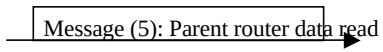
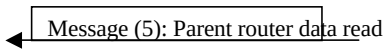
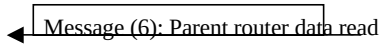
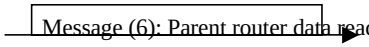
ECHONET Ver.2.11 Errata

Part	Chapter	Section	Paragraph	Before correction	After correction	Remarks
2	2	2		When the subnet changes due to a change in location of the ECHONET Node, its ECHONET address also changes.	When the subnet changes, the ECHONET address also changes.	Second line below the figure on page 2-1.
2	2	2		Specifying the ECHONET Node in an ECHONET Domain before and after movement can be performed using the device-unique data held in the Device Profile Object of each device (see Section 3.3, “Profile Objects”).	Specifying the ECHONET Node in an ECHONET Domain before and after movement can be performed using the device-unique data held in the Device Profile Object of each device (see Section 9.11.1, “Node Profile Class”).	Third line below the figure on page 2-1.
2	4	2	6	Objects for which detailed specifications (including property configurations) have already been formulated will be indicated with a ○ in the Remarks column, with the detailed specifications to be provided in the APPENDIX.	Objects for which detailed specifications (including property configurations) have already been formulated will be indicated with a ○ in the Remarks column, with the detailed specifications to be provided in the APPENDIX.	Sixth line from the bottom of the main text on page 4-11. No change in the English version because the term "konkai," deleted from the Japanese version, was not present in the previous English version.
2	4	2	7	In this case, the "response", "notification", or autonomous "notification" concerning the "request" related to two or more SEOJ-/EPC-stipulated properties is positioned in the CpESV.	In this case, the "response" or autonomous "notification" concerning the "request" related to two or more SEOJ-/EPC-stipulated properties is positioned in the CpESV.	The term "notification" was deleted from the eighth line on page 4-40.
2	4	2	8	In the case of a “request” (0x60,0x61), this indicates a request to write the content shown in EDT to the property stipulated in the EPC of the object stipulated in DEOJ.	In the case of a “write” (0x60,0x61), this indicates a request to write the content shown in EDT to the property stipulated in the EPC of the object stipulated in DEOJ.	First line of the main text on page 4-24.
2	4	2	8	In response to this “request,” when a value indicating a response is stipulated (0x61) and the request is to be (or has already been) received, “response” (0x71) is returned.	In response to this “write,” when a value indicating a response is stipulated (0x61) and the request is to be (or has already been) received, “response” (0x71) is returned.	Second line of the main text on page 4-24.
2	4	2	8	In the case of a “request” (0x64, 0x65), this indicates a request to write the value stipulated in the EDT (includes array element number and write request value data) of the property stipulated in the EPC of the object stipulated in the DEOJ.	In the case of a “write” (0x64, 0x65), this indicates a request to write the value stipulated in the EDT (includes array element number and write request value data) of the property stipulated in the EPC of the object stipulated in the DEOJ.	First line of the main text on page 4-27.
2	4	2	8	When the relevant object itself does not exist, neither “response” nor “response not possible” is returned. (See Fig. 4.8-2 for the exchange procedure.)	When the relevant object itself does not exist, neither “response” nor “response not possible” is returned. (See Fig. 4.8-2 for the exchange procedure.) In the case of an autonomous "notification", the DEA is set to a general broadcast for a required status change notification. Otherwise, however, the DEA can be set as desired regardless of whether "broadcast" or "individual" is selected.	Second line from the bottom of the text on page 4-29.
2	4	2	11	In this case, the "response", "notification", or autonomous "notification" concerning the "request" related to two or more SEOJ-/EPC-stipulated properties is positioned in the CpESV.	In this case, the "response" or autonomous "notification" concerning the "request" related to two or more SEOJ-/EPC-stipulated properties is positioned in the CpESV.	Second line of [1] on page 4-40.
2	4	2	11	(c) When the object relevant to the request does not exist	(c) When the object relevant to the request does not exist	Correction of a Japanese typo in (c) on page 4-43. There is no change to the English version.

Part	Chapter	Section	Paragraph	Before correction	After correction	Remarks
2	4	2	11	Fig. 4.12 shows the relationship between a write request requiring no response and write addition response for situations where request m cannot be accepted.	Fig. 4.12 shows the relationship between a write request requiring no response and write "process not possible" response for situations where request m cannot be accepted.	In the third line from the bottom of the text on page 4-43, 'write addition response', which is a translation of a typo, was corrected to 'write "process not possible" response.'
2	4	2	11	(b) A write "process not possible" response (CpESV = 0x51) will be returned.	(b) A write "process not possible" response (2)(CpESV = 0x51) will be returned.	Third line of (b) on page 4-45.
2	4	2	11	(c) When the object relevant to the request does not exist	(c) When the object relevant to the request does not exist	Correction of a Japanese typo in (c) on page 4-45. There is no change to the English version.
2	4	2	11	(c) When the object relevant to the request does not exist	(c) When the object relevant to the request does not exist	Correction of a Japanese typo in (c) on page 4-47. There is no change to the English version.
2	5	2	2	There are some operations (ESV = 0x60 to 0x6F) that an ECHONET node performs in relation to properties. The figure below shows the ECHONET node's basic sequence that is performed upon receipt of ESV = 0x60, 0x64, 0x68, 0x6A, or 0x6D (no response required):	There are some operations (ESV = 0x60 to 0x6E) that an ECHONET node performs in relation to properties. The figure below shows the ECHONET node's basic sequence that is performed upon receipt of ESV = 0x60, 0x64, 0x68, 0x6A, or 0x6D (no response required):	In the eighth line of the text on page 5-5, "0x6F" was corrected to "0x6E.."
2	5	2	2		It is necessary to provide an explanation of CpESV as well.	An explanation of CpESV should have been provided.
2	5	2	2	Fig. 5.5 shows the basic sequence, for each ESV, for an ECHONET node that has received a property value-related manipulation from another ECHONET node (ESV=0x60 to 0x6F), where ESV=0x61 to 0x63,0x65 to 0x67,0x69,0x6B,0x6C,0x6E (response required).	Fig. 5.5 shows the basic sequence, for each ESV, for an ECHONET node that has received a property value-related manipulation from another ECHONET node (ESV=0x60 to 0x6E), where ESV=0x61 to 0x63,0x65 to 0x67,0x69,0x6B,0x6C,0x6E (response required).	First sentence of the text on page 5-6. 0x6F was corrected to 0x6E.. (The Japanese corresponding to "from another ECHONET" was corrected to mean "from another ECHONET node," but there is no change to the English version because the error had not been reflected in the English version.)
2	5	3	2	Further, if the Net ID information provided by message (2) in Fig. 5.8-1 or message (2) in Fig. 5.8-2 does not match the own information, the process for an ECHONET node cold start is performed as described in the previous section.	Further, if the Net ID information provided by message (2) in Fig. 5.8-1 or message (3) in Fig. 5.8-2 does not match the own information, the process for an ECHONET node cold start is performed as described in the previous section.	Seventh line of the main text on page 5-10.
2	5	3	2		T2: Message (2) reception wait timeout lasting for period T2 (design guideline: 60 sec).	Addition of "T2" to the table in Fig. 5.8-2.
2	5	3	2	T2 elapsed No response	T3 elapsed No response	"T2"in Fig. 5.8-2 was corrected to "T3."
2	5	3	2	T2	T3	"T2"at the bottom of the table in Fig. 5.8-2 was corrected to "T3."
2	5	4		Fig. 5.9 Schematic of Master Router Data Fig. 5.9 Basic Sequence for Parent Router Cold Start Fig. 5.10-1 Basic Sequence for Parent Router Warm Start (Case 1) Fig. 5.10-1 Basic Sequence for Parent Router Warm Start (Case 2)	Fig. 5.9 Schematic of Master Router Data Fig. 5.10-1 Basic Sequence for Parent Router Cold Start Fig. 5.10-2 Basic Sequence for Parent Router Warm Start (Case 1) Fig. 5.10-3 Basic Sequence for Parent Router Warm Start (Case 2)	Two figures were titled Fig. 5.9. The second Fig. 5.9 was renamed Fig. 5.10-1, and Figs. 5.10-1 and 5.10-2 were renamed Figs. 5.10-2 and 5.10-3, respectively.
2	5	4	1	When two or more subnets are formed, only one parent router (Net ID server function) exists in a domain.	Only one parent router (Net ID server function) exists in a domain.	Part of the first sentence in the main text was removed.

Part	Chapter	Section	Paragraph	Before correction	After correction	Remarks
2	5	4	1	・Sets own SEA value with SEA.	・Sets the A side EA whose Net ID is 0x00 with SEA for the subnet A side, and sets the B side EA whose Net ID is 0x00 with SEA for the subnet B side.	Explanation of message (2) in the table in Fig. 5.9.
2	5	4	1	Stipulates general broadcast (0x01FF) with DEA.	Stipulates general broadcast (0x00FF) with DEA.	Explanation of message (3) in the table in Fig. 5.9 on page 5-14.
2	5	4	1	If a Net ID read response is returned within T3 (design guideline: 60 sec), the parent router (Net ID server function) stops operating (does not start up).	If a Net ID read response is returned within T3 (design guideline: 60 sec), the parent router (Net ID server function) does not start up.	Explanation of T3 in the table in Fig. 5.9 on page 5-14.
2	5	4	2	Figs. 5.10-1 and 5.10-2 show the parent router warm start basic sequence to be performed in Cases 1 to 3 on the presumption that the parent router is connected to one subnet. If the parent router is connected to two or more subnets, the number of transmissions of a message is equal to the number of connected subnets. Case 4 is explained under "Note" in Fig. 5.10-2.	Figs. 5.10-2 and 5.10-3 show the parent router warm start basic sequence to be performed in Cases 1 to 3 assuming that the parent router is connected to one subnet. If the parent router is connected to two or more subnets, the number of transmissions of a message is equal to the number of connected subnets. Case 4 is explained under "Note" in Fig. 5.10-3.	Last paragraph on page 5-15.
2	5	4	2	Sets own SEA value with SEA.	Sets the own EA whose Net ID is 0x00 with SEA.	Explanation of message (2) in the table in Fig. 5.10-1.
2	5	4	2	・Stipulates general broadcast (0x01FF) with DEA.	・Stipulates general broadcast (0x00FF) with DEA.	Explanation of message (3) in the table in Fig. 5.10-1.
2	5	4	2	Stipulates intra-domain general broadcast (0x00FF) with DEA.	Stipulates the message (2) SEA or intra-subnet general broadcast (0x01FF) with DEA (If there are two or more message (2)s, select as appropriate.)	Explanation of message (3) in the table in Fig. 5.10-2 on page 5-17.
2	5	4	2	If a response is received before the timeout, one of the sequences shown in Figs. 5.10-2 to 5.10-4 is performed.	If a response is received before the timeout, the sequence shown in Fig. 5.10-2 is performed.	Explanation of T3 in the table in Fig. 5.10-1 on page 5-16.
2	5	4	2	The starting point of T4 had been the receipt of message (6).	The starting point of T4 was corrected to the transmission of message (5).	The T4 arrow in Fig. 5.10-2 on page 5-17.
2	5	4	2	・Stipulates intra-domain general broadcast (0x00FF) with DEA.	・Stipulates intra-subnet general broadcast (0x01FF) with DEA.	Explanation of message (3) in the table in Fig. 5.10-2.
2	5	4	3	Reconnection to ECHONET	Connection to ECHONET	Upper section of Fig. 5.13.
2	5	4	3			Middle section of Fig. 5.13.
2	5	4	3			Middle section of Fig .5.13.
2	5	4	3	Message (7): Register request router EA write request	Message (7): Register request router data write request	Middle section of Fig. 5.13 on page 5-19.
2	5	4	3	Message (8): Router data write request	Message (8): Own router data write request	Middle section of Fig. 5.13 on page 5-19.
2	5	4	3	Message (9): Router data write response	Message (9): Own router data write response	Middle section of Fig. 5.13 on page 5-19.

Part	Chapter	Section	Paragraph	Before correction	After correction	Remarks
2	5	4	3		*5 Even if communication cannot be performed with the parent router, the normal node startup process is performed without activating the router functions.	Middle section of Fig. 5.13 on page 5-19.
2	5	4	3	Sets retained own SEA value as SEA.	For SEA, an undesignated default value (0x00) is set for the retained own SEA NetID.	Explanation of message (1) in the table in Fig. 5.13 on page 5-20.
2	5	4	3	Stipulates node profile object (0x0Ef001) with DEOJ.	Stipulates node profile object (0x0EF001) with DEOJ.	Explanation of message (3) in the table in Fig. 5.13 on page 5-20.
2	5	4	3	Stipulates notification request (0x63) with ESV.	Stipulates read request (0x62) with ESV.	Explanation of message (3) in the table in Fig. 5.13 on page 5-20.
2	5	4	3	Stipulates master router EA property (0xE3) with EPC.	Stipulates master router data property (0xE3) with EPC.	Explanation of message (5) in the table in Fig. 5.13 on page 5-20.
2	5	4	3	Sets subnet B EA value with SEA.	Sets subnet A EA or subnet B EA value with SEA.	Explanation of message (13) in the table in Fig. 5.13 on page 5-20.
2	5	4	3	・Sets subnet B EA value with SEA.	・Sets with SEA the subnet B EA value with the NetID set to 0x00.	Explanation of message (12) in the table in Fig. 5.13.
2	5	4	3	・Sets subnet B EA value with SEA.	・Sets the subnet A EA value with SEA for the subnet A side, and sets the subnet B EA value with SEA for the subnet B side.	Explanation of message (13) in the table in Fig. 5.13.
2	5	4	4	The processes performed in these cases can be classified into types A through D, which are explained below.	The processes performed in these cases can be classified into types A through E, which are explained below.	Fourth line of the main text on page 5-21.
2	5	4	4	Stipulates node profile object (0x0Ef001) with DEOJ.	Stipulates node profile object (0x0EF001) with DEOJ.	Explanation of message (3) in the table in Fig. 5.14 on page 5-23.
2	5	4	4	Stipulates notification request (0x63) with ESV.	Stipulates read request (0x62) with ESV.	Explanation of message (3) in the table in Fig. 5.14 on page 5-23.
2	5	4	4	Stipulates master router EA property (0xE3) with EPC.	Stipulates master router data property (0xE3) with EPC.	Explanation of message (5) in the table in Fig. 5.14 on page 5-23.
2	5	4	4	・Sets subnet B EA value with SEA.	・Sets with SEA the subnet B EA value with the NetID set to 0x00.	Explanation of message (14) in the table in Fig. 5.14.
2	5	4	4	・Sets subnet B EA value with SEA.	・Sets the subnet A EA value with SEA for the subnet A side, and sets the subnet B EA value with SEA for the subnet B side.	Explanation of message (15) in the table in Fig. 5.14.
2	5	5		(3) Basic sequence for Net ID resetup	(3) Basic sequence for Net ID write request receipt	Sixth line of the main text on page 5-24.
2	5	5	3		Message (9) ・Sets with SEA the subnet B EA value with the NetID set to 0x00.	An addition to the explanation on message (9) in the table in Fig. 5.17.
2	5	5	3	Message (8): Net ID write request	Message (9): Net ID write request	Message number (8) was corrected to (9) in the lower section of Fig. 5.17.
2	5	5	3	Message (9): Subnet B instance change class announcement	Message (10): Subnet B instance change class announcement	Message number (9) was corrected to (10) in the lower section of Fig. 5.17.
2	6	8		Required (N+R)		"Required (N+R)" in the Implementation Status column for M3(a) in the table on page 6-12 was deleted.
2	6	8		Required (N+R)		"Required (N+R)" in the Implementation Status column for M3(c) in the table on page 6-12 was deleted.

Part	Chapter	Section	Paragraph	Before correction	After correction	Remarks
2	7	7		Required		"Required" in the Implementation Status column for C1(a) in the table on page 7-9 was deleted
2	7	7		Required		"Required" in the Implementation Status column for C2(a) in the table on page 7-9 was deleted
2	7	7		Required		"Required" in the Implementation Status column for C5(a) in the table on page 7-10 was deleted
2	7	7		Required		"Required" in the Implementation Status column for C5(c) in the table on page 7-10 was deleted
2	9	9	1	Requests the redistribution of an ECHONET secure communication common key when a property value (ESV = 0x60, 0x61) is written to this property (0x00).	Requests the redistribution of an ECHONET secure communication common key when a property value (ESV = 0x61) is written to this property (0x00).	Second line of 9.9.1(2) on page 9-13.
2	9	11	1	○ in the "Mandatory" column for "Secure communication common key setup (Service Provider Key)."		Secure communication common key setup (Service Provider Key) is not mandatory.
2	9	11	1	○ in the "Mandatory" column for "Secure communication common key switchover setup (Service Provider Key)."		Secure communication common key switchover setup (Service Provider Key) is not mandatory.
2	9	11	1	Sets the ECHONET common key (User Key) switchover state that reflects the ECHONET common key (Service Provider Key) update by the Service Provider Key setup property.	Sets the ECHONET common key (Service Provider Key) switchover state that reflects the ECHONET common key (Service Provider Key) update by the Service Provider Key setup property.	The "Contents of Property" description for "Secure communication common key switchover setup (Service Provider Key)."
2	9	11	1	*1 Must be mounted when the secure communication function is implemented.	*1 Must be mounted when the secure communication function is implemented. However, the node that retains the secure communication common key setup node class must not accept a Set or SetM request.	Addition of a second sentence.
2	9	11	1	Common key distribution takes place when the property value (ESV = 0x62) is written into this property in an authentication/enciphered message format based on the node Serial Key or User Key of a common key (User Key) for User Level authentication/enciphered communication in an ECHONET secure communication proces.	Common key distribution takes place when the property value (ESV = 0x61) is written into this property in an authentication/enciphered message format based on the node Serial Key or User Key of a common key (User Key) for User Level authentication/enciphered communication in an ECHONET secure communication process.	First line of (21) "Secure communication common key setup (User Key)."
2	9	11	1	Common key distribution takes place when a property value element-stipulated write (ESV = 0x65) is performed in an authentication/enciphered message format based on the node Serial Key or Service Provider Key of a common key (Service Provider Key) for Service Provider Level authentication/enciphered communication in an ECHONET secure communication process	Common key distribution takes place when a property value element-stipulated write (ESV = 0x65) is performed in an authentication/enciphered message format based on the node Serial Key, User Key or Service Provider Key of a common key (Service Provider Key) for Service Provider Level authentication/enciphered communication in an ECHONET secure communication process	Second and third lines of (22) "Secure communication common key setup (Service Provider Key)."
2	9	11	1	User Key	Service Provider Key	(22) Second line from the bottom of (22) "Secure communication common key setup (Service Provider Key)(in two places)."

Part	hapt	ectio	agra	Before correction	After correction	Remarks
2	9	11	1	property value write (ESV = 0x61)	property value write (ESV = 0x62)	(23) Second line from the bottom of (23) "Secure communication common key switchover setup (User Key)(in two places)."
2	9	11	2	When there are two or more nodes within a device, each node has this profile class. However, these nodes must have common values for all properties other than the Net ID (0xE1) because they constitute a single device.	When there are two or more nodes within a device, each node has this profile class. However, these nodes must have common values for all properties other than the Net ID (0xE1), Operating status (0x80), Communications definition for malfunction status (0x88), and Communications definition for malfunction content (0x89) because they constitute a single device.	Second note below the table.
2	9	11	2		It does not function as a router unless a value other than 0x00 is specified by the NetID server.	Addition to the text of (3) "Current router data."
2	9	11	2	This property is mandatory for an ECHONET node having Net ID server functions.	This property is held only by ECHONET nodes having Net ID server functions	Correction of (6) "Registration request router data."
2	9	11	3	Transition state（EPC：0x8E）	Transition state（EPC：0x8F）	"0x8E" has already been assigned to the date of manufacture ("Profile object super class)."
2	9	11	4	Transition state（EPC：0x8E）	Transition state（EPC：0x8F）	"0x8E" has already been assigned to the date of manufacture ("Profile object super class)."
2	9	11	5	Transition state（EPC：0x8E）	Transition state（EPC：0x8F）	"0x8E" has already been assigned to the date of manufacture ("Profile object super class)."
2	9	15		Array element number mask value: 1 byte	Array element number mask value: 2 bytes	The figure entitled "When the associated property is of an array type" under the heading "Fire Sensor Class Communications Definition Objects for Linked (Action) Settings: Detailed Specifications."
2	9	15		Masked array element number: 1 byte	Masked array element number: 2 bytes	The figure entitled "When the associated property is of an array type" under the heading "Fire Sensor Class Communications Definition Objects for Linked (Action) Settings: Detailed Specifications."
2	9	17		0x00 write（ESV＝0x60, 0x61）	0x00 write（ESV＝0x61）	First lines of (1) to (8).
4	4	3	45	(3) Syntax long MidGetReceiveEpcMulti(short id_kind, short id, long eoj_code, short epc_code, short buff_size, short esv_code, short opc_code, const char* pdc_code, const char* epcedt_code, long *eoj_code2)	(3) Syntax long MidGetReceiveEpcMulti(short id_kind, short id, long eoj_code, short epc_code, short buff_size, short opc_code, const char* pdc_code, const char* epcedt_code, long *eoj_code2)	"short epc_code" was deleted from the syntax shown under (3) "Syntax" in "4.3.45 MidGetReceiveEpcMulti."

Part	Chapter	Section	Paragra	Before correction	After correction	Remarks
4	4	3	45	(4) Explanation Reads received data about a request for writing data into ECHONET properties of the object specified by id and eoj_code. This function can be called whenever the received data is to be read. id_kind : [in] ID type APIVAL_NODE_KIND : 0 (Device ID) APIVAL_EA_KIND : 1 (ECHONET address) Id : [in] Device ID or ECHONET address seoj_code : [in] SEOJ code (Only 3 low-order bytes are used.) When SEOJ does not exist, set to -1. deoj_code : [in] DEOJ code (Only 3 low-order bytes are used.) When WEOJ does not exist, set to -1. buff_size : [in] Area size esv_code : [in] ESV code ESV_SetI : 0x60 (Request for writing a property value not requiring a response) ESV_SetC : 0x61 (Request for writing a property value requiring a response) ESV_Get : 0x62 (Request for reading a property value) ESV_Inf_Req : 0x63 (Request for notifying a property value) ESV_INF : 0x73 (Notice of a property value) opc_code : [in] Set the EPC element count. pdc_code : [in] Pointer to the beginning of the array into which EPC codes and EDT code size information are to be entered. The number of elements is specified by the opc_code value. epcedt_code : [in] Pointer to the beginning of the array into which an EPC code and EDT code are to be entered.	(4) Explanation Reads received data on ECHONET properties of the object specified by id and eoj_code. This function can be called whenever the received data is to be read. id_kind : [in] ID type APIVAL_NODE_KIND : 0 (Device ID) APIVAL_EA_KIND : 1 (ECHONET address) id : [in] Device ID or ECHONET address eoj_code : [in] EOJ code (Only 3 low-order bytes are used.) When EOJ does not exist, set to -1. buff_size : [in] Area size esv_code : [in] ESV code opc_code : [in] OPC code pdc_code : [out] Pointer to the beginning of the array into which EPC codes and EDT code size information are to be entered. The number of elements is specified by the opc_code value. epcedt_code : [out] Pointer to the beginning of the array into which an EPC code and EDT code are to be entered. eoj_code2: [out] SEOJ or DEOJ code for communication (In the case where eoj_code2 exists, it will be the communication DEOJ code when the eoj_code specifies another node's EOJ, and the communication SEOJ code when the eoj_code specifies an own NodeID EOJ.)	The text of "4.3.45 MidGetReceiveEpcMulti" was corrected.
7	3	6	8	SD(3)LF 0x02	SD(3)LF 0x01	The listing under the heading "Processing result" in (9) "Lower-layer communication software profile acquisition service."
7	3	6	8	SD(5)LF 0x02	SD(5)LF 0x01	The listing under the heading "Processing result" in (9) "Lower-layer communication software profile acquisition service."
7	3	6	8	SD(9)LF 0x02	SD(9)LF 0x01	The listing under the heading "Processing result" in (9) "Lower-layer communication software profile acquisition service."
7	3	6	8	SHD：0x9D	SHD：0x9C	The listing under the heading "Processing result" in (19) "Stop notice service."