification, the previously acquired House Code value shall be discarded and the operation mode indication shall change to “setting error.”

(E) Terminal identifier

A terminal attribute (such as the name of the manufacturer of the node, terminal type or the magic number “Nmagic” specified in “11.4.2 Layer2,” “(3) Transmission timing”) shall be used as the terminal identifier.

This ECHONET Specification does not specify requirements for terminal attribute types, but the terminal identifier must be an 8-byte code that allows terminals to be uniquely identified as far as possible.

(F) Communication sequence

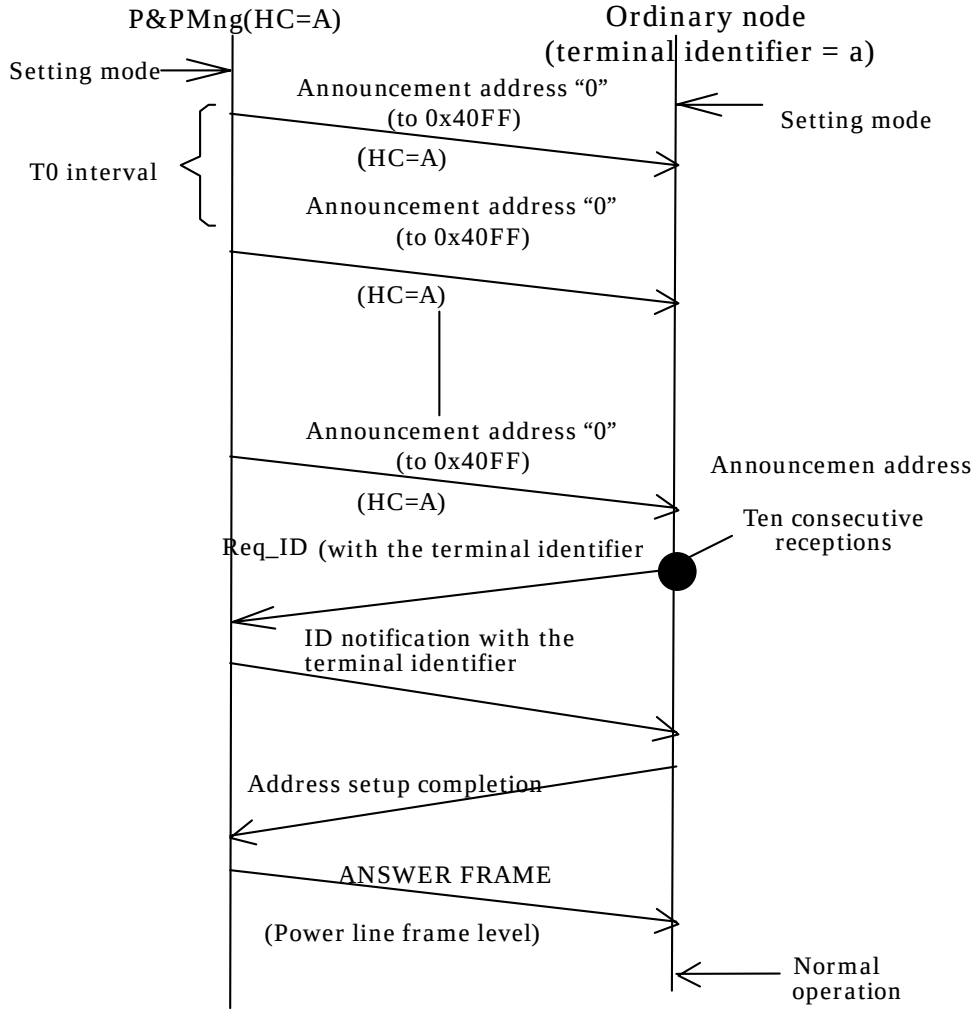


Fig. 11.9 Basic Register_ID Communication Sequence (1)

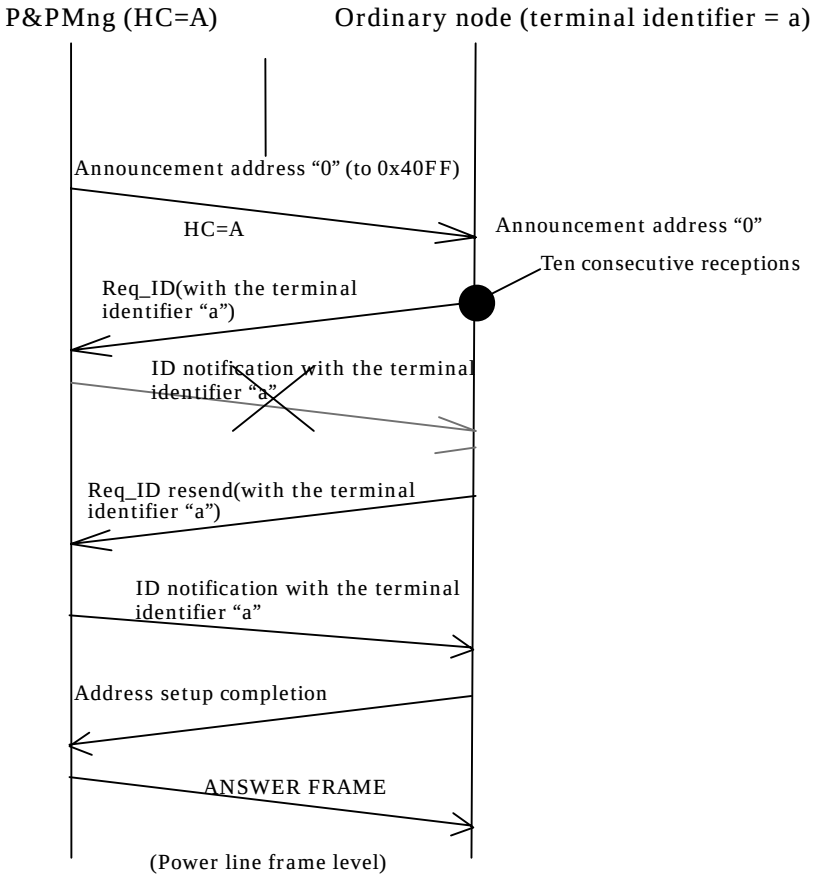


Fig.11.10 Retransmission Sequence for Req_ID, ID Notification, etc.

After the P&PMng transmits the announcement address “0” as described above, ECHONET Node A shall transmit a formal address assignment request Req_ID with the terminal identifier value to the P&PMng following the “overhear” process, using the formal House Code value it received from the transmission of the announcement address “0”. The P&PMng shall assign the formal address value “0x4001” to ECHONET Node A, which matches the terminal identifier value. Because the MAC address value of ECHONET Node A at this point is the provisional address value and it is necessary to send the formal address assignment command to the provisional address, the terminal identifier value shall be added to it. This prevents other ECHONET nodes whose address values at this point are the provisional address value from accidentally receiving the formal address assignment command.

Each ECHONET node whose MAC address value is the provisional address value shall send a formal address assignment request, and, if the terminal identifier value contained in the formal address assignment command matches its own terminal identifier value, shall receive the formal address assignment command and adopt the presented MAC address value as its MAC address value.

Each ECHONET node that has changed its MAC address value from the provisional address value to a formal address value shall send an address setup completion notification to the P&PMng using the formal House Code and MAC address values, and, upon receipt

of an ANSWER FRAME from the P&PMng to the address setup completion notification, complete the sequence (normal completion) after storing the formal House Code and MAC address values in a non-volatile memory.

The P&PMng shall assign a formal address value to each of the ECHONET nodes in this manner, referring to the formal address values and the terminal identifier values which are the unique terminal attribute values of the ECHONET nodes.

This ECHONET Specification does not specify requirements for terminal attribute types, but the terminal identifier must be an 8-byte code that allows terminals to be uniquely identified as far as possible (e.g. the name of the manufacturer of the node, terminal type, a magic number).

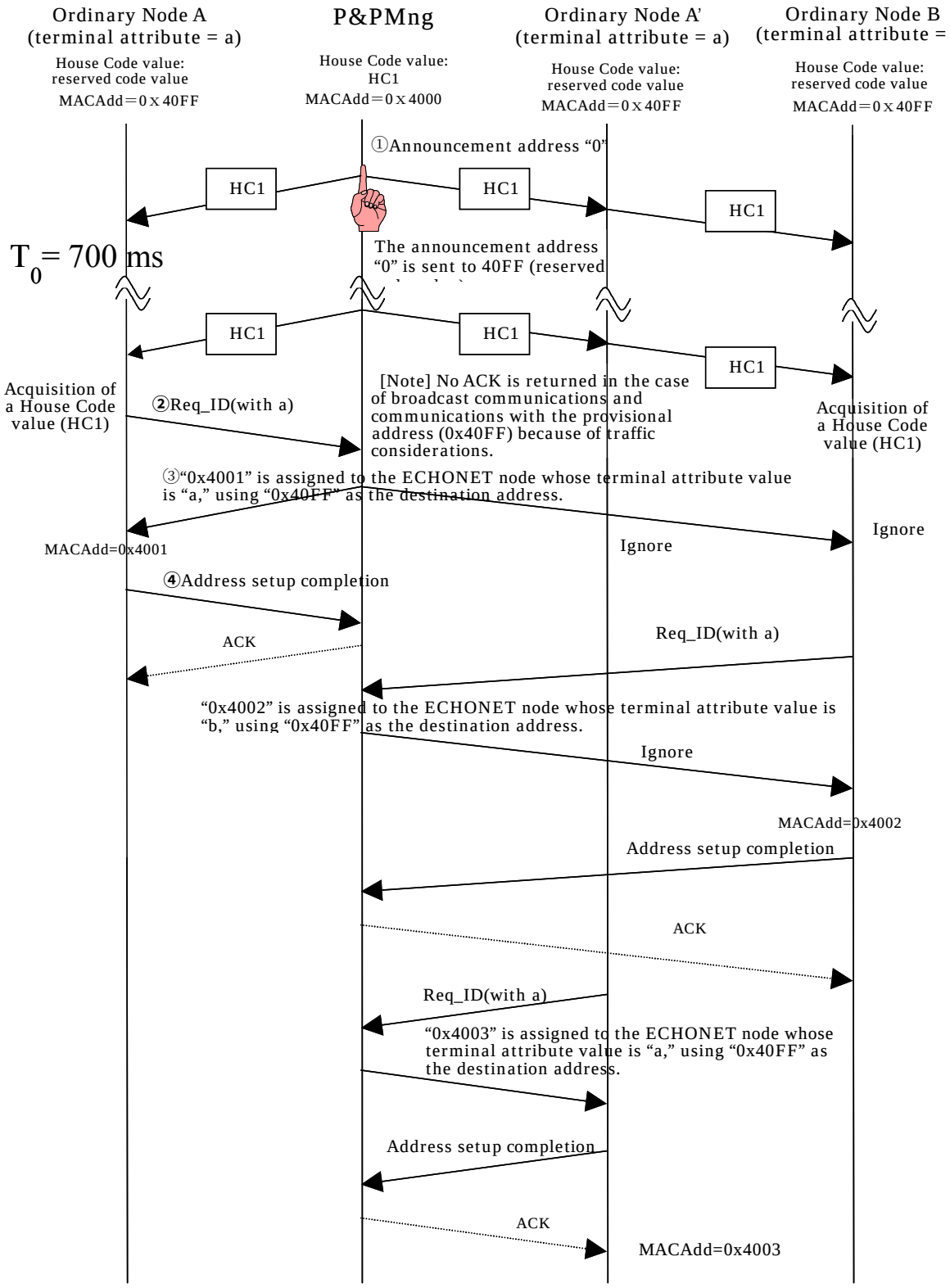


Fig. 11.11 Basic Register_ID Communication Sequence (2)

(G) P&P setup command set unique to power line communication protocols

The processing for the assignment of House Code and MAC address values by the

P&PMng (Register_ID) uses the unique command set described below. The Register_ID function is executed through a local command unique to power line communication protocols which is set apart from ECHONET commands by the Command Switching switch (Refer to "Supplement 2.1　Command Set Unique to Power Line Communication Protocols.")

For the processing sequence, refer to the Register_ID processing procedure shown in Fig. 11.9.

Ordinary nodes shall return an ACK to the P&PMng using an ANSWER FRAME after successful reception of "③ ID notification" and "④ Address setup completion."

The bracketed numbers in the STATE/DATA row in the tables below indicate the numbers of bytes.

Transmission of the "① Announcement address '0'" and "③ ID notification" commands, which are sent by the P&PMng, uses "highest-priority" slots.

The commands explained below are the commands for the case where an 8-byte magic number is used as the terminal identifier (terminal attribute). This ECHONET Specification does not specify requirements for terminal attribute types, but the terminal attribute should be an 8-byte code that allows terminals to be uniquely identified as far as possible.

As shown in Fig. 11.11, because the P&PMng uses a highest-priority slot (Refer to "11.4.2 (3) Transmission timing," "2. Transmission slot"), it can transmit an ID notification frame in response to the Request_ID from ECHONET Node A in preference to the Request_ID from ECHONET Node A', which means that no confusing situation will occur where two or more terminals having the same terminal attribute value (such as the same magic number value) have the same formal address value.

① Announcement address "0"

ATTRIBUTE		METHOD			STATE/DATA
TS	Property_name	methodtype	action	subtype	DATA
P&P	Housecode	INDICATE	do	normal	Housecode（8）
0x20	0x81	0x04	0x00	0x00	0x0123456789ABCDEF

② Request_ID

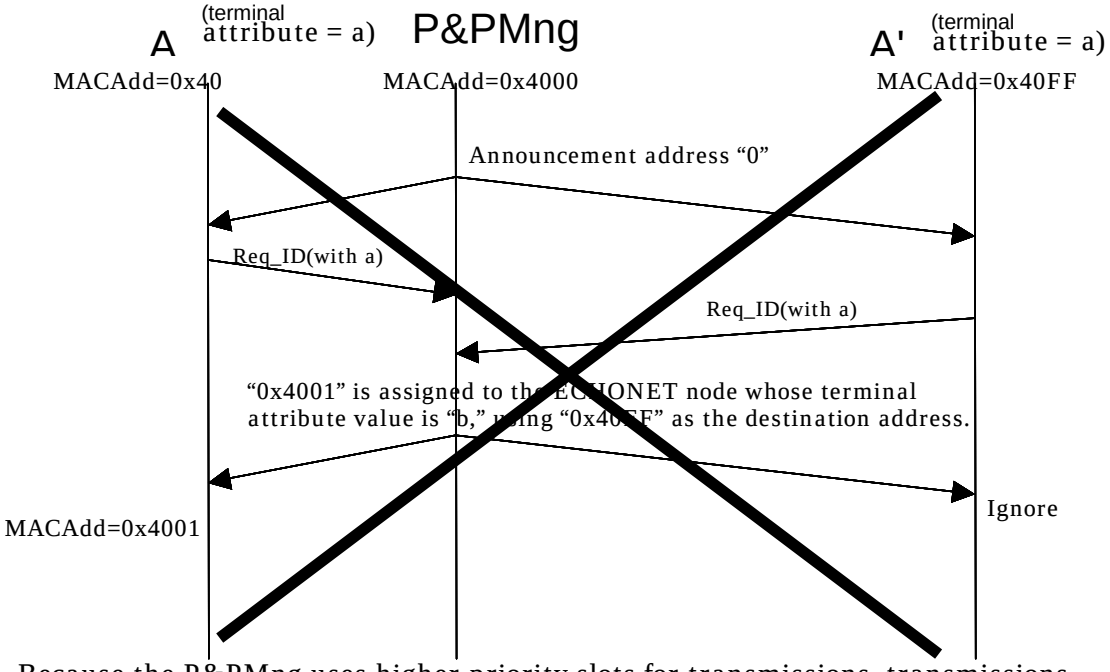
ATTRIBUTE		METHOD			STATE/DATA
TS	Property_name	methodtype	action	subtype	DATA
P&P	MacAddress	INQUIRE	do with housecode	normal	magic_number(8)
0x20	0x01	0x02	0x01	0x00	0x0123

③ ID notification

ATTRIBUTE		METHOD			STATE/DATA
TS	property_name	methodtype	action	subtype	DATA
P&P	MacAddress	WRITE	Do with housecode	normal	magic_number(8)、MacAddress(2)
0x20	0x01	0x01	0x01	0x00	‘0x0123,0x4010

④ Address setup completion

ATTRIBUTE		METHOD			STATE/DATA
TS	property_name	methodtype	action	subtype	DATA
P&P	MacAddress	RESPONSE	done	normal	MacAddress(2)
0x20	0x01	0x05	0x10	0x00	0x4010



Because the P&PMng uses higher-priority slots for transmissions, transmissions will not be made in the order shown in the figure above, which means that no confusing situation will occur where the same address value is assigned to two or more terminals having the same terminal attribute value
In addition, the “extended announcement address ‘0’” function is used as necessary to confirm the presence/absence and states of terminals.

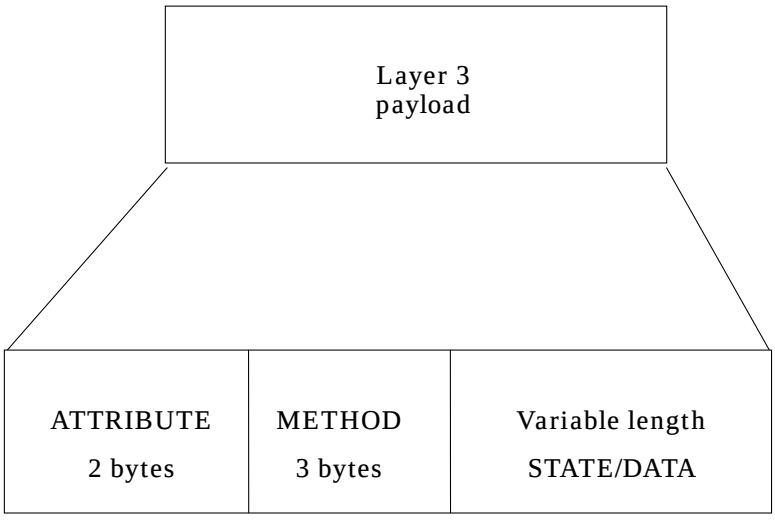
Fig. 11.12 Basic Register_ID Communication Sequence (An example of sequences that will not occur in practice)

Supplement 2.1 Command Set Unique to Power Line Communication Protocols

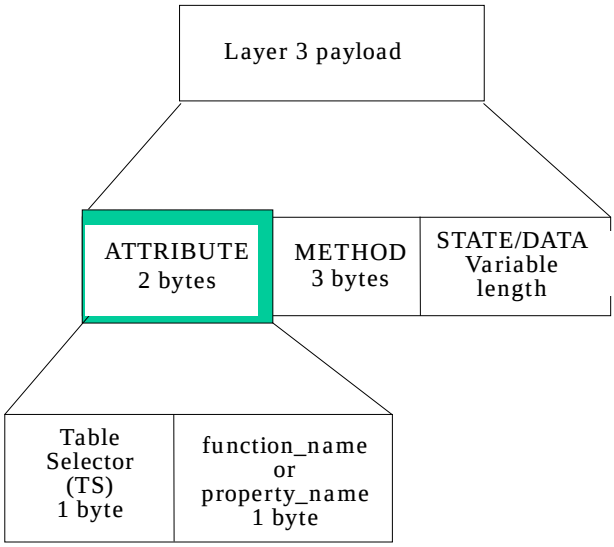
This section provides, as reference information, explanations on the structure of the command set unique to power line communication protocols, which consists of local commands other than ECHONET commands that are used in Layer 3 payloads.

The parameters of this command set are ATTRIBUTE (2 bytes), METHOD (3 bytes) and STATE/DATA (variable length). ATTRIBUTE specifies the control target and METHOD specifies the processing to be performed for the target specified by ATTRIBUTE.

The Register_ID function for the processing for the assignment of MAC address values by the plug and play manager described in “11.5 Basic Sequence,” “11.5.3 ‘Initialization processing in progress’ state” is executed through the use of this unique command set.



ATTRIBUTE consists of "Table Selector (TS)" (1 byte) and "function_name" or "property_name" (1 byte):

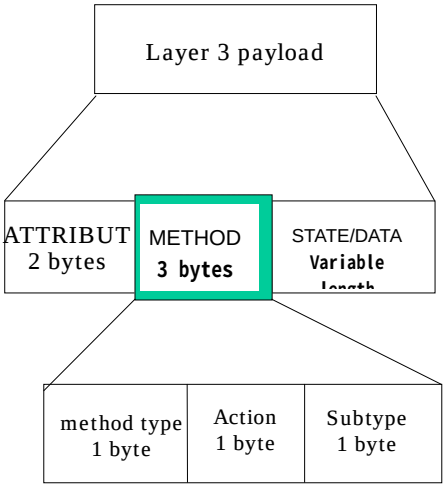


A list of the Table Selector options and a list of the property_name options for the case where Table Selector = P&P are given below as reference information.

Type	Table Selector	Table Selector value
Plug and play type	P&P	0x20

property_name	value
serial_number	0x00
MacAddress	0x01
magic_number	0x02
seed	0x03
maker	0x10
model	0x11
type	0x12
type_id	0x13
P&P	0x20

METHOD consists of “method type” (1 byte), “action” (1 byte) and subtype (1 byte):



A list of the commands is given below as reference information.

method type	Function	Remark	Value
READ	Read		0 x 00
WRITE	Write		0 x 01
INQUIRE	Request		0 x 02
RESET	Cancel a request		0 x 03
INDICATE	Present		0 x 04
RESPONSE	Respons		0 x 05
MAKE	Add an item	Optional	0 x 06
REMOVE	Delete an item	Optional	0 x 07
OPEN	Start connection	Optional	0 x 10
CLOSE	Terminate connection	Optional	0 x 11

action	Function	Remark	Value
do	Request for execution	For use in requests	0 x 00
do with housecode	Request for execution	For use in requests	0 x 01
do with certification method	Request for execution	For use in requests	0 x 02
done	Execution completed	For use in responses	0 x 10
cannot	Execution not possible	For use in responses	0 x 20
busy	Execution not possible at present	For use in responses	0 x 21
classified	Execution not possible (not eligible))	For use in responses	0 x 22

subtype	Function	Remark	Value
normal	Ordinary		0 x 00
with certification	Use authentication		0 x 01
with encryption	Use authentication and encryption		0 x 02

Supplement 2.2 Determination of the P&PMng to Use (Set_P&PMng)

Because it is possible that a leak of a P&PMng declaration from a neighbor may result in impersonation by the P&PMng of a neighbor, the "knockout" system whereby the last P&PMng that made a P&PMng declaration becomes the final P&PMng shall not be used.

P&PMng changes shall be made only when an installer intends to make such changes.

Supplement 2.3 P&PMng changes shall be made only when an installer intends to make such changes.

The "extended announcement address '0'" function is a function to ensure that ECHONET nodes are correctly assigned with MAC address values that are unique in the subnet.

This function allows the P&PMng to periodically check for ECHONET nodes to which it assigned MAC address values but which were removed thereafter and for ECHONET nodes to which it assigned MAC address values but which are currently not capable of performing communications because of distance- or noise-related problems on the power line or distortion problems.

This ECHONET Specification does not specify the method of making the above-mentioned checks, but an appropriate command shall be sent to the applicable devices and a check shall be made to confirm whether a response to the command has been received.

The MAC address values of the devices that were determined not to be present by the extended announcement address "0" function may be deleted from the P&PMng's list of registered addresses and returned to the list of unregistered addresses to make a larger number of MAC addresses available for assignment by the P&PMng. This ECHONET Specification does not specify the method of deleting MAC addresses.