

S10mini.

HITACHI

S10mini
ハードウェアマニュアル

オプション
FL.NET

本製品を輸出される場合には、外国為替及び外国貿易法の規制並びに米国輸出管理規則など外国の輸出関連法規をご確認の上、必要な手続きをお取りください。
なお、不明な場合は、弊社担当営業にお問合わせください。

2000年 9月 (第1版) SMJ-1-101 (A) (廃版)

2003年 7月 (第2版) SMJ-1-101 (B) (廃版)

2008年 3月 (第3版) SMJ-1-101 (C)

- このマニュアルの一部、または全部を無断で転写したり複製することは、固くお断りいたします。
- このマニュアルの内容を、改良のため予告なしに変更することがあります。

このマニュアルは、以下のハードウェアの説明をしたものです。

<ハードウェア>

FL.NET (LQE000)

変更内容 (SMJ-1-101(C))

追加・変更内容	ページ
7. 1. 1 モジュールの交換、増設を追加	145

上記追加変更の他に、記述不明瞭な部分、単なる誤字・脱字などについては、お断りなく訂正しました。

はじめに

このたびは、S10mini用オプション FL.NETモジュールをご利用いただきましてありがとうございます。

この「ハードウェアマニュアル オプション FL.NET」は、FL.NETモジュールの取り扱いについて述べたものです。このマニュアルをお読みいただき正しくご使用いただくようお願いいたします。

S10miniの製品には、標準仕様品と耐環境仕様品があります。耐環境仕様品は、標準仕様品と比べ部品のメッキ厚、コーティングが強化されています。

耐環境仕様品型式は、標準仕様品型式の後に“-Z”が付いています。

(例) 標準仕様品型式：LQE000

耐環境仕様品型式：LQE000-Z

このマニュアルは、標準仕様品と耐環境仕様品とで共通の内容となっています。このマニュアルには、標準仕様品のモジュール型式のみを記載していますが、耐環境仕様品をご使用の場合も、このマニュアルに従って、正しくご使用いただくようお願いいたします。

<商標について>

Microsoft® Windows® operating system, Microsoft® Windows® 95 operating system, Microsoft® Windows® 98 operating system, Microsoft® Windows® 2000 operating system, Microsoft® Windows® XP operating systemは、米国Microsoft Corporationの米国およびその他の国における登録商標です。

<記憶容量の計算値についての注意>

- 2^n 計算値の場合（メモリ容量・所要量、ファイル容量・所要量など）

1KB（キロバイト）＝1,024バイトの計算値です。

1MB（メガバイト）＝1,048,576バイトの計算値です。

1GB（ギガバイト）＝1,073,741,824バイトの計算値です。

- 10^n 計算値の場合（ディスク容量など）

1KB（キロバイト）＝1,000バイトの計算値です。

1MB（メガバイト）＝1,000²バイトの計算値です。

1GB（ギガバイト）＝1,000³バイトの計算値です。

目 次

1	概 要	1
1.1	FL-netとは	2
1.2	FL-netの特長	3
2	一般的なご注意	5
2.1	安全上のご注意	6
2.2	使用上の一般的なご注意	7
2.3	保証・サービス	9
3	FL.NETモジュール	11
3.1	システム構成	12
3.1.1	機能・性能仕様	13
3.1.2	サポートツール仕様	13
3.1.3	リンクデータ仕様	14
3.1.4	リンクパラメータの設定領域	16
3.1.5	プロファイルシステムパラメータの設定領域	17
3.2	FL.NETモジュールの各部名称と機能	18
3.2.1	外観・構造	18
3.2.2	各部名称と機能	19
4	FL.NETモジュールの実装	21
4.1	モジュールの実装	22
4.2	モジュールの実装方法	23
4.3	モジュールスイッチの設定方法	24
5	FL.NETモジュールの配線方法	25
5.1	通信ケーブルの接続	26
5.2	適用通信ケーブル	28
5.3	電源配線	29
5.4	アース配線	30
6	利用の手引き	31
6.1	イーサネットについて	32
6.1.1	10BASE-5システム	32

6.1.2	10BASE-Tシステム	36
6.2	FL-netについて	39
6.2.1	FL-netの概要	39
6.2.2	接続台数とノード番号	41
6.2.3	データ通信の種類	42
6.2.4	伝送データ量	43
6.2.5	転送周期とその監視	44
6.2.6	データ領域とメモリ	45
6.2.7	通信管理テーブル	46
6.2.8	サイクリック伝送と領域	48
6.2.9	メッセージ伝送	53
6.3	FL.NETの設定方法	65
6.3.1	立ち上げ手順	65
6.3.2	モジュールの実装とスイッチの設定	67
6.3.3	ツールの接続方法	67
6.3.4	ツールの立ち上げ方法	67
6.4	FL.NETの使い方	68
6.4.1	リンクパラメータの設定方法	68
6.4.2	CPUメモリの割り付け方法	71
6.4.3	ビットデータの使い方	80
6.4.4	ワードデータの使い方	81
6.4.5	メッセージ通信の使い方	81
6.4.6	管理テーブルの使い方	132
6.4.7	FL.NETモジュールの通信時間	139
6.4.8	通信ログの使い方	141
7	保守・点検	143
7.1	保守点検項目	144
7.1.1	モジュールの交換、増設	145
8	トラブルシューティング	147
8.1	故障かなと思ったら	148
8.2	ネットワークの不具合とその対策	149
8.3	FL-netを使用するうえでの注意事項	152
8.4	エラー表示と対策	153

9	付 録	163
9.1	システム構築ガイド	164
9.1.1	イーサネットの概要	164
9.1.2	10BASE-5の仕様	165
9.1.3	10BASE-Tの仕様	166
9.1.4	その他イーサネットの仕様	167
9.2	システム構成例	169
9.2.1	小規模構成	169
9.2.2	基本構成	170
9.2.3	大規模構成	171
9.2.4	長距離分散構成	172
9.2.5	局所集中構成	173
9.2.6	局所長距離分散構成	174
9.2.7	FL-netのシステムの考え方	175
9.2.8	汎用のイーサネットとFL-netの相違点	175
9.3	ネットワークシステムの定義	176
9.3.1	通信プロトコルの規格	176
9.3.2	通信プロトコルの階層構造	176
9.3.3	FL-netの物理層について	177
9.3.4	IPアドレス	177
9.3.5	サブネットマスク	178
9.3.6	TCP/IP, UDP/IP通信プロトコル	178
9.3.7	ポート番号	178
9.3.8	FL-netのデータフォーマット	179
9.3.9	FL-netのトランザクションコード	181
9.4	FL-netのネットワーク管理	183
9.4.1	FL-netのトークン管理	183
9.4.2	FL-netへの加入/FL-netからの離脱	186
9.4.3	ノードの状態管理	188
9.4.4	FL-netの自ノード管理テーブル	188
9.4.5	FL-netの参加ノード管理テーブル	189
9.4.6	FL-netの状態管理	190
9.4.7	FL-netのメッセージ通番管理	190
9.5	ネットワーク構成部品	191
9.5.1	イーサネットの構成部品一覧	191
9.5.2	10BASE-5関連	193

9.5.3	10BASE-T関連	203
9.6	FL-netのネットワーク施工方法	205
9.6.1	10BASE-5同軸ケーブルの配線	205
9.6.2	10BASE-T (UTP) ケーブル	221
9.7	FL-netシステムの接地	223
9.7.1	FL-netシステムの接地の概要	223
9.7.2	電源配線と接地	224
9.7.3	FL-netシステムのネットワーク機器の電源配線とアース接地	225
9.7.4	配線ダクト・電線管の配線と接地	226
9.8	FL-net工事施工チェックシート	227
9.9	FL-netのプロファイル	228
9.9.1	機器通信情報の分類	228
9.9.2	共通パラメータの詳細	229
9.9.3	デバイス固有パラメータの詳細 (使用する場合)	229
9.9.4	システムパラメータの例 (PLCの例)	230
9.9.5	システムパラメータの例 (CNCの例)	235

目 次

図 1-1	FAコントロールネットワーク構成例	2
図 1-2	FL-netプロトコルの基本構造	3
図 3-1	FAコントロールネットワーク構成例	12
図 3-2	FL.NETモジュールの外観	18
図 3-3	FL.NETモジュールの各部名称	19
図 4-1	モジュールの実装（参考例）	22
図 4-2	モジュールの実装方法（参考例）	23
図 4-3	モジュールスイッチの設定方法	24
図 5-1	モジュールの10BASE-5トランシーバケーブル接続	26
図 5-2	モジュールの10BASE-Tケーブル接続	27
図 5-3	モジュールの電源配線	29
図 5-4	ユニットのアース配線	30
図 6-1	10BASE-5システムの基本接続方法（リピータなし、最大距離500m）	32
図 6-2	10BASE-5システムの基本接続方法（リピータ使用、最大距離1,500m）	33
図 6-3	10BASE-5システムの基本接続方法（リピータ使用、最大距離2,500m）	34
図 6-4	10BASE-Tシステムの基本接続方法1	36
図 6-5	10BASE-Tシステムの基本接続方法2	36
図 6-6	イーサネットのIPアドレスのクラス区分	38
図 6-7	イーサネットのクラスCのIPアドレス例	38
図 6-8	FL-netのコンセプト	39
図 6-9	FAリンクプロトコル	40
図 6-10	FL-netのIPアドレス	41
図 6-11	FL-netの接続台数とノード番号	41
図 6-12	FL-netのデータ通信の種類	42
図 6-13	コモンメモリとサイクリック伝送の例	42
図 6-14	メッセージ伝送の例	43
図 6-15	サイクリック伝送のデータ量	43
図 6-16	メッセージ伝送のデータ量	44
図 6-17	リフレッシュサイクル許容時間の例	44
図 6-18	データ領域とメモリ	45
図 6-19	トークン巡回とサイクリック伝送1	49
図 6-20	トークン巡回とサイクリック伝送2	49
図 6-21	サイクリック伝送のコモンメモリ領域例1	50
図 6-22	サイクリック伝送のコモンメモリ領域例2	51

図6-23	コモンメモリ領域1と領域2	51
図6-24	データの同時性保証	52
図6-25	メッセージ伝送概要	53
図6-26	バイトブロック読み出し	55
図6-27	バイトブロック書き込み	56
図6-28	ワードブロック読み出し	57
図6-29	ワードブロック書き込み	58
図6-30	ネットワークパラメータ読み出し	59
図6-31	ネットワークパラメータ書き込み	60
図6-32	運転・停止指令	61
図6-33	プロファイル読み出し	62
図6-34	通信ログデータの読み出し	63
図6-35	通信ログデータのクリア	63
図6-36	メッセージ折り返し機能	64
図6-37	透過形メッセージ伝送	64
図6-38	IPアドレス設定画面	66
図6-39	リンクパラメータ設定画面	68
図6-40	FLNETモジュールのコモンメモリデータ転写イメージ	71
図6-41	自ノードのCPUメモリ割り付け画面	73
図6-42	参加他ノード用コモンメモリ領域設定画面	75
図6-43	自ノードの各状態フラグエリア設定画面	77
図6-44	参加他ノードの各状態フラグエリア設定画面	79
図6-45	領域1データのビット変換	80
図6-46	透過型受信フラグエリア、透過型受信起動タスク設定画面	87
図6-47	参加ノード番号一覧画面1	132
図6-48	自ノード情報（自ノード管理テーブル）表示画面	133
図6-49	参加ノード番号一覧画面2	137
図6-50	参加他ノードリンク情報表示画面	138
図6-51	ネットワーク状態（ネットワーク管理テーブル）表示画面	138
図6-52	RAS情報ダイアログボックス	141
図8-1	エラーフリーズ情報テーブル内スタックフレームの詳細	157
図8-2	モジュールのエラーランプ	161
図9-1	IEEE802.3標準化動向	164
図9-2	10BASE-5イーサネット構成例	165
図9-3	10BASE-Tイーサネット構成例	166
図9-4	10BASE-2イーサネット構成例	167

図9-5	光イーサネット構成例	167
図9-6	無線イーサネット構成例	168
図9-7	小規模構成例	169
図9-8	基本構成例	170
図9-9	大規模構成例	171
図9-10	長距離分散構成例	172
図9-11	局所集中構成例	173
図9-12	局所長距離分散構成例	174
図9-13	FAリンクプロトコルの位置付け	176
図9-14	FL-netのIPアドレス	177
図9-15	FL-netのデータフォーマット概要	179
図9-16	FL-netのデータ（1フレーム）例	180
図9-17	FL-netヘッダ	181
図9-18	トークンの流れ	184
図9-19	立ち上げ時のタイムチャート1	186
図9-20	立ち上げ時のタイムチャート2	187
図9-21	イーサネットの構成部品一覧	191
図9-22	トランシーバ概観	193
図9-23	トランシーバとトランシーバケーブル（AUI）	193
図9-24	タップと同軸ケーブルの取り付け	194
図9-25	タップとトランシーバ本体の取り付け	194
図9-26	イーサネットのトランシーバ（タップ形）	195
図9-27	イーサネットのトランシーバ（コネクタ形）	196
図9-28	イーサネットのマルチポートトランシーバ	197
図9-29	イーサネットのリピータ	198
図9-30	イーサネットの同軸ケーブル	198
図9-31	イーサネットの同軸コネクタ	199
図9-32	イーサネットの中継コネクタ	199
図9-33	イーサネットのターミネータ（終端抵抗）	199
図9-34	イーサネットの同軸ケーブルアース端子	200
図9-35	イーサネットのトランシーバケーブル	200
図9-36	イーサネットの10BASE-5/T変換器	201
図9-37	イーサネットの10BASE-5/T変換器取り付け	201
図9-38	イーサネットの同軸／光変換メディアコンバータ・リピータ	202
図9-39	イーサネットのハブ	203
図9-40	イーサネットの10BASE-Tケーブル	204

図9-41	イーサネットの10BASE-T/光変換メディアコンバータ・リピータ	204
図9-42	同軸ケーブル被覆（PVCシース）剥ぎ	206
図9-43	同軸ケーブルアルミテープ除去1	206
図9-44	同軸ケーブルアルミテープ除去2	206
図9-45	同軸ケーブル絶縁体剥ぎ	207
図9-46	同軸ケーブルの部品組み込みおよびシールド処理	207
図9-47	同軸ケーブルのシールド処理およびピンコンタクトはんだ付け	207
図9-48	同軸ケーブルのコネクタ組み立て	207
図9-49	トランシーバ各部の名称	209
図9-50	トランシーバのシールド圧接ピンをタップ本体挿入	210
図9-51	トランシーバのタップフレームの押えビス緩め	210
図9-52	トランシーバのタップフレームとタップ本体装着	210
図9-53	トランシーバのタップフレームと同軸ケーブルの挿入	211
図9-54	同軸ケーブルの心線プローブ用の穴あけ	211
図9-55	同軸ケーブルへの心線プローブ取り付け	212
図9-56	タップコネクタへのトランシーバ本体挿入	212
図9-57	トランシーバ本体とタップの固定	213
図9-58	コネクタ形トランシーバ取り付け	214
図9-59	リピータ接続	214
図9-60	リピータの設置スペース	215
図9-61	中継コネクタの絶縁	215
図9-62	L型コネクタの絶縁	216
図9-63	トランシーバおよびトランシーバケーブルの壁面設置例1	216
図9-64	トランシーバおよびトランシーバケーブルの壁面設置例2	217
図9-65	トランシーバおよびトランシーバケーブルの天井内、床下内設置例	218
図9-66	アース端子本体への挿入つめ取り付け	219
図9-67	アース端子取り付け	219
図9-68	挿入つめの余分を切断	220
図9-69	同軸ケーブルのアース端子取り付け	220
図9-70	UTPケーブルのシース剥ぎ	221
図9-71	UTPケーブルの信号線の切断	221
図9-72	UTPケーブルのコネクタへの挿入	222
図9-73	UTPケーブルコネクタの組み立て	222
図9-74	コントローラ制御盤の接地方法例1（鉄骨接地の場合）	223
図9-75	コントローラ制御盤の接地方法例2（コントローラ単独D種接地の場合）	223
図9-76	FL-netシステムの電源配線と接地の例	224

図9-77 FL-netシステムのネットワーク機器の電源配線とアース接地例	225
図9-78 配線ダクトを使用した場合の配線例	226
図9-79 電線管を使用した場合の配線例	226
図9-80 機器通信情報の分類	228

表 目 次

表 3-1	一般仕様	12
表 3-2	機能・性能仕様	13
表 3-3	コモンメモリ領域をCPUメモリに割り付け可能なレジスタ	14
表 3-4	仮想アドレス空間と物理メモリ	14
表 3-5	仮想アドレス空間と各シンボル名称対応	15
表 3-6	リンクデータのエラーメモリ領域	16
表 3-7	プロファイルシステムパラメータの設定領域（記載例）	17
表 3-8	FL.NETモジュールの各部名称とその機能	19
表 4-1	モジュールNo.設定スイッチ詳細	24
表 5-1	モジュール適用通信ケーブル（10BASE-5トランシーバケーブル）	28
表 5-2	モジュール適用通信ケーブル（10BASE-Tツイストペアケーブル）	28
表 6-1	イーサネットのシステム構成上の一般仕様	35
表 6-2	自ノード管理テーブル	46
表 6-3	参加ノード管理テーブル	47
表 6-4	ネットワーク管理テーブル	47
表 6-5	サポートメッセージ伝送一覧	54
表 6-6	ネットワークパラメータ情報	59
表 6-7	コモンメモリ領域をCPUメモリに割り付けできるアドレス範囲	72
表 6-8	Cモードハンドラー一覧	83
表 6-9	演算ファンクション一覧	84
表 6-10	透過型受信フラグ割り付け	88
表 8-1	ネットワークに関する不具合と対策（通信ができない場合）	149
表 8-2	ネットワークに関する不具合と対策（通信が不安定な場合）	150
表 8-3	CPUモジュールインディケータ表示	153
表 8-4	FAプロトコルエラー時のCPUモジュールインディケータ表示詳細	154
表 8-5	ハードウェアエラー時のCPUモジュールインディケータ表示詳細	155
表 8-6	エラーフリーズ情報テーブル詳細	156
表 8-7	検出コード一覧	158
表 9-1	FL-netの通信プロトコル	176
表 9-2	IPアドレスのクラス	177
表 9-3	FL-netのポート番号	178
表 9-4	トランザクションコード一覧	182
表 9-5	トークンとデータ	185
表 9-6	ノードの状態管理のテーブル概要	188

表 9-7	自ノード管理テーブル	188
表 9-8	参加ノード管理テーブル	189
表 9-9	ネットワーク管理テーブル	190
表 9-10	メッセージ通番管理の送信用管理データ	190
表 9-11	メッセージ通番管理の受信用管理データ	190
表 9-12	イーサネットの構成部品一覧	192
表 9-13	同軸ケーブルの諸元	205
表 9-14	SQEスイッチの設定	213
表 9-15	FL-net工事施工チェックシート	227
表 9-16	共通パラメータの詳細	229
表 9-17	デバイス固有パラメータ詳細	229
表 9-18	システムパラメータの表形式文書表記 (PLCの例)	230
表 9-19	システムパラメータの表形式文書表記 (CNCの例)	235

1 概 要

1. 1 FL-netとは

FL-netは通産省の外郭団体である（財）製造科学技術センターにおいて、FAオープン推進協議会（JOP）が標準化を行ったオープンFAネットワークです。

図1-1に示すように、多数の異なるメーカーのプログラマブルコントローラ（PLC）や数値制御装置（CNC）などの各種FAコントローラやパーソナルコンピュータを相互接続し、制御・監視を実現できます。

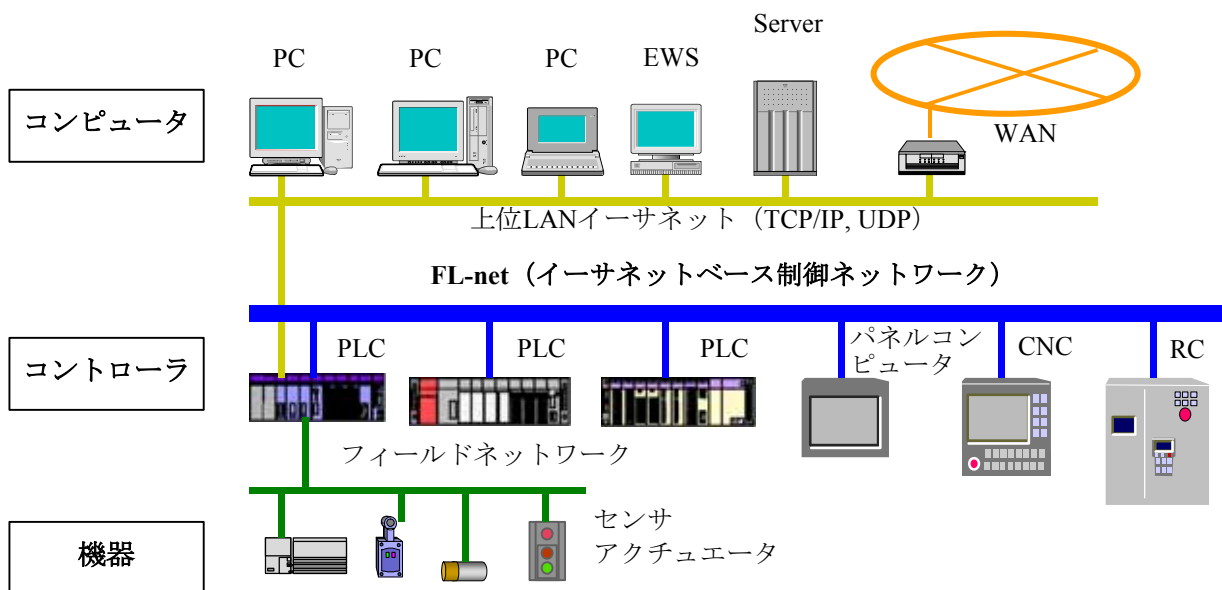


図1-1 FAコントロールネットワーク構成例

1. 2 FL-netの特長

FL-netには次のような特長があります。

- ・オープン化
- ・マルチベンダの実現
- ・多くの異なるメーカーのプログラマブルコントローラ（PLC）や数値制御装置（CNC）などのコントローラやパーソナルコンピュータなどを相互接続し、制御・監視を実現できます。

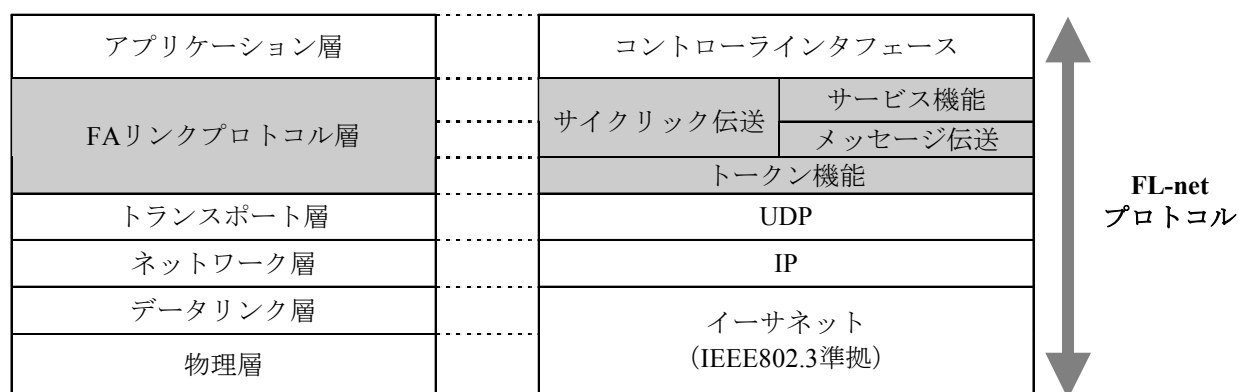


図1-2 FL-netプロトコルの基本構造

<広く普及した標準規格に準拠>

特にOAの機器で標準となっているイーサネットをベースに、標準のUDP/IPを使って効率のよい通信を実現しています。

- ・低価格
普及した通信デバイスにより構成できるため、低価格を実現しています。
- ・ネットワーク機器が広く普及
トランシーバやハブ、ケーブル、パーソナルコンピュータ用LANカードなどイーサネット用として広く普及したネットワーク用機器が使用できます。
- ・将来の高速化
将来10Mbps→100Mbps→1Gbpsと伝送速度の向上が期待できます。
- ・光通信化
また、これもイーサネット用に普及している光リピータなどを使用することにより、必要な部分を光ファイバ化することで、500m以上の長距離伝送や、耐ノイズ性の向上、屋外配線時の雷サージ対策を実施できます。

1 概 要

<FAコントローラ間に必要な通能をサポート>

ユーザの要求仕様がスタートとなっているため、FAに必要な各種の特長を持っています。

- ・大規模ネットワーク

最大254台の機器（ノード）が接続できます。

- ・用途に応じた2種類の通信機能

サイクリック通信により各ノードが同一のデータを常に共有できるコモンメモリ機能と、必要なときに必要な情報だけをやり取りするメッセージ通信機能の両方をサポートしています。

- ・大容量コモンメモリ

コモンメモリは8kビット+8kワードと大容量です。

- ・高速応答

FL-net対応モジュール同士の接続では、接続する機器により性能が異なりますが、S10mini FL.NETモジュール同士を接続した場合、64ms/32ノード（2kビット+2kワード時）の高速応答が実現できます。

- ・マスタレス方式による高い信頼性

マスタが存在しないことから、各ノードの加入／離脱が他のノードの通信に影響を与えず自由にできるため、どのノードも自由に電源の入／切やメンテナンスができます。

2 一般的なご注意

2 一般的なご注意

2.1 安全上のご注意

取り付け、運転、保守・点検の前に必ずこのマニュアルとその他の付属書類をすべて熟読し、正しくご使用ください。機器の知識、安全の情報そして注意事項のすべてについて熟読してご使用ください。また、このマニュアルは最終保守責任者のお手元に必ず届くようにしてください。


このマニュアルでは、安全注意事項のランクを「危険」「注意」として区分してあります。



：取り扱いを誤った場合に、危険な状況が起こりえて、死亡または重傷を受ける可能性が想定される場合。




：取り扱いを誤った場合に、危険な状況が起こりえて、中程度の傷害や軽傷を受ける可能性が想定される場合および物的損害だけの発生が想定される場合。

なお、に記載した事項でも、状況によっては重大な結果に結びつく可能性があります。

いずれも重要な内容を記載していますので必ず守ってください。

禁止、強制の絵表示の説明を次に示します。



：禁止（してはいけないこと）を示します。例えば分解禁止の場合はとなります。



：強制（必ずしなければならないこと）を示します。例えば接地の場合はとなります。

2.2 使用上の一般的なご注意

このFL.NETモジュールを使用する場合は、下記の点に特に注意してください。

(1) 取り付けについて

注 意

- カタログ、マニュアルに記載の環境で使用してください。高温、多湿、じんあい、腐食性ガス、振動、衝撃がある環境で使用すると、感電、火災、誤動作の原因になります。
- マニュアルに従って取り付けをしてください。取り付けに不備があると、落下、故障、誤動作の原因になります。
- 電線くずなどの異物が入らないようにしてください。火災、故障、誤動作の原因になります。

(2) 配線について

強 制

必ず接地（FG）を行ってください。接地しない場合、感電、誤動作の恐れがあります。

注 意

- 定格にあった電源を接続してください。定格と異なった電源を接続すると火災の原因になります。
- 配線は、資格のある専門家が行ってください。配線を誤ると火災、故障、感電の恐れがあります。

注 意

このモジュールに関する配線、設置については、必ずマニュアルの記載事項に従って行ってください。また、配線、設置に関する情報は必ず配線工事担当者に指示してください。異なる配線、設置によるノイズ耐量の低下で誤動作の原因になります。

2 一般的なご注意

(3) 使用上の注意

危険

- 通電中は端子に触れないでください。感電の恐れがあります。
- 非常停止回路、インタロック回路などはこの製品の外部で構成してください。この製品の故障により機械の破損や事故の恐れがあります。

注意

- 運転中のプログラム変更、強制出力、RUN、STOPなどの操作は十分に安全を確認してから行ってください。誤操作により、機械の破損や事故の恐れがあります。
- 電源は順序に従って投入してください。順序を誤ると誤動作により、機械の破損や事故の恐れがあります。
- 通電中は10BASE-5コネクタに触れないでください。静電気などによりシステムが誤動作することがあります。

(4) 保守について

禁止

分解、改造はしないでください。火災、故障、誤動作の原因になります。

注意

- モジュール／ユニットは電源をOFFしてから着脱してください。感電、誤動作、故障の原因になります。
- ヒューズは指定品と交換してください。火災、故障の原因になります。

2.3 保証・サービス

特別な保証契約がない場合、この製品の保証は次のとおりです。

1. 保証期間と保証範囲

【保証期間】

この製品の保証期間は、ご注文のご指定場所に納入後1年といたします。

【保証範囲】

上記保証期間中に、このマニュアルに従った製品仕様範囲内の正常な使用状態で故障が生じた場合は、その機器の故障部分をお買い上げの販売店または（株）日立エンジニアリング・アンド・サービスにお渡しください。交換または修理を無償で行います。ただし、郵送いただく場合は、郵送料金、梱包費用はご注文主のご負担になります。

次のいずれかに該当する場合は、この保証の対象範囲から除外いたします。

- 製品仕様範囲外の取り扱いおよび使用により故障した場合。
- 納入品以外の事由により故障した場合。
- 納入者以外の改造または修理により故障した場合。
- リレーなどの消耗部品の寿命により故障した場合。
- 上記以外の天災、災害など、納入者側の責任ではない事由により故障した場合。

ここでいう保証とは、納入した製品単体の保証を意味します。したがって、弊社ではこの製品の運用および故障を理由とする損失、逸失利益等の請求につきましては、いかなる責任も負いかねますのであらかじめご了承ください。また、この保証は日本国内でのみ有効であり、ご注文主に対して行うものです。

2. サービスの範囲

納入した製品の価格には技術者派遣などのサービス費用は含まれておりません。次に該当する場合は別個に費用を申し受けます。

- 取り付け調整指導および試運転立ち会い。
- 保守点検および調整。
- 技術指導、技術教育、およびトレーニングスクール。
- 保証期間後の調査および修理。
- 保証期間中においても、上記保証範囲外の事由による故障原因の調査。

<このページは余白です>

3 FL.NETモジュール

3 FL.NETモジュール

3.1 システム構成

FL.NETは、FL-netのプロトコルに準拠した当社のモジュールです。

図3-1にFL-net通信を使用した場合のシステム構成例を示します。

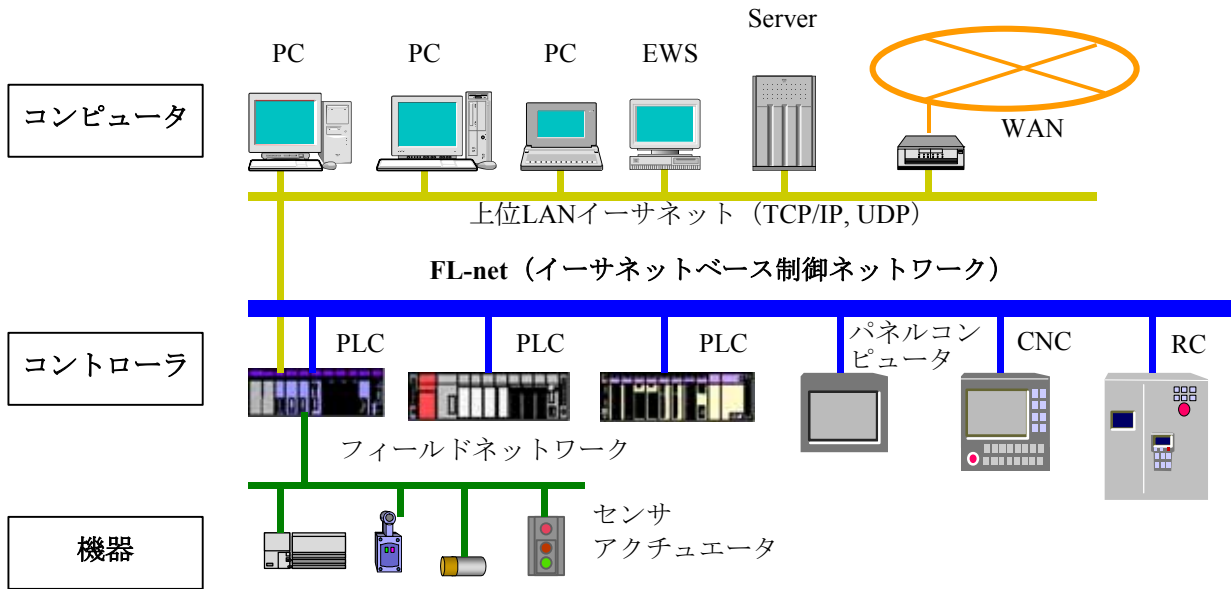


図3-1 FAコントロールネットワーク構成例

表3-1 一般仕様

項目	仕様
使用周囲温度	0~55℃
保存周囲温度	-20~75℃
使用周囲雰囲気	塵埃0.1mg/m ³ 以下 腐食性ガスなきこと
使用周囲湿度	30~90%RH (結露しないこと)
保存周囲湿度	10~90%RH (結露しないこと)
耐振動	JIS C 0040準拠
耐衝撃	JIS C 0041準拠

3. 1. 1 機能・性能仕様

表 3-2 機能・性能仕様

項目	仕様	備考
型式	LQE000	
最大実装ユニット数	2モジュール/CPU (左詰めで実装)	
伝送速度	10Mbps	
電氣的インタフェース	IEEE802.3準拠 (CSMA/CD準拠)	
伝送プロトコル	UDP/IP、FAリンクプロトコル	
接続台数	最大254台/ネットワーク	
接続コネクタ	AUIコネクタ (10BASE-5) RJ45コネクタ (10BASE-T)	
外部AUI供給電源 (端子台)	DC12V、500mA以下 (M3ねじ端子台)	10BASE-5使用時のみ必要
モジュール寸法	34 (W) × 130 (H) × 100.2 (D) mm	
質量	240g	
転送語数	サイクリック 最大8.5kワード (ノード) メッセージ 最大1024バイト (ノード)	
転送距離	10BASE-5時：最長2.5km (ノード間) 10BASE-T時：最長1.5km (ノード間)	
ケーブル長	トランシーバケーブル：最長50m (10BASE-5) ツイストペアケーブル：最長100m (10BASE-T)	

3. 1. 2 サポートツール仕様

このFL.NETモジュールを使用するには、設定ツール [FL.NET For Windows®] を使用してノード番号、コモンメモリ設定などの各種設定を行う必要があります。

設定ツールに関する詳しい仕様は、「ソフトウェアマニュアル オプション FL.NET For Windows® (マニュアル番号 SAJ-3-139)」を参照してください。

3 FL.NETモジュール

3.1.3 リンクデータ仕様

(1) コモンメモリ領域

このFL.NETモジュールでは、コモンメモリ領域をノード単位で設定できます。

コモンメモリ領域の設定方法に関しては、「6.4.1 リンクパラメータの設定方法」を参照してください。また、コモンメモリ領域をCPUメモリへ割り付けするには、「6.4.2 CPUメモリの割り付け方法」を参照してください。

表3-3 コモンメモリ領域をCPUメモリに割り付け可能なレジスタ

No.	指定可能レジスタ/アドレス	備 考
1	XW000~XWFF0	外部入力
2	YW000~YWFF0	外部出力
3	JW000~JWFF0	トランスファレジスタ
4	QW000~QWFF0	レシーブレジスタ
5	GW000~GWFF0	グローバルリンクレジスタ
6	RW000~RWFF0	内部レジスタ
7	EW400~EWFF0	イベントレジスタ
8	MW000~MWFF0	内部レジスタ
9	DW000~DWFFF	ファンクションデータレジスタ
10	FW000~FWBFF	ファンクションワークレジスタ
11	/100000~/4FFFFE (拡張メモリ)	CPUモジュールに拡張メモリが実装されている必要があります。

(2) 仮想アドレス空間と物理メモリ

表3-4 仮想アドレス空間と物理メモリ

項 目	説 明	
領域名	S10miniメモリアドレス (0x000000~0xFFFFF)	
アクセス単位	WORD	
領域の大きさ	16777216バイト	
アクセス属性	読み出し/書き込み 条件：一部ROMエリアおよびOSエリアにつき、書き込み不可	
仮想アドレスとの対応 (ワードブロック) (*)	ベンダ固有表記	仮想アドレス
	0x000000, 0x000001	0x00000000
	0x000002, 0x000003	0x00000001
	} }	}
	0xFFFFFE, 0xFFFFF	0x007FFFFF
データの並び	2バイトエリアがワードブロック1ワードに対応 <div style="display: flex; justify-content: space-around; align-items: center;"> <div style="text-align: center;"> <small>MSB</small> <div style="border: 1px solid black; padding: 2px;">アドレス n</div> </div> <div style="text-align: center;"> <small>LSB</small> <div style="border: 1px solid black; padding: 2px;">アドレス n+1</div> </div> </div>	
その他のアクセス方法	特になし	

「0x000012AB」は、16進数 (000012AB hex) を示します。

(*) 仮想アドレスとの対応にて、バイトブロックは未サポートです。

表3-5に、仮想アドレス空間とS10miniメモリアドレス、各レジスタの関係を示します。

表 3-5 仮想アドレス空間と各シンボル名称対応

レジスタ名称	シンボル名称	S10mini メモリアドレス	仮想アドレス空間
外部入力	X000~XFFF	0x0A0000~0x0A1FFE	0x00050000~0x00050FFF
外部出力	Y000~YFFF	0x0A4000~0x0A5FFE	0x00052000~0x00052FFF
内部レジスタ	R000~RFFF	0x0AC000~0x0ADFFE	0x00056000~0x00056FFF
グローバルレジスタ	G000~GFFF	0x0A8000~0x0A9FFE	0x00054000~0x00054FFF
オンディレイタイマ	T000~T1FF	0x0B3000~0x0B33FE	0x00059800~0x000599FF
ワンショットタイマ	U000~U0FF	0x0B5000~0x0B51FE	0x0005A800~0x0005A8FF
カウンタ	C000~C0FF	0x0B7000~0x0B71FE	0x0005B800~0x0005B8FF
キープ	K000~KFFF	0x0B0000~0x0B1FFE	0x00058000~0x00058FFF
システムレジスタ	S000~SBFF	0x0BE800~0x0BFFFE	0x0005F400~0x0005FFFF
ゼットレジスタ	Z000~Z3FF	0x0BE000~0x0BE7FE	0x0005F000~0x0005F3FF
内部レジスタ	M000~MFFF	0x0AE000~0x0AFFFE	0x00057000~0x00057FFF
トランスファレジスタ	J000~JFFF	0x0A2000~0x0A3FFE	0x00051000~0x00051FFF
レシーブレジスタ	Q000~QFFF	0x0A6000~0x0A7FFE	0x00053000~0x00053FFF
外部入力 (ワード)	XW000~XWFF0	0x0E0000~0x0E01FE	0x00070000~0x000700FF
外部出力 (ワード)	YW000~YWFF0	0x0E0400~0x0E05FE	0x00070200~0x000702FF
内部レジスタ (ワード)	RW000~RWFF0	0x0E0C00~0x0E0DFE	0x00070600~0x000706FF
グローバルレジスタ (ワード)	GW000~GWFF0	0x0E0800~0x0E09FE	0x00070400~0x000704FF
オンディレイタイマ (ワード)	TW000~TW1F0	0x0E1300~0x0E133E	0x00070980~0x0007099F
ワンショットタイマ (ワード)	UW000~UW0F0	0x0E1500~0x0E151E	0x00070A80~0x00070A8F
カウンタ (ワード)	CW000~CW0F0	0x0E1700~0x0E171E	0x00070B80~0x00070B8F
キープ (ワード)	KW000~KWFF0	0x0E1000~0x0E11FE	0x00070800~0x000708FF
システムレジスタ (ワード)	SW000~SWBF0	0x0E1E80~0x0E1FFE	0x00070F40~0x00070FFF
ゼットレジスタ (ワード)	ZW000~ZW3F0	0x0E1E00~0x0E1E7E	0x00070F00~0x00070F3F
内部レジスタ (ワード)	MW000~MWFF0	0x0E0E00~0x0E0FFE	0x00070700~0x000707FF
トランスファレジスタ (ワード)	JW000~JWFF0	0x0E0200~0x0E03FE	0x00070100~0x000701FF
レシーブレジスタ (ワード)	QW000~QWFF0	0x0E0600~0x0E07FE	0x00070300~0x000703FF
ファンクションデータレジスタ	DW000~DWFFF	0x061000~0x062FFE	0x00030800~0x000317FF
ファンクションワークレジスタ	FW000~FWBFF	0x0E2000~0x0E37FE	0x00071000~0x00071BFF
拡張メモリアドレス	拡張メモリアドレス	0x100000~0x4FFFFFFE	0x00080000~0x0027FFFF

3 FL.NETモジュール

(3) エラーメモリ領域

このFL.NETモジュールでは、エラー情報を記録するためのテーブルを用意しています。

表3-6 リンクデータのエラーメモリ領域

項 目	仕 様
エラーフリーズ情報 テーブル	モジュールにて異常動作が発生した 際の情報を格納します。
エラーメッセージデータ テーブル	他ノードからの異常応答メッセージ を格納します。

各テーブルに関する詳細情報は、「8 トラブルシューティング」を参照してください。

(4) ステータスメモリ領域

FL.NETモジュールのステータスメモリ領域は、設定ツール [FL.NET For Windows®] にて参照できます (詳細は、「ソフトウェアマニュアル オプション FL.NET For Windows® (マニュアル番号 SAJ-3-139)」を参照してください)。

3. 1. 4 リンクパラメータの設定領域

FL.NETモジュールのリンクパラメータは、設定ツール [FL.NET For Windows®] にて参照できます (詳細は、「ソフトウェアマニュアル オプション FL.NET For Windows® (マニュアル番号 SAJ-3-139)」を参照してください)。

3. 1. 5 プロファイルシステムパラメータの設定領域

FL.NETモジュールのプロファイルシステムパラメータは、設定ツール [FL.NET For Windows®] にて参照できます（詳細は、「ソフトウェアマニュアル オプション FL.NET For Windows®（マニュアル番号 SAJ-3-139）」を参照してください）。

FL.NETモジュールでは、プロファイルシステムパラメータ内のベンダ名、メーカー型式、プロトコルのバージョンは固定データです。また、FAリンクの状態／自ノードの状態は、FL.NETモジュールが通信状態などに応じて自動的に変更します。ユーザは設定ツールを使用して、ノード名のみ設定できます。

表 3-7 プロファイルシステムパラメータの設定領域（記載例）

項 目	バイト長	データ	内 容
ベンダ名	10バイト	“HITACHI”	ベンダの名称
メーカー型式	10バイト	“S10mini”	メーカーの型式、デバイスの名称
ノード名（設備名）	10バイト	—————	ユーザ設定によるノードの名称
プロトコルのバージョン	1バイト	—————	0x80固定
FAリンクの状態	1バイト	—————	加入／離脱など
自ノードの状態	1バイト	—————	ノード番号重複検知など

3 FL.NETモジュール

3.2 FL.NETモジュールの各部名称と機能

3.2.1 外観・構造

図3-2にFL.NETモジュールの外観を示します。

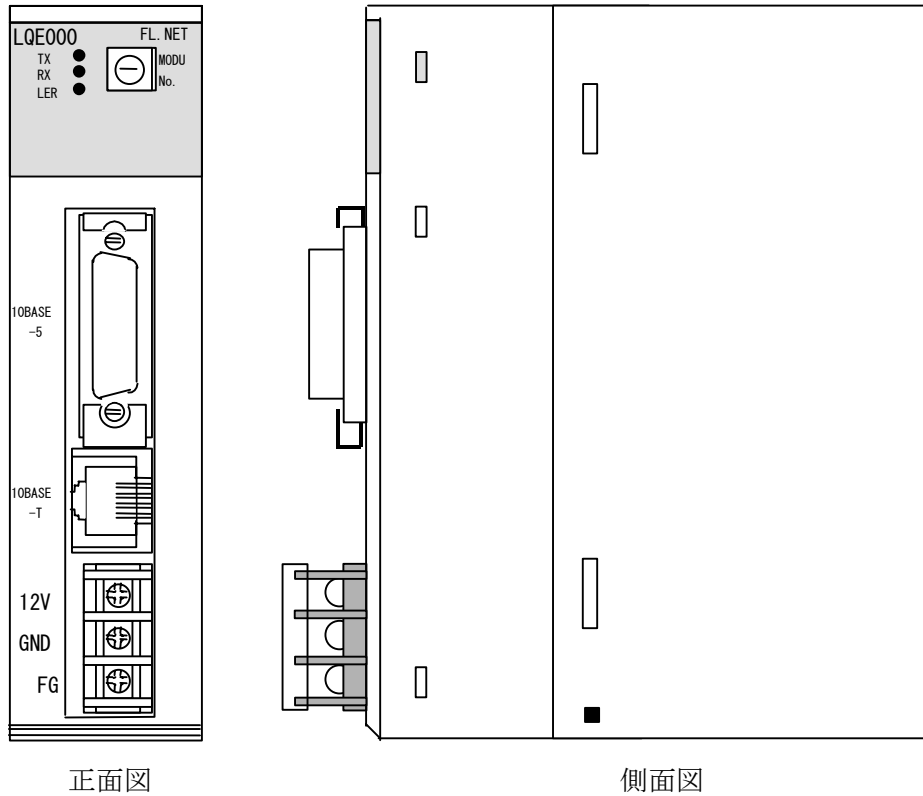


図3-2 FL.NETモジュールの外観

3. 2. 2 各部名称と機能

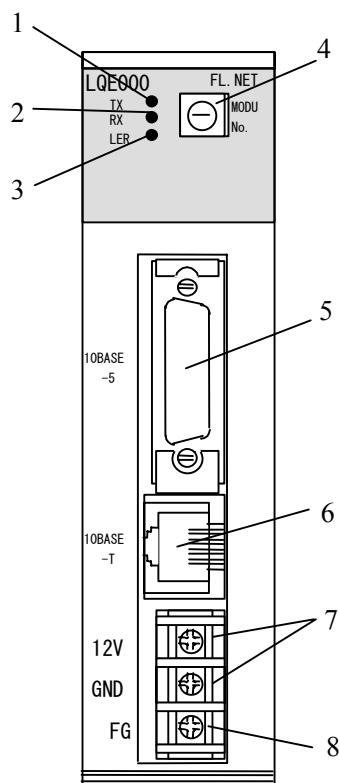


図 3-3 FL.NETモジュールの各部名称

表 3-8 FL.NETモジュールの各部名称とその機能

No.	名 称	機 能
1	TX LED	データ転送時に点灯します。
2	RX LED	伝送路上にデータが流れているとき（キャリア検出時）、点灯します。
3	LER LED	ハードウェア異常、FAリンク未接続時、FAリンク通信異常発生時に点灯します。
4	モジュールNo. 設定スイッチ	メインモジュールとサブモジュールの設定および使用する通信ポート種類を設定します。
5	10BASE-5 I/F コネクタ	S10miniまたは他のコントローラと接続します。
6	10BASE-T I/F コネクタ	S10miniまたは他のコントローラと接続します。
7	電源入力端子	10BASE-5 I/Fコネクタに接続するトランシーバの電源（DC12V）を接続します。
8	フレームグラウンド	トランシーバケーブルのシールド線に接続します。

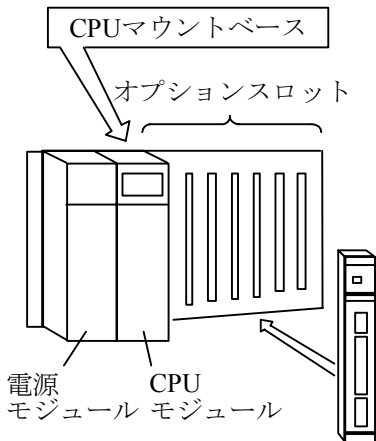
<このページは余白です>

4 FL.NETモジュールの実装

4 FL.NETモジュールの実装

4.1 モジュールの実装

FL.NETモジュールを、CPUマウントベースのオプションスロットに実装してください。



<機器構成>

CPU拡張マウントベース : HSC-1020, HSC-1040, HSC-1080

PSスロット : 電源モジュール

(LQV000, LQV020, LQV100) を実装。

CPUスロット : CPUモジュール

(LQP000, LQP010, LQP011, LQP120) を実装。

LQP000 : モデルS

LQP010 : モデルH

LQP011 : モデルF

LQP120 : モデルD

スロット0~7

: CPUオプションモジュールまたはI/Oを実装。

図4-1 モジュールの実装 (参考例)

- FL.NETモジュールは、CPUモジュールとの間にI/Oモジュールが入らないように左詰めで実装してください。
- FL.NETモジュールを1枚実装する場合は、必ずメインモジュールの設定にして使用してください。

4.2 モジュールの実装方法

CPUマウントベースにこのモジュールを実装し、ねじ止めを行ってください。

このモジュール実装に際しては、マウントベースに対して正面からまっすぐに実装してください（悪い例のように斜めに実装すると、コネクタが壊れる恐れがあります）。

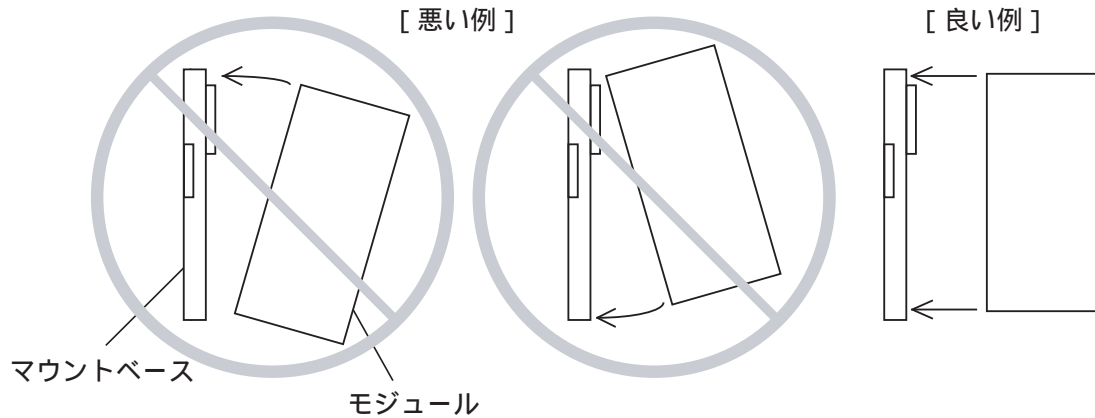


図4-2 モジュールの実装方法（参考例）

注意

筐体の構造上、マウントベースが頭上に配置されている場合、モジュールは脚立などを使用してまっすぐに取り付けてください。斜めに取り付けるとコネクタを破損する恐れがあります。

4 FL.NETモジュールの実装

4.3 モジュールスイッチの設定方法

FL.NETモジュールを使用する際、メインモジュール、サブモジュール、および通信ポートを設定してください。

モジュールNo.は、モジュールNo.設定スイッチの矢印を、指定したいモジュールNo.に合わせてください。

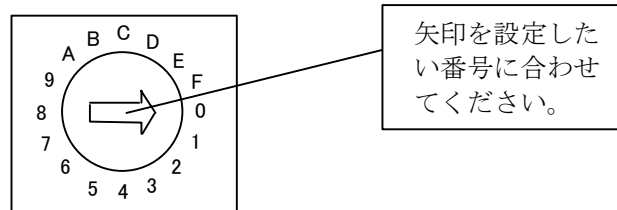


図4-3 モジュールスイッチの設定方法

表4-1にモジュールNo.設定スイッチについて示します。

表4-1 モジュールNo.設定スイッチ詳細

MODU No.		設 定
メイン モジュール	サブ モジュール	
0	1	10BASE-5の通信
2	3	10BASE-Tの通信
4	5	これらのモジュールNo.の設定は無効です。 モジュールスイッチ設定エラーが表示され、 通信などをしません。
6	7	
8	9	
A	B	
C	D	
E	F	

システムが誤動作する恐れがあります。モジュールNo.は、電源を切った状態で設定してください。

5 FL.NETモジュールの 配線方法

5. 1 通信ケーブルの接続

(1) 10BASE-5トランシーバケーブルの接続

モジュールNo.設定スイッチを0または1に設定した場合、このモジュールは10BASE-5 I/Fコネクタを使用して、他モジュールと通信します。

その際、10BASE-5ケーブルを図5-1に示す位置に差し込んでください。

また、10BASE-5により通信するには、DC12V外部供給電源による電源供給が必要となります。外部電源供給の接続については、「5. 3 電源配線」を参照してください。

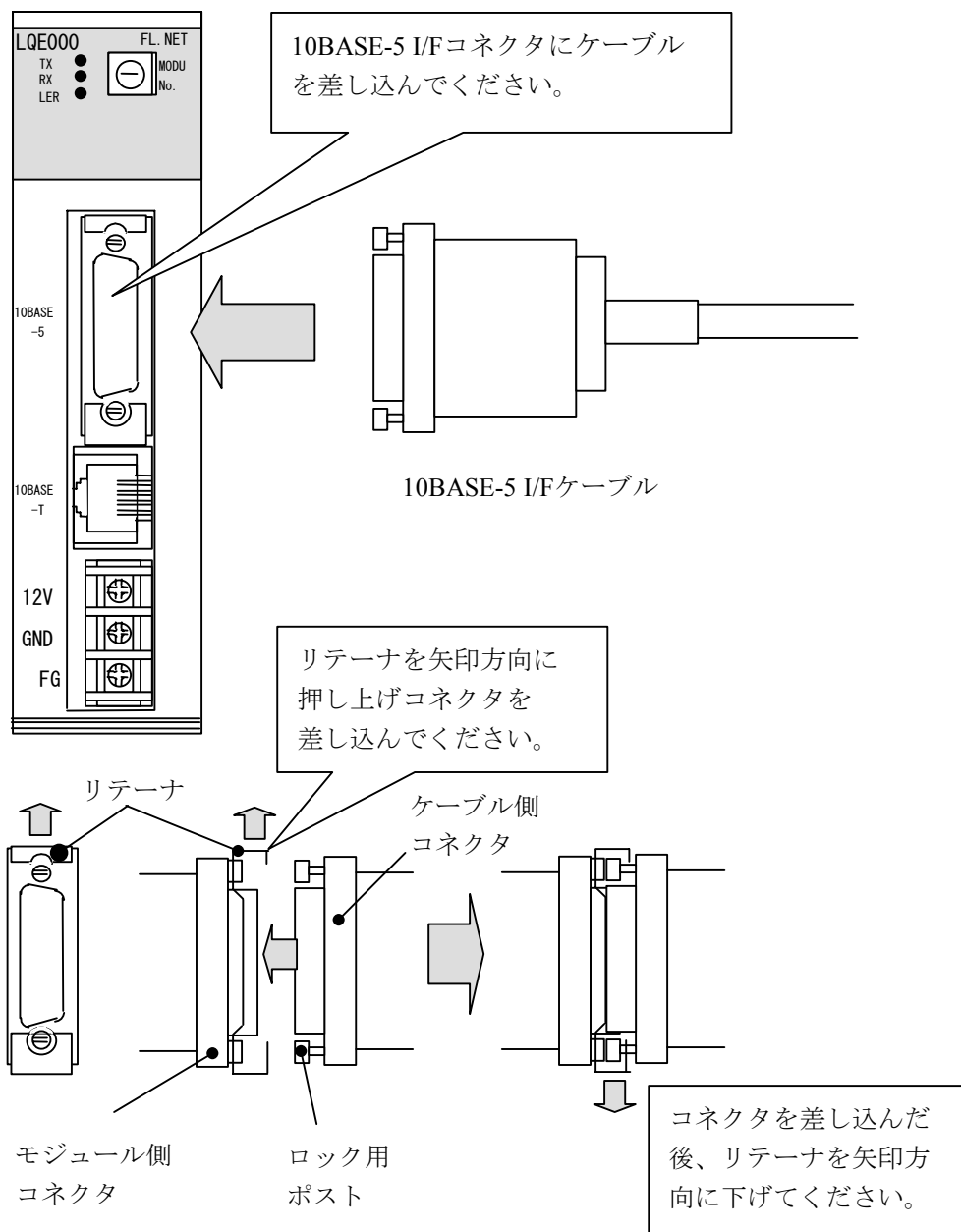


図5-1 モジュールの10BASE-5トランシーバケーブル接続

- 接触不良または断線により誤動作する恐れがあります。10BASE-5用コネクタを接続したとき、ロック用ポストがリテーナによってロックされているか確認してください。
- 通電中は10BASE-5コネクタに触れないでください。静電気などによりシステムが誤動作することがあります。

(2) 10BASE-Tケーブルの接続

モジュールNo.設定スイッチを2または3に設定した場合、このモジュールは10BASE-T I/Fコネクタを使用して、他モジュールと通信します。

その際、10BASE-Tケーブルを図5-2に示す位置に差し込んでください。

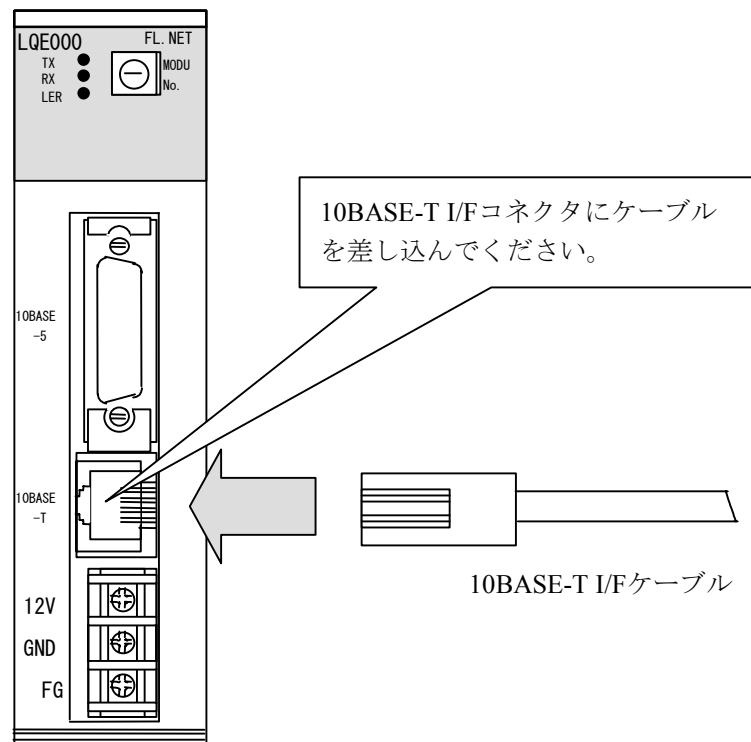


図5-2 モジュールの10BASE-Tケーブル接続

5 FL.NTEモジュールの配線方法

5.2 適用通信ケーブル

(1) 10BASE-5トランシーバケーブル

表5-1 モジュール適用通信ケーブル (10BASE-5トランシーバケーブル)

品名	形式	メーカー
トランシーバケーブル	HBN-TC-100	日立電線 (株)

(2) 10BASE-Tケーブル

表5-2 モジュール適用通信ケーブル (10BASE-Tツイストペアケーブル)

品名	形式	メーカー
ツイストペアケーブル	HUTP-CAT5 4P	日立電線 (株)

5.3 電源配線

10BASE-5を使用する場合、図5-3のように外部供給電源で電源供給してください。

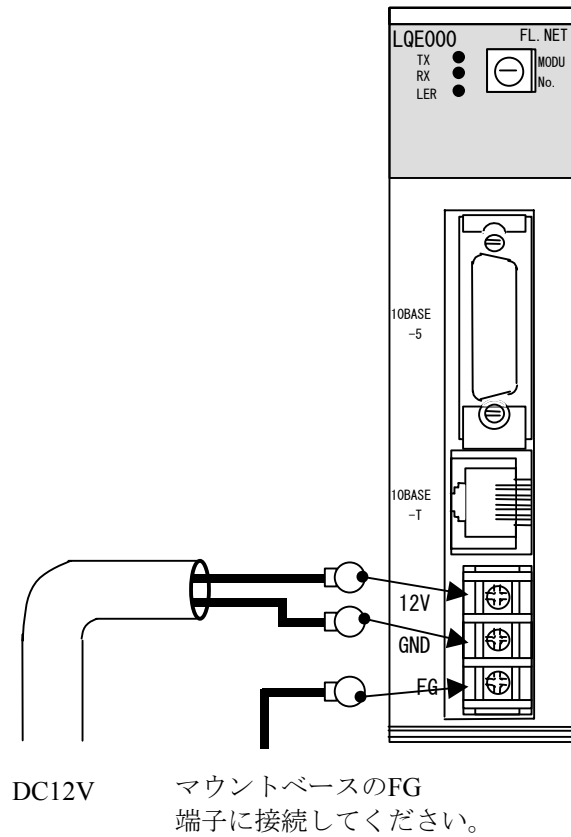


図5-3 モジュールの電源配線

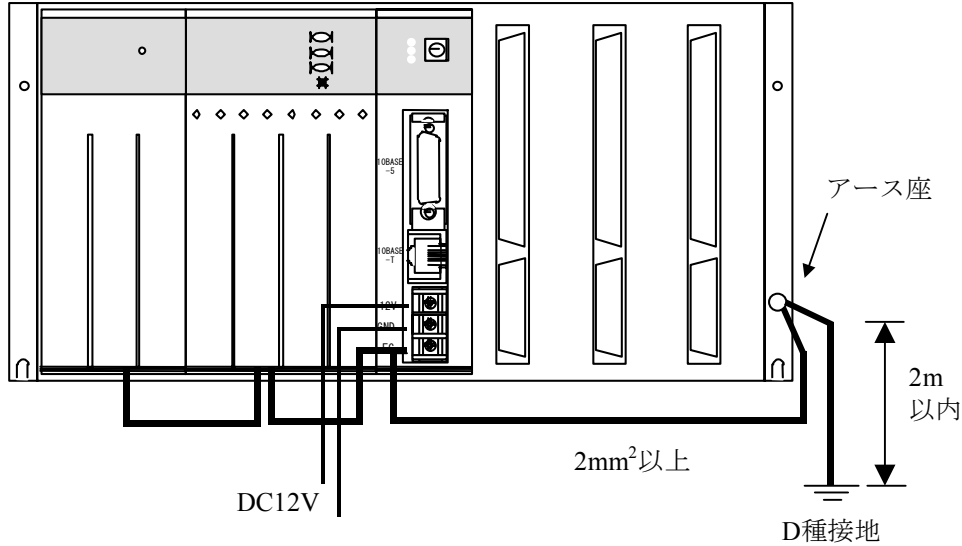
DC12V外部供給電源は、下記型式の電源を推奨します。推奨型式の電源を利用してください。
電源型式：HK-25A-12（メーカー：デンセイラムダ（株））

5 FL.NTEモジュールの配線方法

5.4 アース配線

10BASE-5を使用する場合は、図5-4に従いアース配線してください。

● 10BASE-5時のアース配線例



● 10BASE-T時のアース配線 (FL.NETモジュールのFGには配線しないでください。)

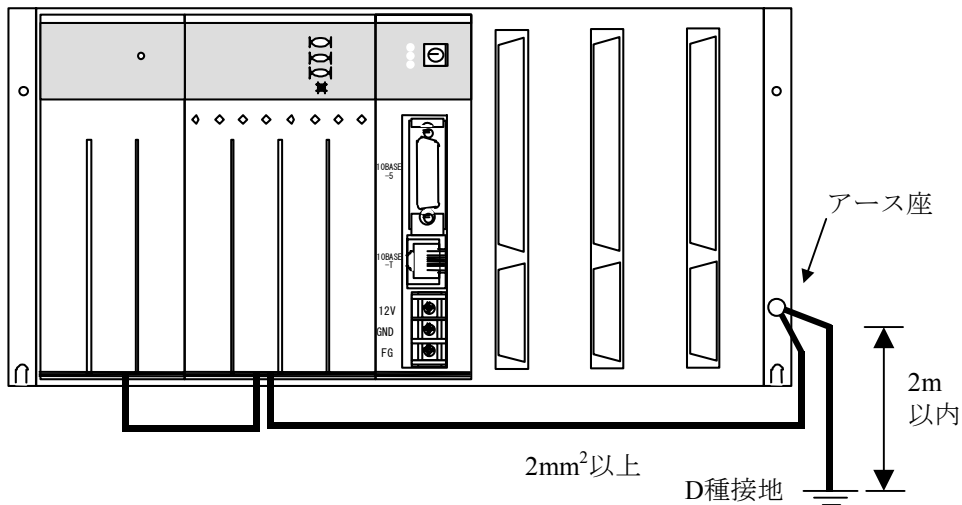


図5-4 ユニットのアース配線

強制

- FG (フレームグラウンド) のアース配線は、外部端子のある各モジュールのFG端子を、マウントベースのアース座に接続してください。アースの配線距離は2m以内とし、マウントベースのアース座からD種接地してください。
- アース線は、線径2mm²以上のものを用いてください。
- 通電中は10BASE-5コネクタに触れないでください。静電気などによりシステムが誤動作することがあります。

6 利用の手引き

6 利用の手引き

6.1 イーサネットについて

6.1.1 10BASE-5システム

基本構成は、図6-1のように最長500mの同軸ケーブルとそれに接続されるノードからなります。ノードはトランシーバケーブル（AUIケーブル）とトランシーバを介して同軸ケーブルに接続されます。トランシーバには、トランシーバケーブル（AUIケーブル）を1本だけ接続できるシングルポートトランシーバと、複数本接続できるマルチポートトランシーバの2種類があります。

この基本構成をセグメントといい、1セグメントのノード数は最大100台です。

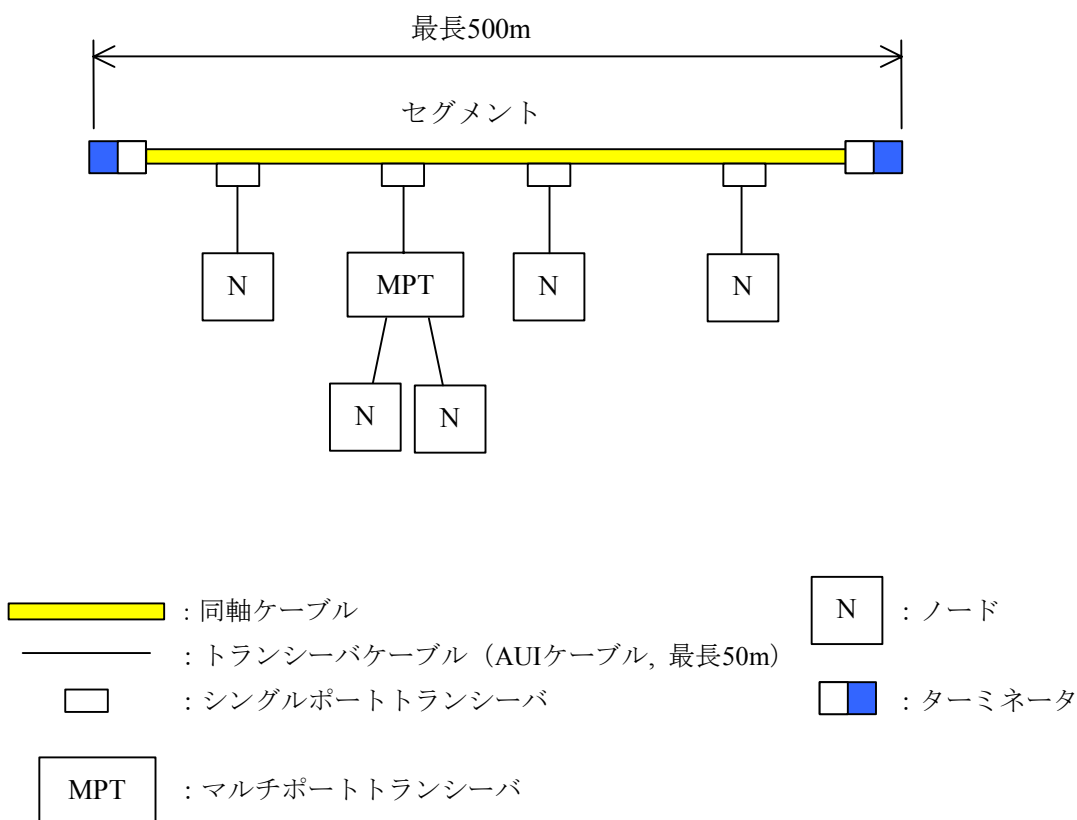


図6-1 10BASE-5システムの基本接続方法（リピータなし、最大距離500m）

ノード間の距離が500m以上となる場合は図6-2に示すようにリピータを接続して分岐状にセグメントの数を増やすことになります。図6-3はノード間の距離が最長1,500m以内のシステム例であり、任意に選んだ2つのノード間は、どの経路を通過してもリピータの数が2台以下となるように構成してください。

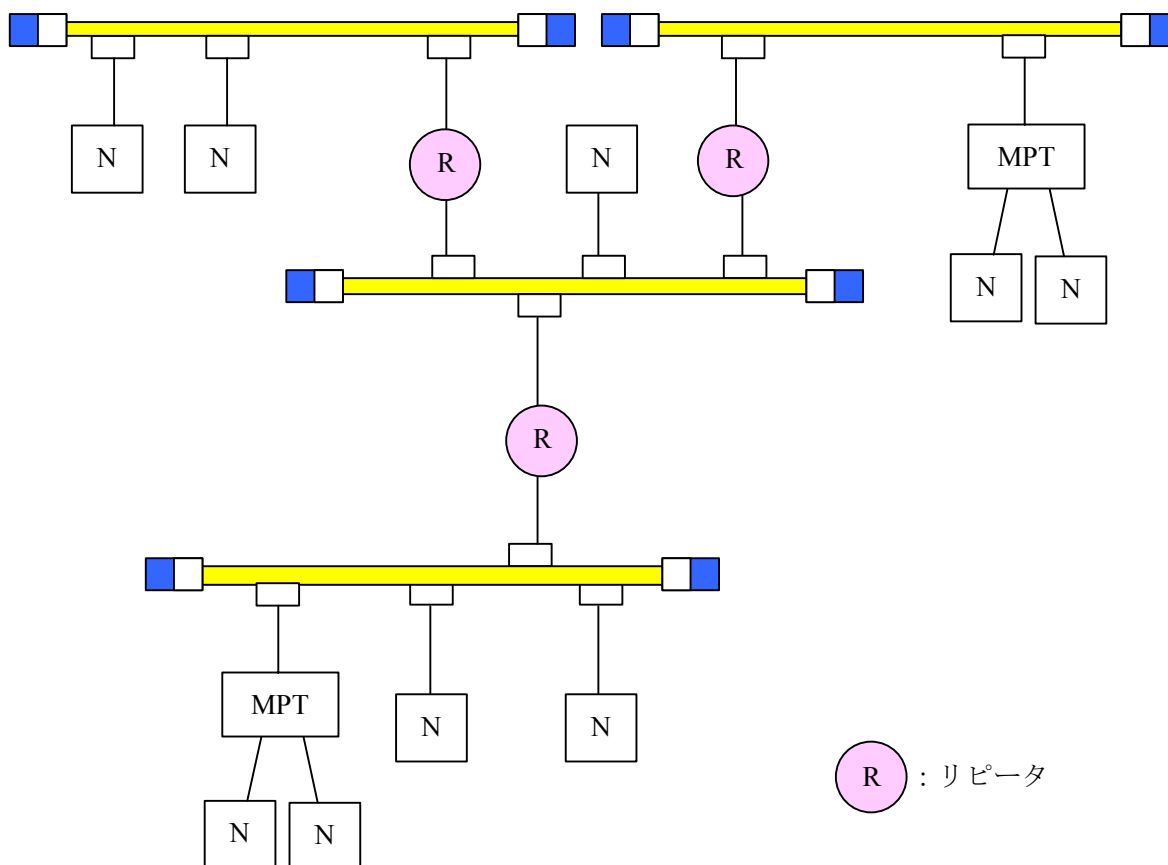


図6-2 10BASE-5システムの基本接続方法（リピータ使用、最大距離1,500m）

- リピータはトランシーバケーブルとトランシーバを介して同軸ケーブルに接続してください。
- リピータは、同軸セグメント中のどの位置のトランシーバにも取り付けられます。
- トランシーバの取り付け間隔は、2.5mの整数倍としてください。

6 利用の手引き

図6-3はノード間の距離を最長2,500mとした例です。伝送距離を大きくするため、リピータを両端に設置したリンクケーブル（同軸ケーブルの場合、最長500m）を用いており、これをリンクセグメントと呼びます。

リンクセグメントにはノードを接続せず、そのかわり両端のリピータを含めて点線で囲った部分を1台のリピータとして数えることができ、任意ノード間のリピータ合計台数の制限を軽減できます。

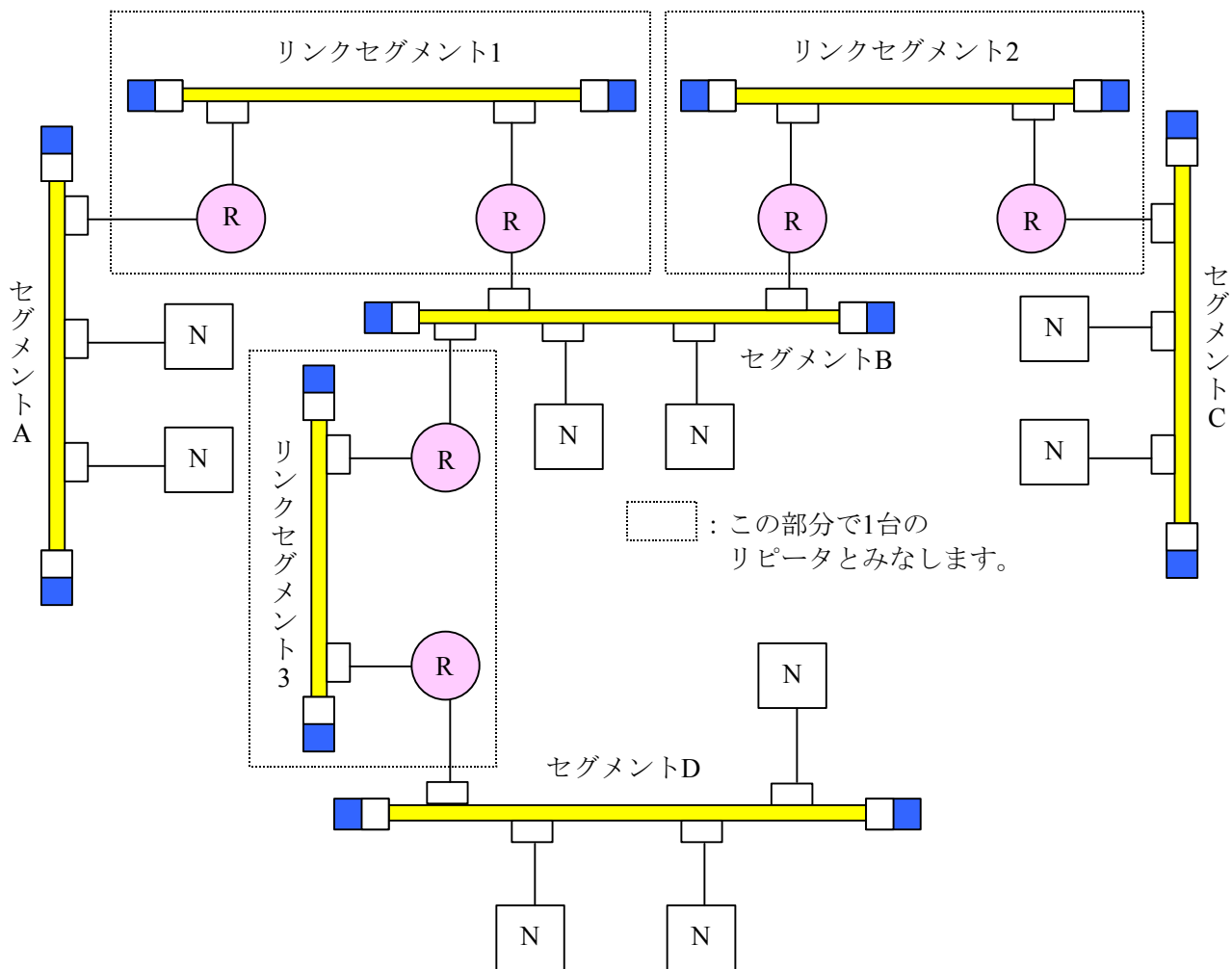


図6-3 10BASE-5システムの基本接続方法（リピータ使用、最大距離2,500m）

- リンクセグメントは最長500mです。
- リンクセグメントにはノードを接続しないでください。
- リンクセグメントは、両端のリピータを含めて図中 部をリピータ1個とみなします。
- 任意のノード間のリピータは、最大2個以下にしてください。
- リピータが2個以上接続できるセグメントは、1つのみとしてください。

システム構成上のパラメータを表6-1に示します。

表6-1 イーサネットのシステム構成上の一般仕様

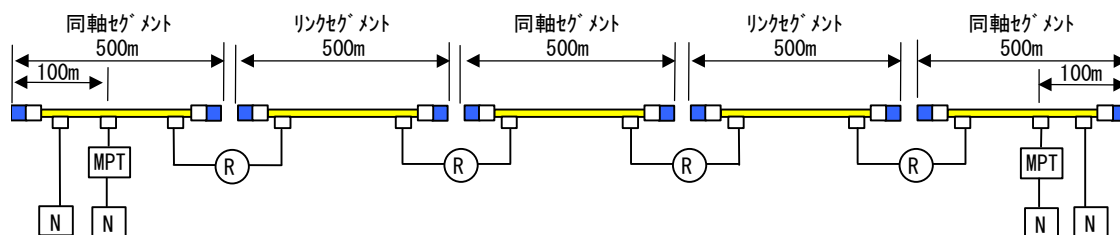
項目	仕様
セグメント最長	500m
セグメント内トランシーバ取り付け最大数	100台
ノード間最大距離	2,500m以下（トランシーバケーブルを除く）
システム最大ノード数	254台
トランシーバケーブル（AUIケーブル）最長	50m
トランシーバとリピータ間のケーブル長	2m以下（推奨値）
ノード間経路内リピータ最大数	2台（ただし、リンクセグメントは両端のリピータを含めて、全体を1台のリピータとみなします。）

マルチポートトランシーバの設置位置制限

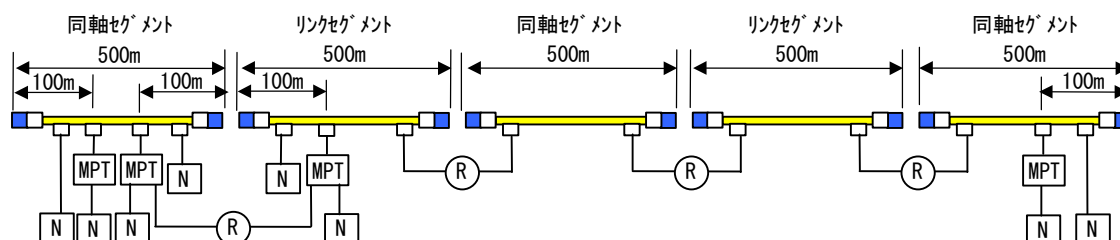
- 同軸ケーブル長として最長2,500m（5セグメント）で構成するシステムにおいて、最遠端の同軸ケーブルセグメント上にマルチポートトランシーバを設置する場合には、データの遅延時間が増加するためマルチポートトランシーバの設置位置に制限が生じます。マルチポートトランシーバを経由したノード間の最大距離は、マルチポートトランシーバを1台通過することにより、同軸ケーブル長に換算して100m減少します。したがって、あるノード間の同軸ケーブル長L[m]には下記制限があります。

$$L[m] \leq 2,500[m] - 100 \times N[m] \quad N: \text{経由するマルチポートトランシーバの数}$$

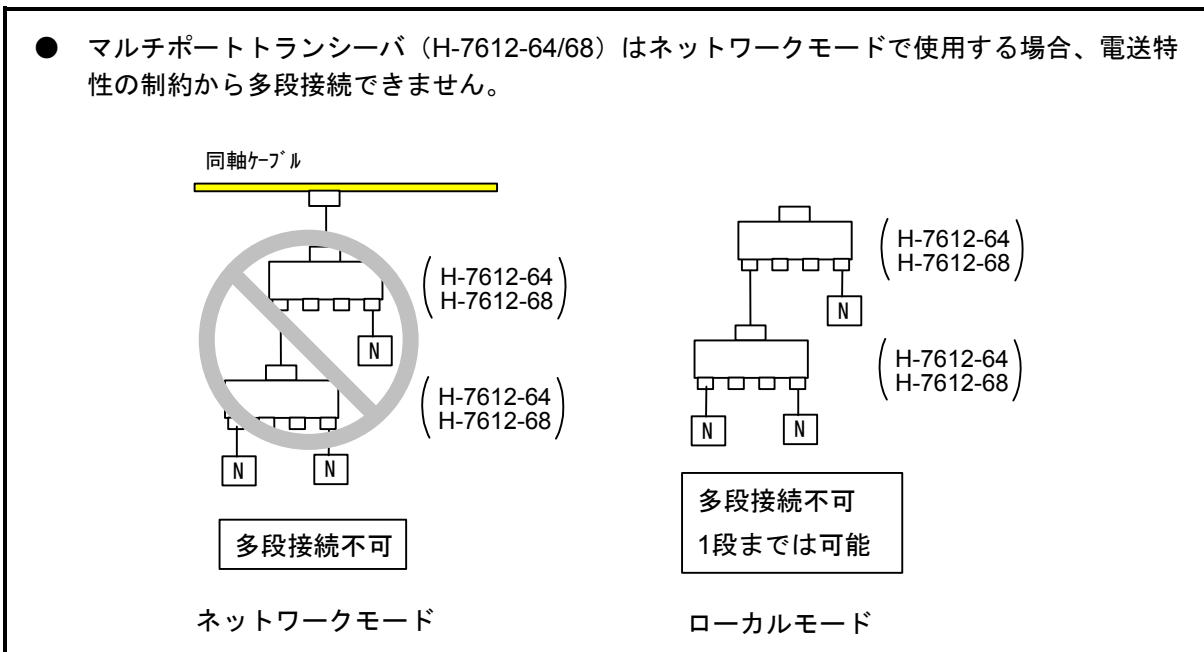
- 2,500mの同軸ケーブルで構成されるシステムにおいて、マルチポートトランシーバは最遠端の同軸ケーブルターミネータから100m以上内側（ノード間の距離を減少させる位置）に設定してください。



- 同様にマルチポートトランシーバを使用してセグメント間のリピータを接続する場合もマルチポートトランシーバ1台通過することにより、最遠端のノード間距離を100m減少させる位置にマルチポートトランシーバを設置する必要があります。



6 利用の手引き



6. 1. 2 10BASE-Tシステム

トランシーバにトランシーバケーブル（AUIケーブル）を経由してハブ（HUB）を接続することにより、ハブに複数のノードを接続できます。これを図6-4に示します。

ハブにノードを接続する場合には、ツイストペアケーブル（10BASE-T）を使用します。

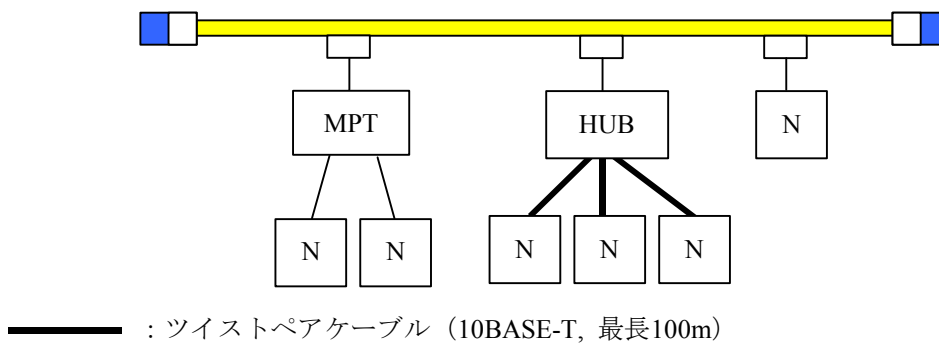


図6-4 10BASE-Tシステムの基本接続方法1

また、ノード間の距離が短い場合は、図6-5のように同軸ケーブルやトランシーバなしで、ハブにツイストペアケーブルを介してノードを接続できます。

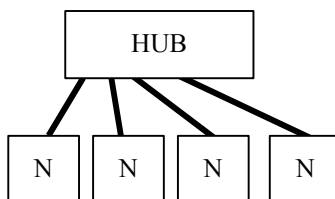
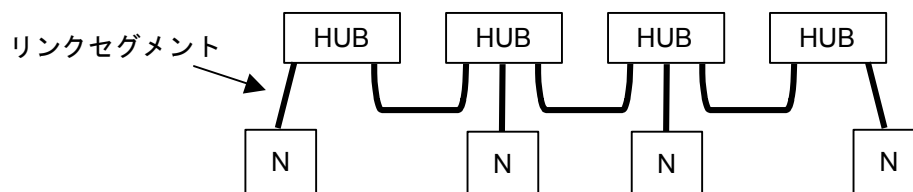


図6-5 10BASE-Tシステムの基本接続方法2

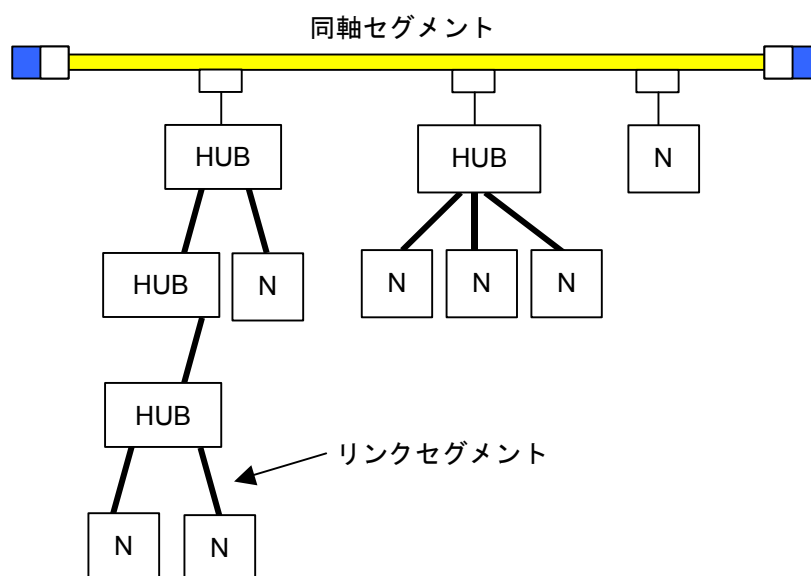
ハブ（HUB）使用時の制約

- ハブの多段接続について

ハブ（マルチポートリピータ）を多段接続して使用する場合は、どの任意のノード間の経路を取っても、通過するハブは最大4段、かつリンクセグメントは最大5本となるように構成してください。



同軸ケーブルに接続して使用する場合も同様にどの任意のノード間の経路を取っても、通過するハブは最大4段、かつセグメントは最大5本（同軸セグメントは3本まで）となるように構成してください。



6 利用の手引き

<IPアドレス>

UDP/IPでは、IPアドレスという32ビットの論理アドレスを使用します。

IPアドレスはネットワークアドレスとホストアドレスからなります。一般的にFA分野ではクラスCが使用されます。

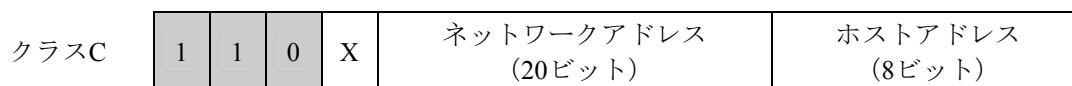


図6-6 イーサネットのIPアドレスのクラス区分

また、このアドレスは8ビットごとにピリオド“.”で区切り、10進数で表します。例えば、クラスCでは以下のように表現します。



FL.NETでは、デフォルト値が192.168.250.N（Nはノード番号1~254）となっています。

図6-7 イーサネットのクラスCのIPアドレス例

6.2 FL-netについて

6.2.1 FL-netの概要

(1) FL-netのコンセプト

FL-netは、イーサネットをベースとしたFAコントロールネットワークです。

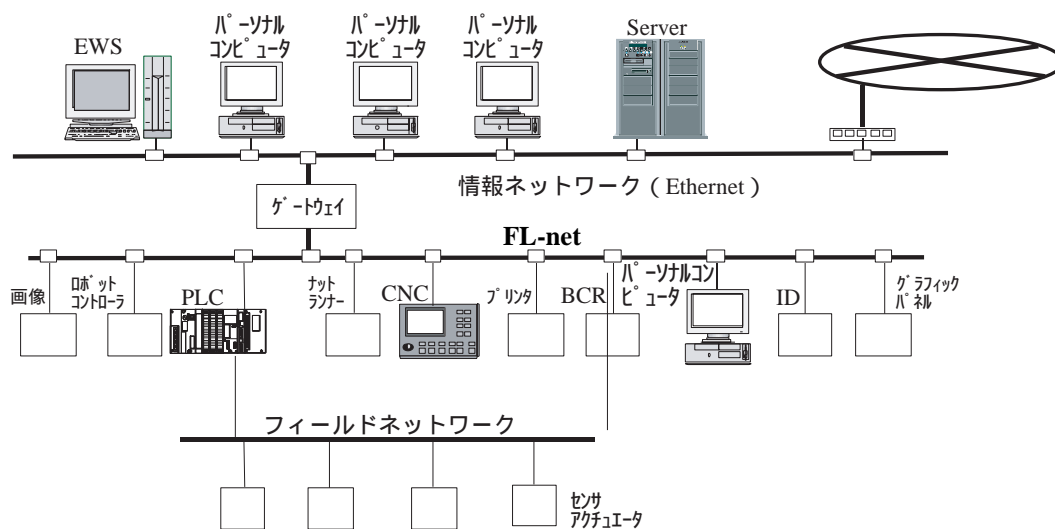
FL-netは、サイクリック伝送機能およびメッセージ伝送機能を持っています。

FL-netの基本的な考え方は次のとおりです。

- ・イーサネットをFAコントローラ間の通信媒体（物理レベル、データリンク）にしています。
- ・イーサネット上で普及しているUDP/IPを使用し、基本的なデータ送達手段を実現しています。
- ・上記の基本的なデータ送達手段を使用しつつ、ネットワーク内各ノードの通信媒体アクセスを管理／制御（衝突回避）して、一定時間内の伝送を保証します。

FL-netの対象は、生産システムにおけるプログラマブルコントローラ（PLC）、ロボットコントローラ（RC）、数値制御装置（CNC）などの制御装置や制御用パーソナルコンピュータ間におけるデータ交換を行うためのFAコントロールネットワークです。

図6-8にFL-netの位置付けを示します。



BCR : バーコードリーダー

ID : IDコントローラ

図6-8 FL-netのコンセプト

6 利用の手引き

(2) FL-netのプロトコル

FL-netは、図6-9のように6つのプロトコル層から構成されます。

トランスポート層、ネットワーク層ではUDP/IPを使用し、データリンク層、物理層ではイーサネットを使用します。

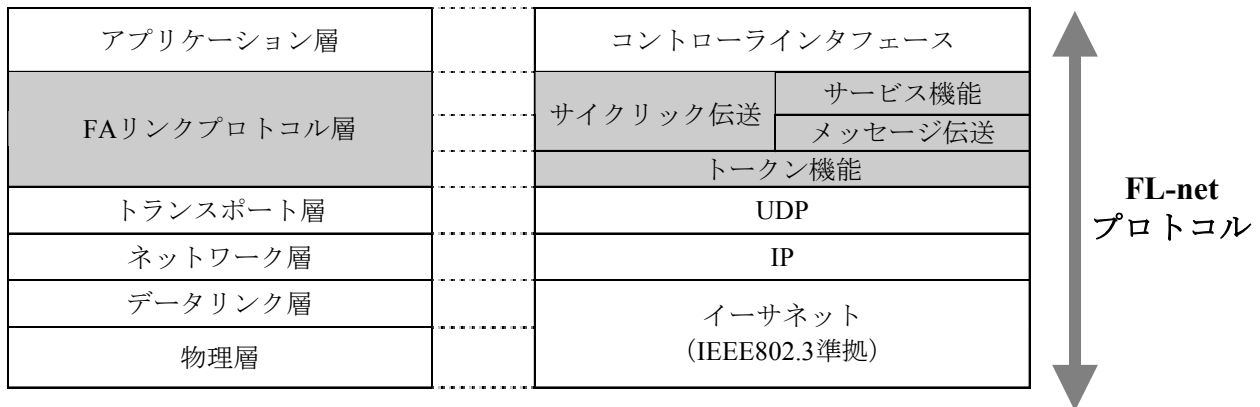


図6-9 FAリンクプロトコル

(3) FL-net伝送方式の特長

FL-netのFAリンクプロトコル層の特長を以下に示します。

- ・マスタレス、トークン方式による送出管理を行い衝突を回避しています。
- ・トークンを一定時間で周廻させることによって、リフレッシュサイクル時間が規定可能です。
- ・定められたトークンをサイクリックデータとともに送信しています。
- ・立ち上がり時一番若いノードからトークンを送信しています。
- ・一定時間トークンが送信されない場合、次ノードがトークンを送信します。
- ・マスタレス、トークン方式によって、一部のノードが故障してもネットワークが停止することはありません。
- ・運転モード (RUN/STOP) / ハードウェア異常 (ALARM) などの情報の管理テーブルを用意し他ノードの動作状態を参照できます。

(4) FL-netのIPアドレス

IPアドレスとは、IP（インターネットプロトコル）による伝送を行う場合に、特定のノード（ステーション）を指し示す「アドレス」です。このため、IPアドレスは重複しないように設定／管理する必要があります。

FL-netの各ノードには、IPアドレスを個別に設定する必要があり、FL-netプロトコル規約によりクラスCのIPアドレスを使用するよう推奨します。

FL-netのIPアドレスのデフォルト値は、192.168.250.***を使用しています。***部分は、ノード番号になります（デフォルト値の値は、FL-netプロトコル規約の推奨値です）。

ネットワークアドレス	ホスト番号 (ノード番号)
192.168.250.	n (n : 1~254)

図6-10 FL-netのIPアドレス

6.2.2 接続台数とノード番号

最大接続台数は254台です。ノード番号は1~254を使用します。

ノード番号：1~249 通常のFL-net機器用

ノード番号：250~254 FL-netメンテナンス用

ノード番号：255 FL-netの内部で使用します。ユーザは使用できません（グローバルアドレスのブロードキャスト伝送に使用）。

ノード番号：0 FL-netの内部で使用します。ユーザは使用できません。

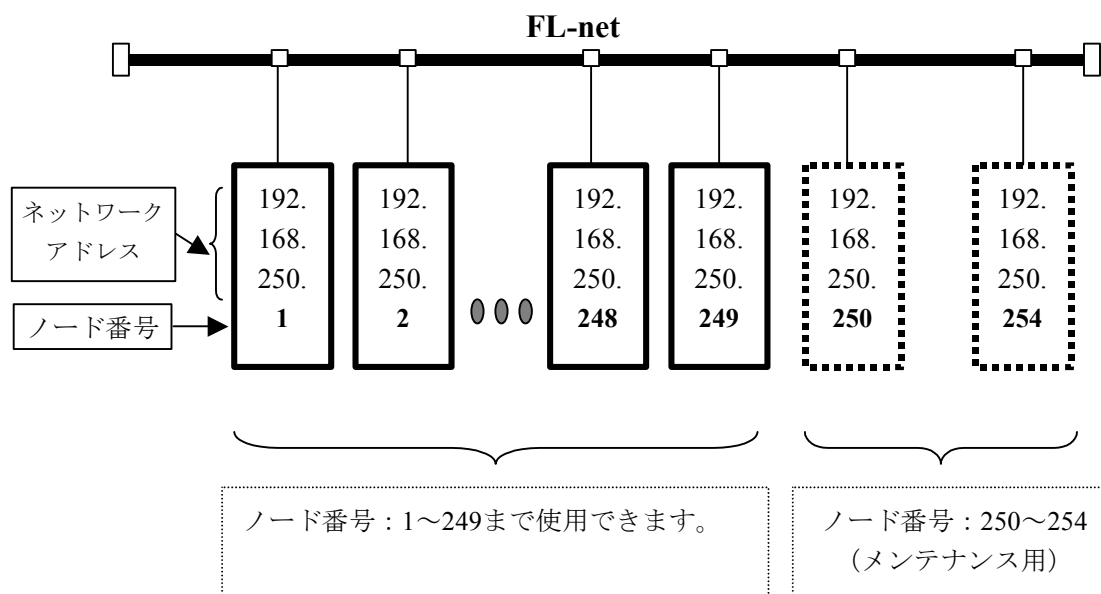


図6-11 FL-netの接続台数とノード番号

6 利用の手引き

6.2.3 データ通信の種類

FL-netのデータ通信は、サイクリック伝送とメッセージ伝送をサポートしています。

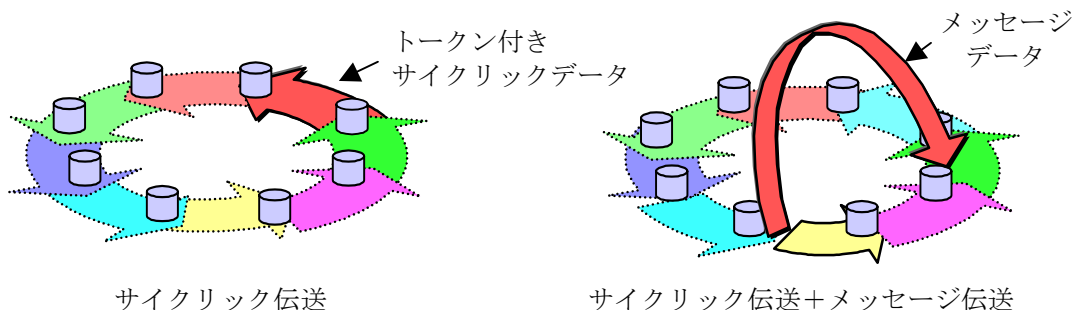


図6-12 FL-netのデータ通信の種類

(1) サイクリック伝送

サイクリック伝送は、周期的なデータの伝送を行います。各ノードは、コモンメモリ（共通メモリ）を介して、データを共有できます。

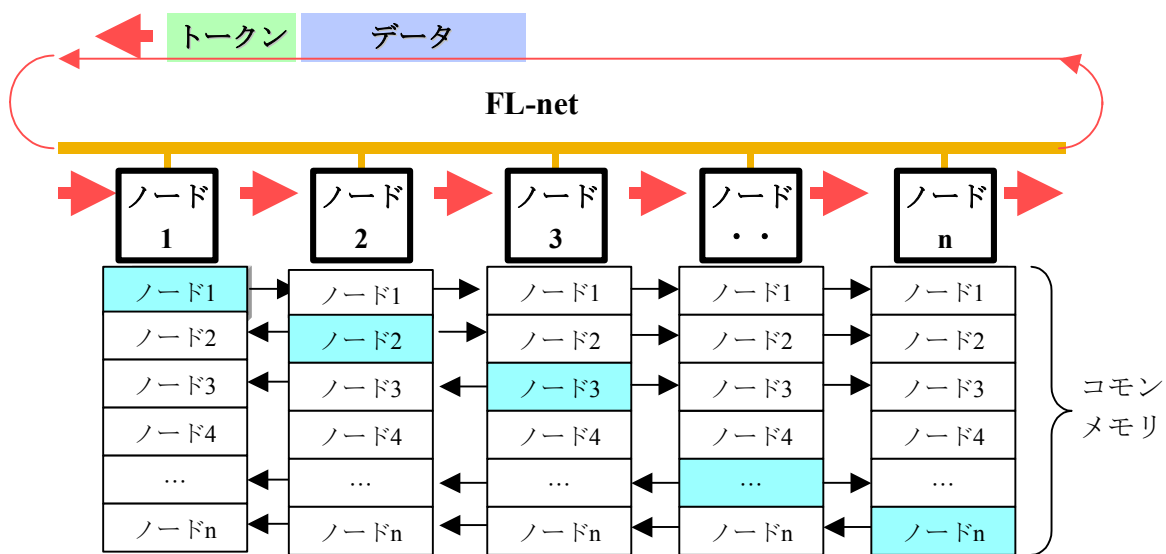


図6-13 コモンメモリとサイクリック伝送の例

(2) メッセージ伝送

メッセージ伝送は非周期的なデータの伝送を行います。

通常は、送信要求があったときに、特定のノードに向けて通信を行います。

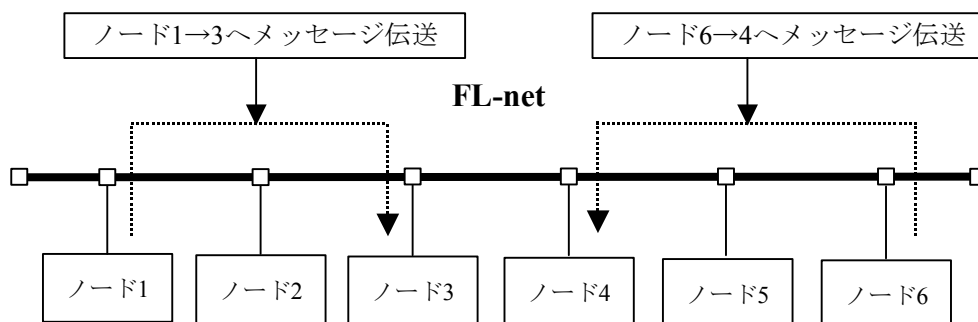


図 6-14 メッセージ伝送の例

6.2.4 伝送データ量

(1) サイクリック伝送

ネットワーク全体で8kビット+8kワード=8.5kワードを持っています。

1ノードあたりで利用できる送信データ量は、最大8.5kワードです。

ただし、1ワードは2バイトです。

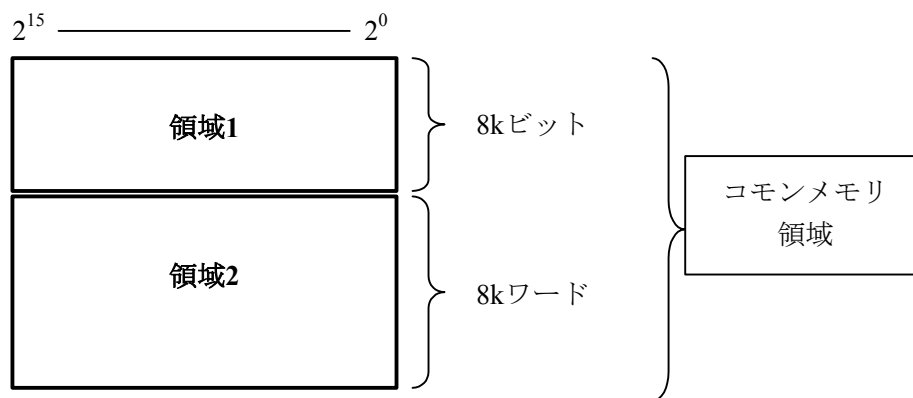


図 6-15 サイクリック伝送のデータ量

6 利用の手引き

(2) メッセージ伝送

1メッセージのフレームデータ量は、最大1024バイトです（ヘッダ部分は含みません）。

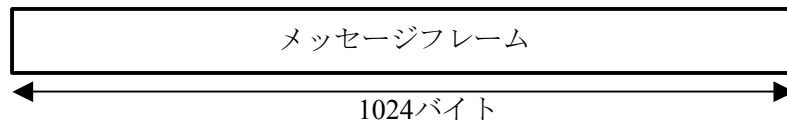


図6-16 メッセージ伝送のデータ量

6.2.5 転送周期とその監視

サイクリックデータ通信は、ほぼ一定周期でコモンメモリをリフレッシュします。単発のメッセージ通信により、コモンメモリのリフレッシュ時間がリフレッシュサイクル許容時間を超えないように、メッセージ通信の送信をコントロールしています。

各ノードは、自ノード宛のトークン受信から次の自ノード宛のトークン受信までにネットワークに流れるメッセージ通信のフレームを常時監視しています。この1周期の間にネットワークに1つもメッセージ通信のフレームが流れないとき、この1周期時間の120%の値をリフレッシュサイクル許容時間とします。

上記の監視処理によって、リフレッシュサイクル許容時間は、ネットワークに加入するノード数によって動的に決定されます。

例：ノード5台でサイクリックデータ通信のみ行っている場合
(どのノードもメッセージ通信は行わない)

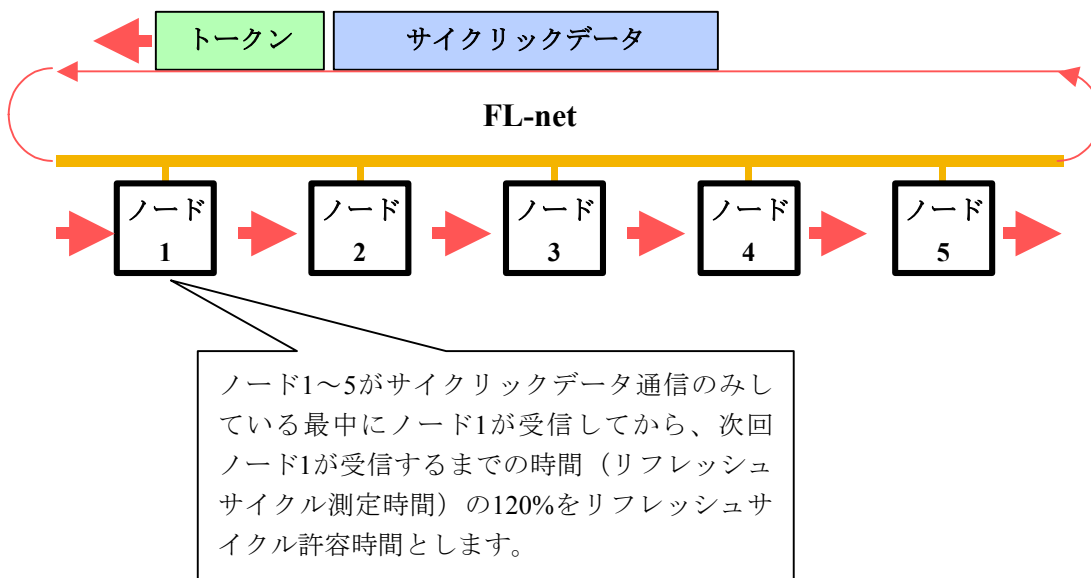


図6-17 リフレッシュサイクル許容時間の例

計算方法に関しては、「6.4.7 FL.NETモジュールの通信時間」を参照してください。

6. 2. 6 データ領域とメモリ

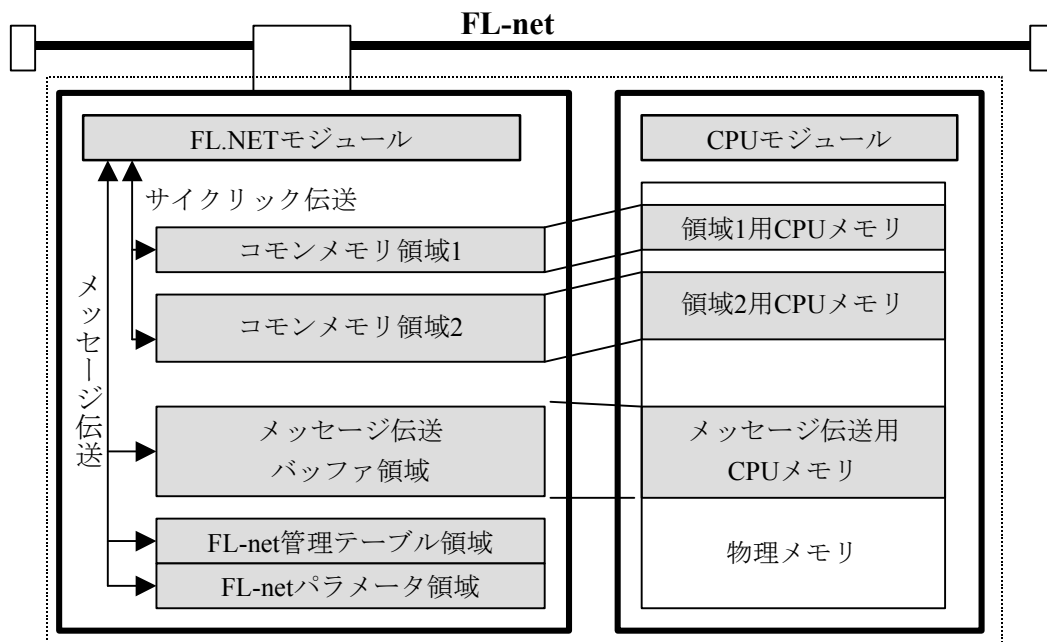


図6-18 データ領域とメモリ

6 利用の手引き

6. 2. 7 通信管理テーブル

ノードの状態管理は、自ノード管理テーブル、参加ノード管理テーブル、ネットワーク管理テーブルで行っています。

(1) 自ノード管理テーブル

自ノード管理テーブルは、自ノードの設定について管理します。

表 6-2 自ノード管理テーブル

名称	バイト長	内容・データ範囲
ノード番号	(1バイト)	1~254
コモンメモリにおける領域1・データ先頭アドレス	(2バイト)	ワード・アドレス (0~0x1FF)
コモンメモリにおける領域1・データサイズ	(2バイト)	サイズ (0~0x1FF)
コモンメモリにおける領域2・データ先頭アドレス	(2バイト)	ワード・アドレス (0~0x1FFF)
コモンメモリにおける領域2・データサイズ	(2バイト)	サイズ (0~0x1FFF)
上位層の状態	(2バイト)	RUN/STOP/ALARM/WARNING/NORMAL
トークン監視時間	(1バイト)	1s単位
最小許容フレーム間隔	(1バイト)	100 μ s単位
ベンダ名	(10バイト)	ベンダの名称
メーカー型式	(10バイト)	メーカーの型式、デバイスの名称
ノード名 (設備名)	(10バイト)	ユーザ設定によるノードの名称
プロトコル・バージョン	(1バイト)	0x80固定
FAリンクの状態	(1バイト)	加入/離脱など
自ノードの状態	(1バイト)	ノード番号重複検知など

(2) 参加ノード管理テーブル

参加ノード管理テーブルは、ネットワークに加入しているノードに関する情報を管理します。

表6-3 参加ノード管理テーブル

名称	バイト長	内容・データ範囲
ノード番号	(1バイト)	1~254
上位層の状態	(2バイト)	RUN/STOP/ALARM/WARNING/NORMAL
コモンメモリにおける 領域1・データ先頭アドレス	(2バイト)	ワード・アドレス (0~0x1FF)
コモンメモリにおける 領域1・データサイズ	(2バイト)	サイズ (0~0x1FF)
コモンメモリにおける 領域2・データ先頭アドレス	(2バイト)	ワード・アドレス (0~0x1FFF)
コモンメモリにおける 領域2・データサイズ	(2バイト)	サイズ (0~0x1FFF)
リフレッシュサイクル許容時間	(2バイト)	1s単位
トークン監視時間	(1バイト)	1s単位
最小許容フレーム間隔	(1バイト)	100 μ s単位
リンクの状態	(1バイト)	加入/離脱情報など

(3) ネットワーク管理テーブル

ネットワーク管理テーブルは、ネットワークに共通する情報を管理します。

表6-4 ネットワーク管理テーブル

名称	バイト長	内容・データ範囲
トークン保持ノード番号	(1バイト)	現在トークンを保持しているノード
最小許容フレーム間隔	(1バイト)	100 μ s単位
リフレッシュサイクル許容時間	(2バイト)	1s単位
リフレッシュサイクル 測定時間 (現在値)	(2バイト)	1s単位
リフレッシュサイクル 測定時間 (最大値)	(2バイト)	1s単位
リフレッシュサイクル 測定時間 (最小値)	(2バイト)	1s単位

6 利用の手引き

6. 2. 8 サイクリック伝送と領域

(1) サイクリック伝送概要

サイクリック伝送とは、コモンメモリを使用し周期的にデータ交換を行う機能です。

- ・ノードがトークンを保持するときに送信します。
- ・ネットワークに参加するノードでサイクリック伝送を行わないものも認めます。
- ・トークンを保持したときに、送信すべきサイクリックデータをすべて送信します。

トークン : トークンは、基本的にネットワークに1つだけが存在します。もしも、ネットワークに2つ以上のトークンが存在した場合、ノードは宛先ノード番号が小さい方を優先し、他方を破棄します。

トークンフレーム : トークンを含むフレーム（トークンフレーム）には、トークンの宛先ノード番号とトークン送出ノード番号があります。各ノードは、受信したトークンフレームのトークンの宛先ノード番号と一致した場合にトークン保持ノードになります。

トークンの順序 : トークンのローテーションの順番は、ノード番号によって決まります。各ノードは参加ノード管理テーブルに登録されているノードの中の昇順でトークンのローテーションを行います。最大ノード番号のノードは、最小ノード番号のノードにトークンを渡します。

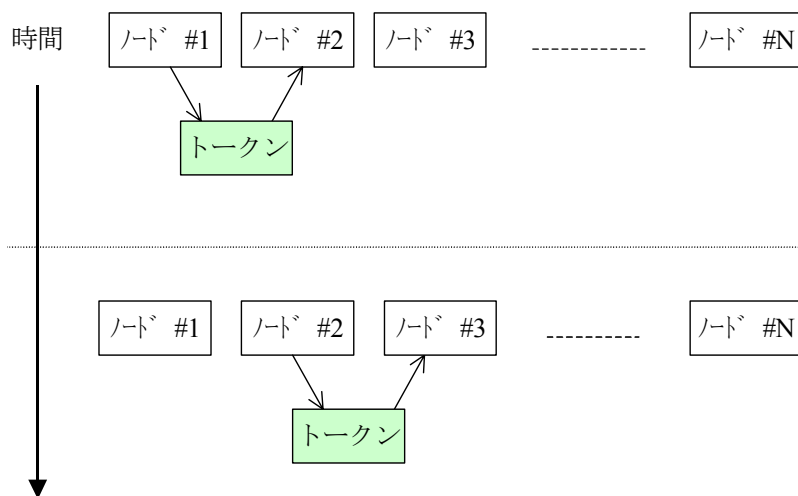


図 6-19 トークン巡回とサイクリック伝送1

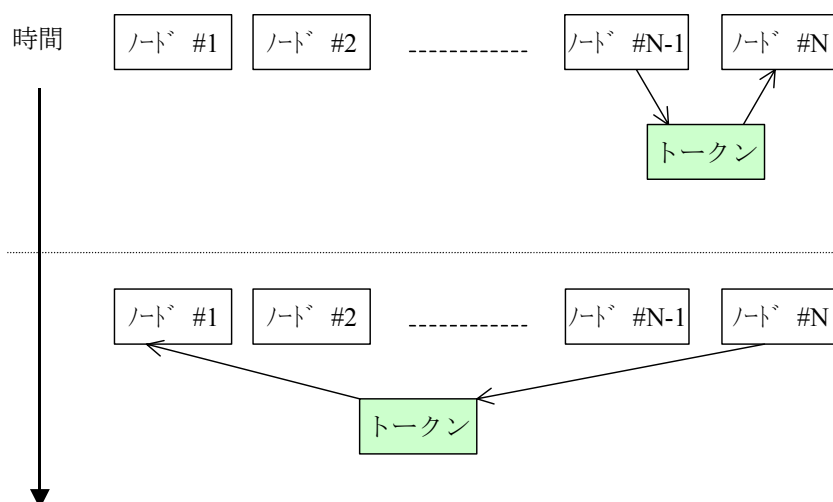


図 6-20 トークン巡回とサイクリック伝送2

6 利用の手引き

(2) コモンメモリ

コモンメモリの考え方は次のとおりです。

- サイクリック伝送を行うノード間で、共通のメモリとして扱うことのできる機能を提供します。
- 1つのノードについて2種類の領域（領域1、領域2）を割り付けられます。
- 1つのノードが送信する領域が1フレームによる伝送サイズ、すなわち1024バイトを超えるとき、複数のフレームによってデータを伝送します。
- 複数の分割されたデータのフレームを受信するとき、コモンメモリは1つのノードからくるすべてのフレームの受信完了まで、コモンメモリを更新しません。すなわちノード単位の同時性を保証します。
- 1ノードの通信部が用意するコモンメモリのための容量は、8kビット+8kワード=8.5kワードの固定サイズです。
- コモンメモリの内、1ノードの送信領域として領域1、領域2とも最大領域の範囲内で任意に設定することができます。
- 一定周期で各ノードは、データをブロードキャストすることで、システム全体で同じデータを共有する機能を提供します。FL-net上の各ノードは互いに重複しない送信領域を分担して受け持ち、データの交換を行います。コモンメモリの動作において、あるノードに割り当てられた送信領域は、他ノードにとっては受信領域となります。



図6-21 サイクリック伝送のコモンメモリ領域例1

コモンメモリは受信領域のみも使用できます。

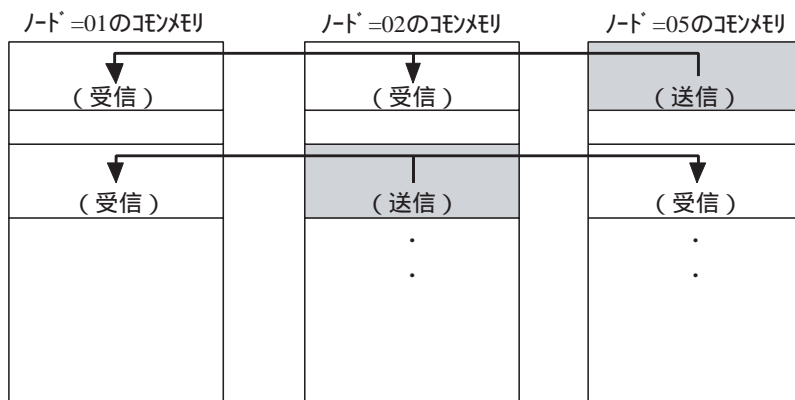


図 6-22 サイクリック伝送のコモンメモリ領域例2

(3) 領域1と領域2

1つのノードは、領域1と領域2という2つのデータ領域をコモンメモリに割り付けできます。送信領域の設定は、領域の先頭アドレスとサイズによって行います。

領域のアクセスは、ワード・アドレスとします。領域1は、0.5kワード、領域2は、8kワードから成り立っています。

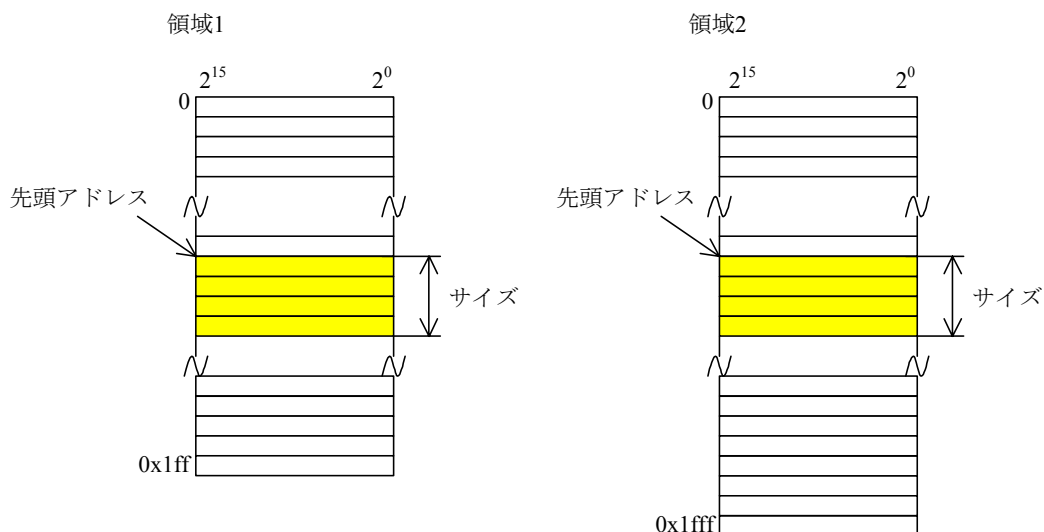


図 6-23 コモンメモリ領域1と領域2

6 利用の手引き

(4) データの同時性保証

サイクリック伝送では、送信するデータ量によってフレームを複数に分割します。

次の手順でノード単位の共通メモリ（FL.NET内のメモリ）の同時性を保証します。

● 送出タイミング

上位層からのデータ送信要求時、自ノードのサイクリックデータをバッファにコピーし、送信準備を行い順次送信します。送信ノードが持っているデータサイズが1フレームで送信できるサイズより大きいとき、バッファのデータを複数のフレームに分割して送信します。

● 受信時のリフレッシュタイミング

受信ノードは、1つのノードからのサイクリックデータをすべて受信完了した時点で、上位層と同期を取りながら対応する領域を更新します。

サイクリックデータが複数のフレームに分割して送信されてくるときも、領域の更新は、1つのノードから送信されるフレームをすべて受信終了した時点で行います。ノードから分割されて送られてくるフレームがすべて揃わなかったときは、そのノードからの全データは破棄します。

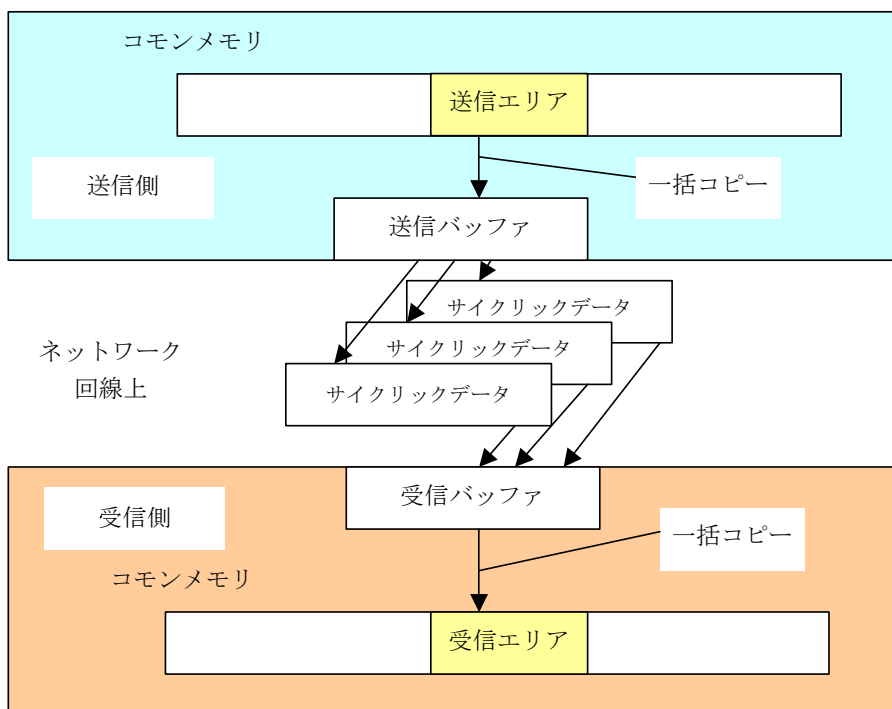


図6-24 データの同時性保証

6.2.9 メッセージ伝送

(1) メッセージ伝送概要

メッセージ伝送とは、ノード間に発生する非同期的なデータ交換をサポートする機能です。

メッセージ伝送の基本機能は、以下のとおりとなります。

- ノードがトークンを受けたとき、サイクリックフレーム送信の前に最大1フレームだけ送信できます。
- 1回の送信で送信できるデータ量は、最大1024バイトです。
- サイクリック伝送のリフレッシュサイクル許容時間を超えないためのアルゴリズムを持ちます。
- 指定された相手ノードだけに送信する1対1伝送と、すべてのノードに送信する1対n伝送の機能を持ちます。
- 1対1メッセージ伝送において、相手先がデータを正しく受信したか確認する送達確認の機能を持ちます。

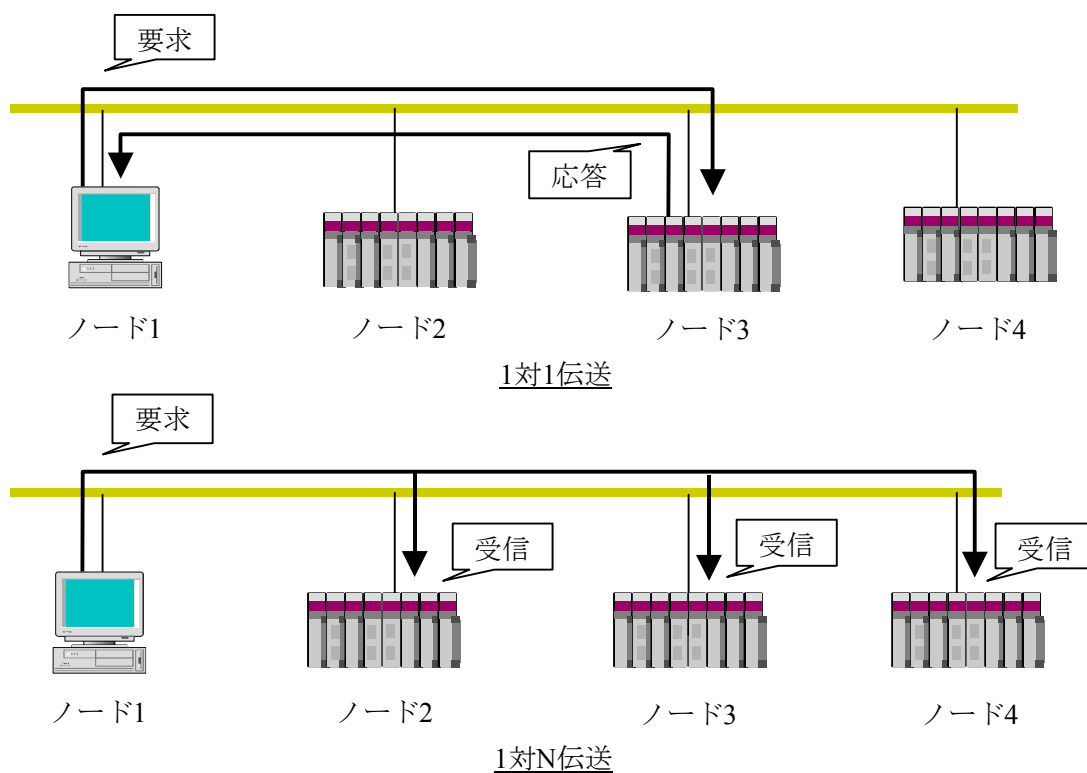


図 6-25 メッセージ伝送概要

6 利用の手引き

(2) サポートメッセージ一覧

S10mini FL.NET (LQE000) にてサポートしているメッセージの種別を表6-5に示します。

表6-5 サポートメッセージ伝送一覧

メッセージ	要求	応答
バイトブロック読み出し	×	×
バイトブロック書き込み	×	×
ワードブロック読み出し	○	○
ワードブロック書き込み	○	○
ネットワークパラメータ読み出し	○	○
ネットワークパラメータ書き込み	○	○
運転/停止指令	○	×
プロファイル読み出し	×	○
通信ログデータの読み出し	○	○
通信ログデータのクリア	○	○
メッセージ折り返し	○	○
透過型メッセージ	○	○

○：サポート、×：未サポート

(3) メッセージ機能説明

● バイトブロック読み出し

ネットワークから相手ノードが持つ仮想アドレス空間（32ビットアドレス空間）に対して、バイト単位（1アドレス8ビット単位）で読み出すメッセージ機能です。この機能は、S10mini FL.NETではサポートしていません。

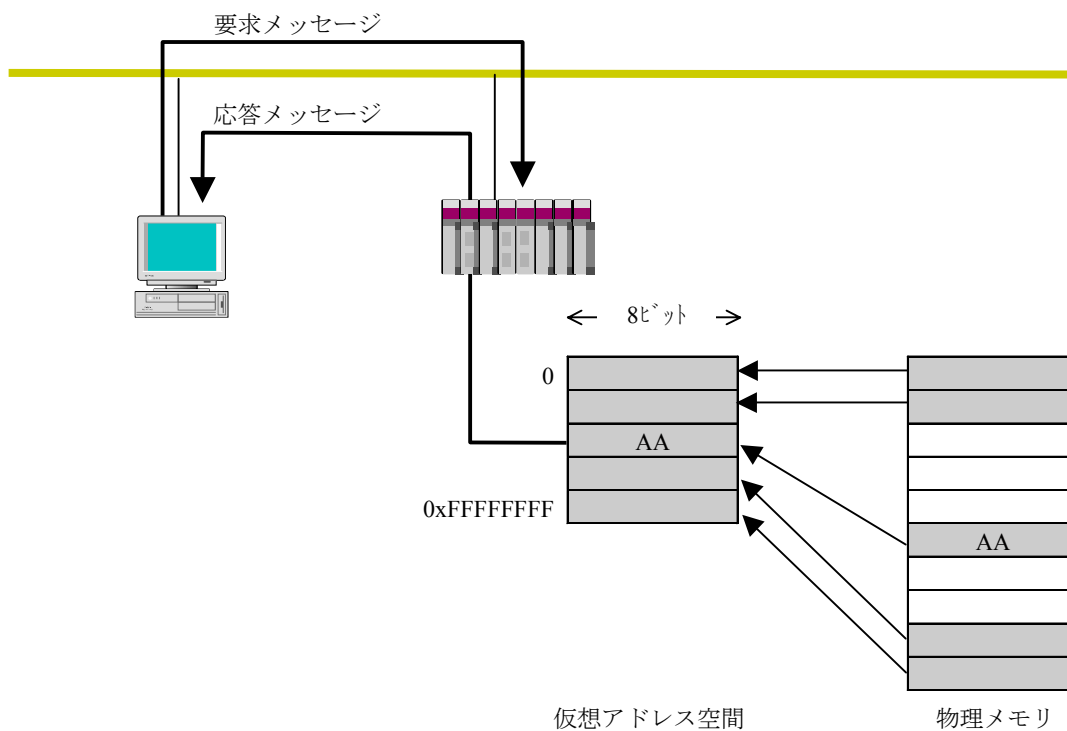


図 6-26 バイトブロック読み出し

6 利用の手引き

● バイトブロック書き込み

ネットワークから相手ノードが持つ仮想アドレス空間（32ビットアドレス空間）に対して、バイト単位（1アドレス8ビット単位）で書き込むメッセージ機能です。この機能は、S10mini FL.NETではサポートしていません。

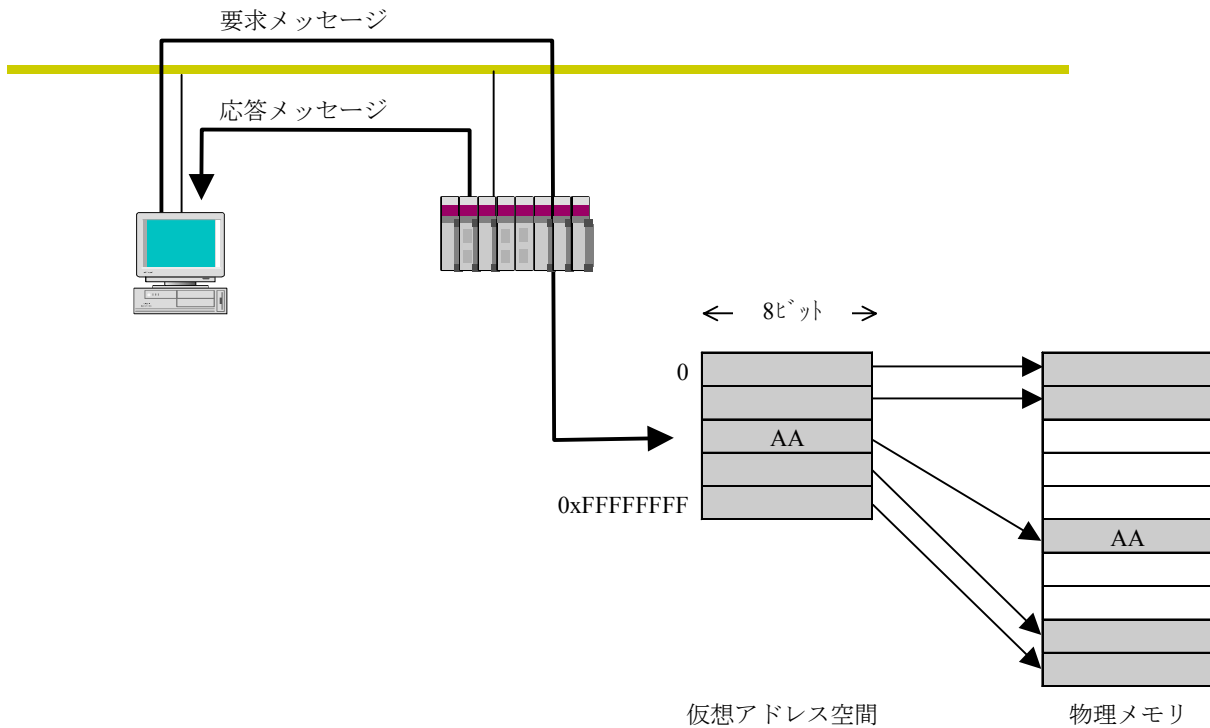


図6-27 バイトブロック書き込み

● ワードブロック読み出し

ネットワークから相手ノードが持つ仮想アドレス空間（32ビットアドレス空間）に対して、ワード単位（1アドレス16ビット単位）で読み出すメッセージ機能です。使用方法に関しては、「6.4 FL.NETの使い方」を参照してください。

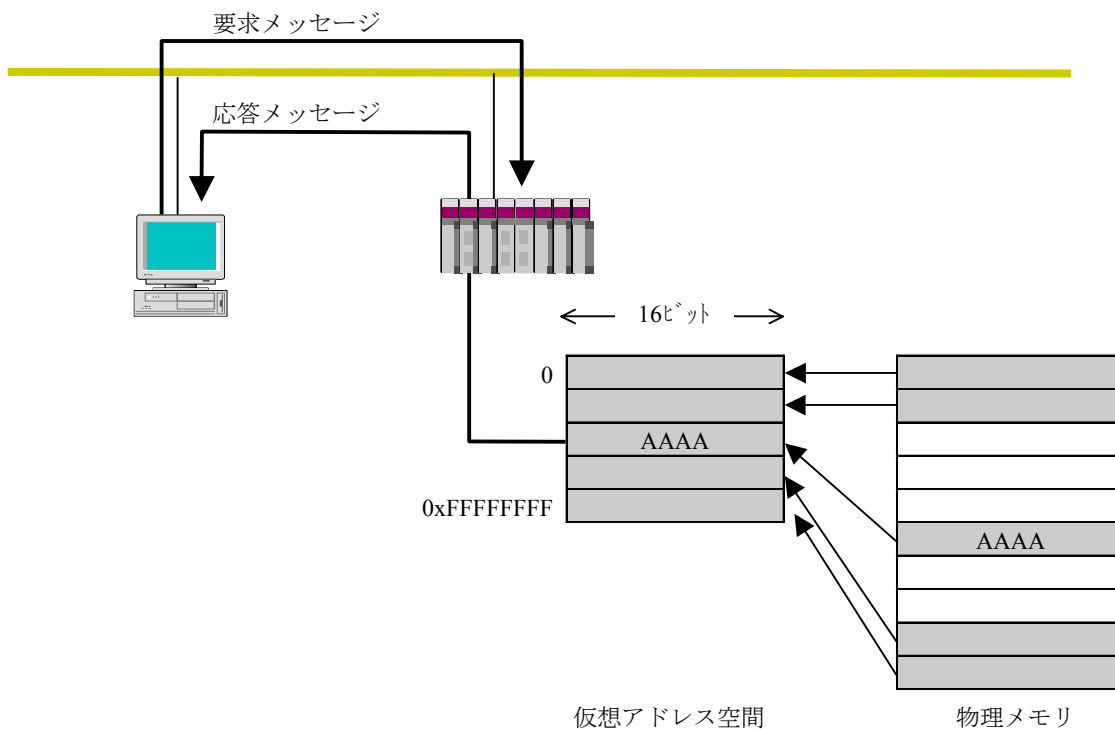


図6-28 ワードブロック読み出し

6 利用の手引き

● ワードブロック書き込み

ネットワークから相手ノードが持つ仮想アドレス空間（32ビットアドレス空間）に対して、ワード単位（1アドレス16ビット単位）で書き込むメッセージ機能です。使用方法に関しては、「6.4 FL.NETの使い方」を参照してください。

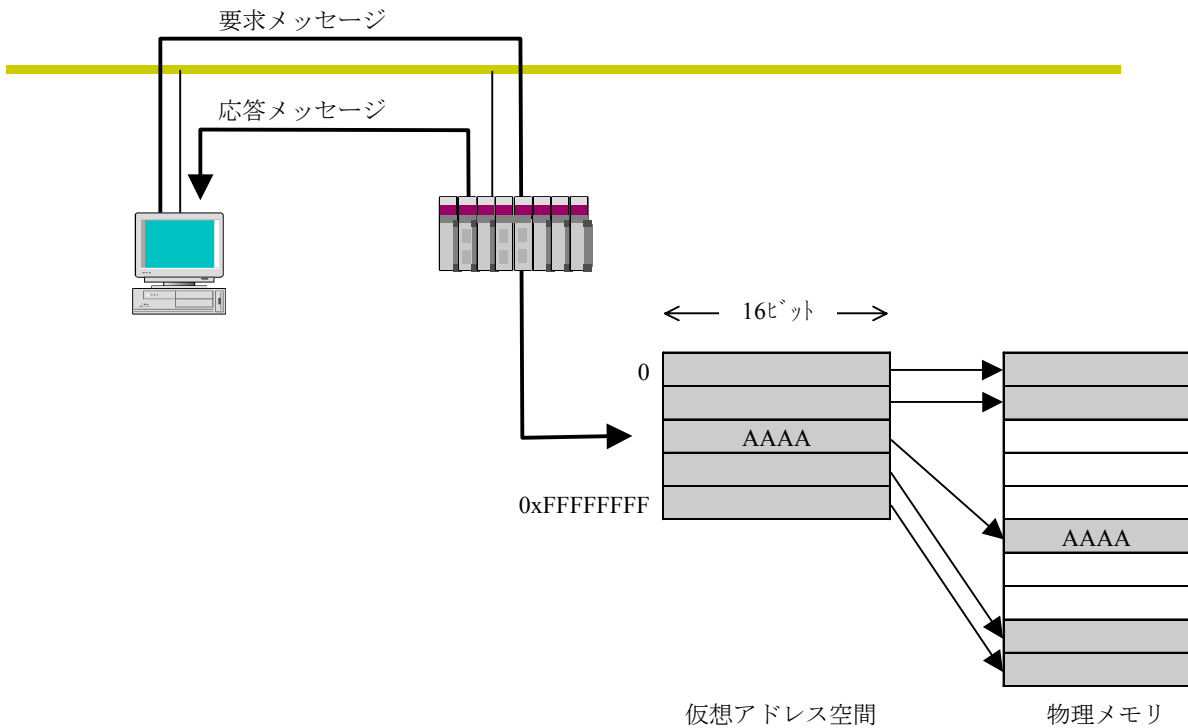


図6-29 ワードブロック書き込み

- ネットワークパラメータ読み出し
ネットワークから相手ノードのネットワークパラメータ情報を読み出す機能です。
以下の情報を読み出します。また、使用方法に関しては、「6.4 FL.NETの使い方」を参照してください。

表6-6 ネットワークパラメータ情報

ノード番号
ベンダ名
メーカー型式
ノード名 (設備名)
コモンメモリのアドレスとサイズ
トークン監視時間
リフレッシュサイクル許容時間
リフレッシュサイクル測定時間 (実測値)
最小許容フレーム間隔
上位層の状態
FL-netの状態
プロトコルのバージョン

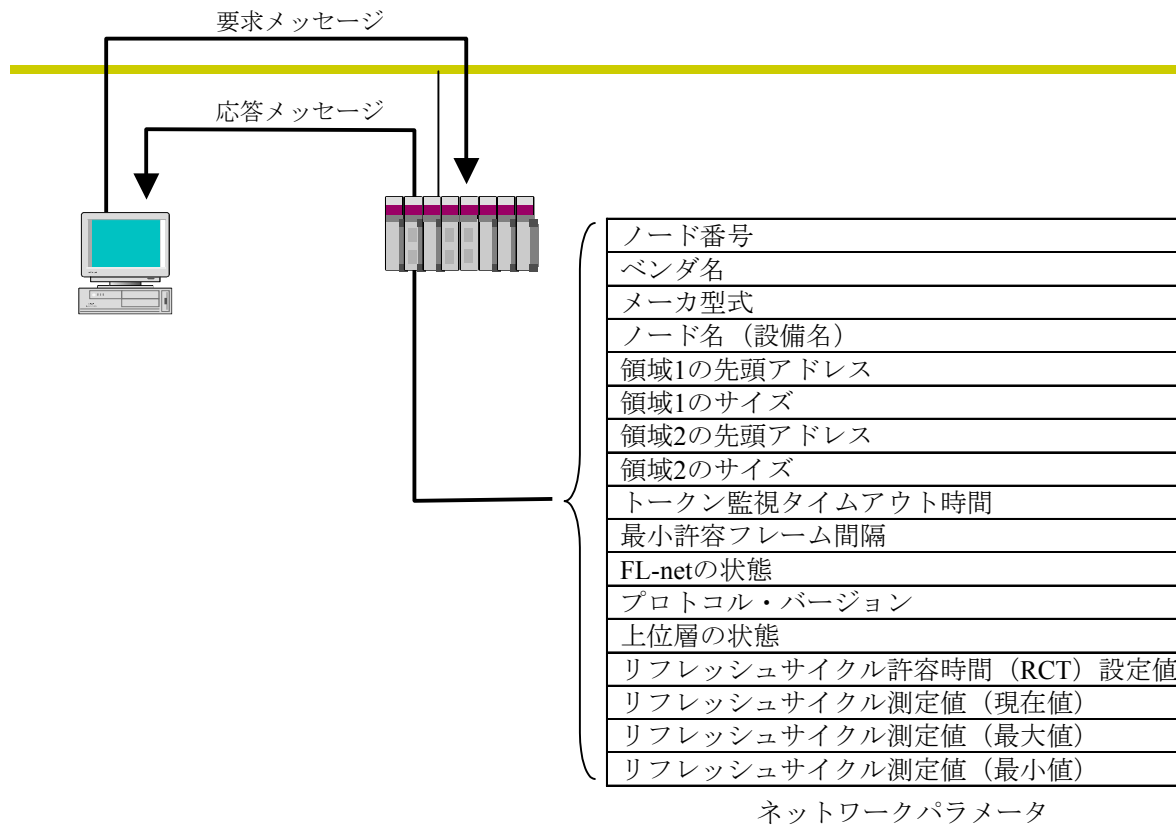


図6-30 ネットワークパラメータ読み出し

6 利用の手引き

● ネットワークパラメータ書き込み

ネットワークから相手ノードのネットワークパラメータ情報を変更する機能です。

以下の情報が変更できます。また、使用方法に関しては、「6.4 FL.NETの使い方」を参照してください。

- ・ノード名（設備名）
- ・コモンメモリのアドレスとサイズ

領域1、領域2の先頭アドレスがコモンメモリの範囲から外れる場合はエラーとなりますが、ノード名に関してはチェックしていません。

コモンメモリのアドレスとサイズを変更した場合、相手ノードはネットワークを一度離脱し、再加入します。ノード名のみ変更した場合、相手ノードは離脱しません。

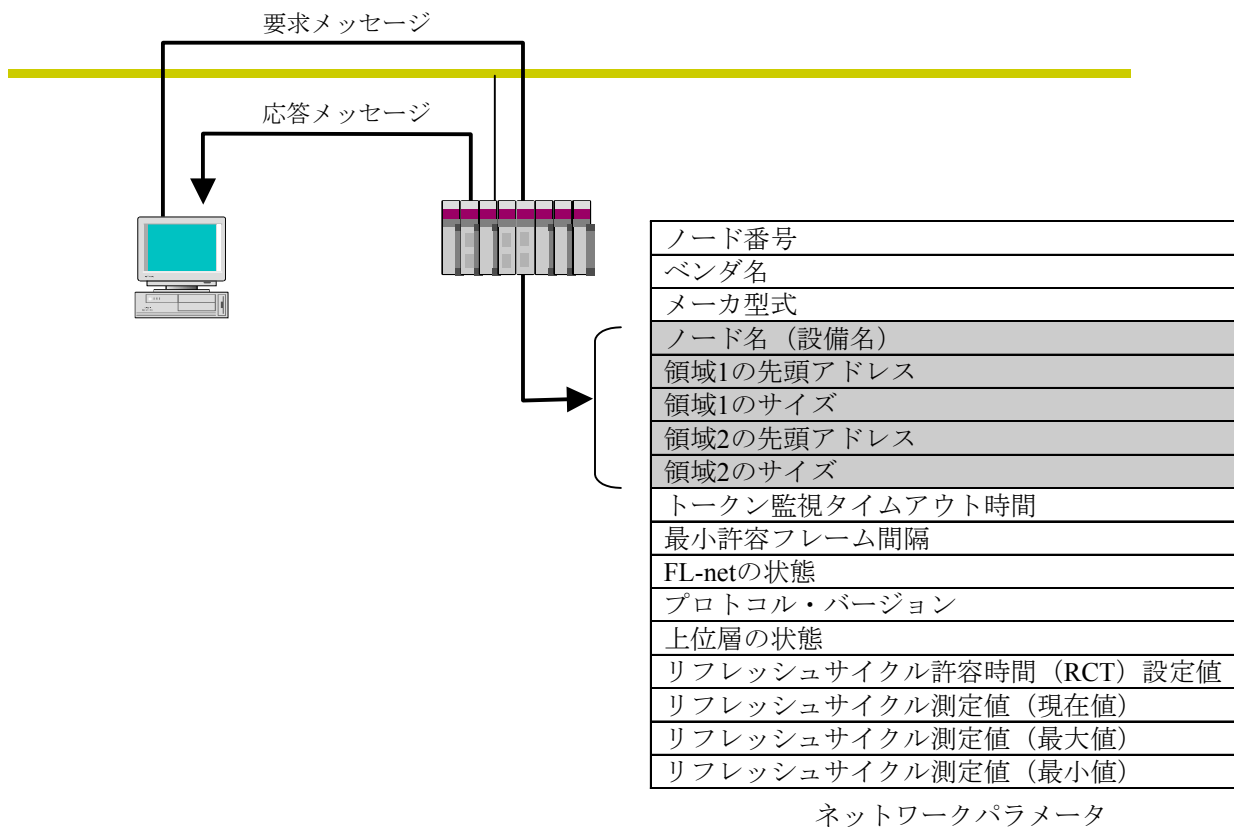


図6-31 ネットワークパラメータ書き込み

● 運転／停止指令

ネットワークからFL-netに接続されている機器の動作をリモート運転／停止させる機能です。

S10mini FL.NETでは、要求のみサポートします。使用方法に関しては、「6.4 FL.NETの使い方」を参照してください。

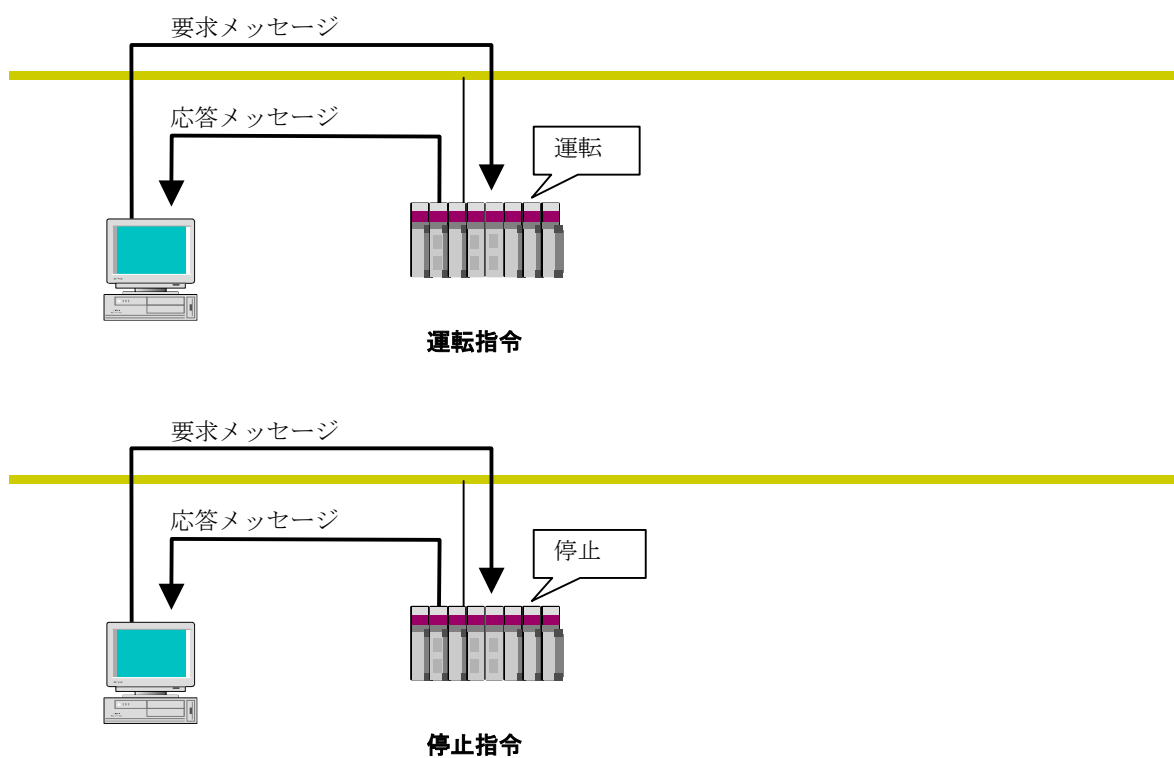


図 6 - 32 運転・停止指令

6 利用の手引き

● プロファイル読み出し

ネットワークから相手ノードの情報であるデバイスプロファイルのシステムパラメータを読み出す機能です。S10mini FL.NETでは、応答のみサポートします。

システムパラメータには、以下のパラメータ情報があります。

- ・ 共通パラメータ (必須)
- ・ デバイス固有パラメータ (任意)

システムパラメータのデバイスプロファイル詳細は、「9 付 録」を参照してください。

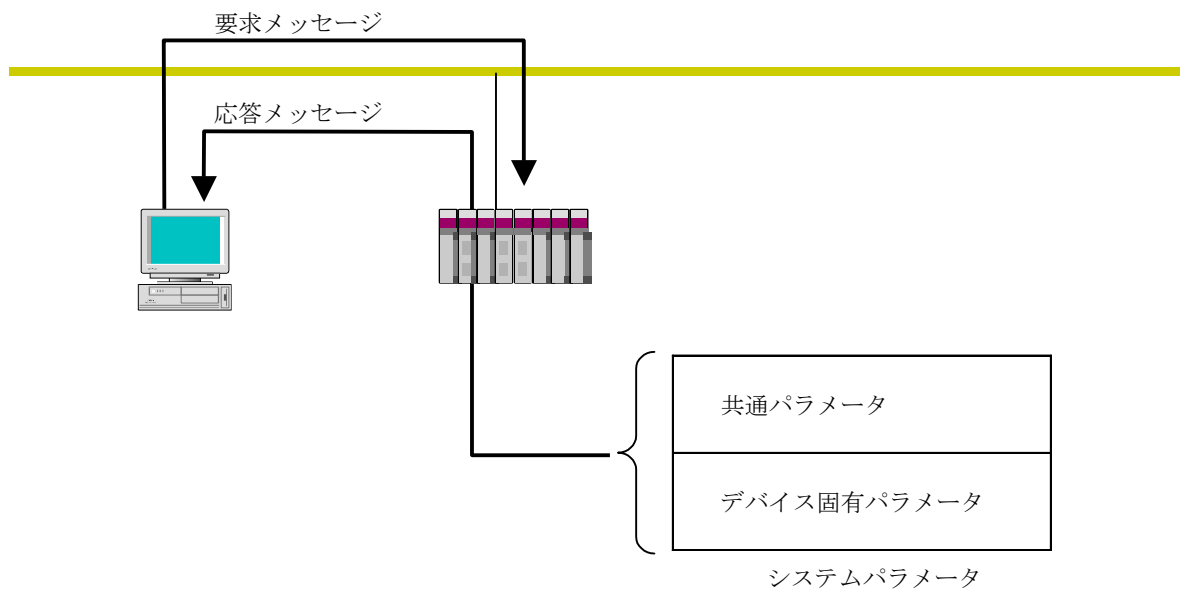


図 6-33 プロファイル読み出し

- 通信ログデータの読み出し

ネットワークから相手ノードのログ情報を読み出す機能です。

使用方法に関しては、「6.4 FL.NETの使い方」を参照してください。

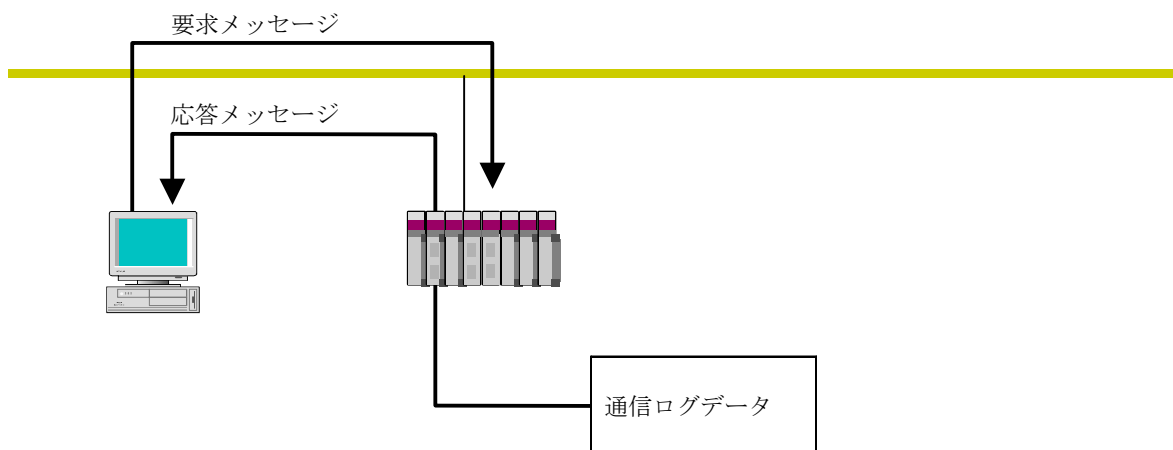


図6-34 通信ログデータの読み出し

- 通信ログデータのクリア

ネットワークから相手ノードのログ情報をクリアする機能です。

使用方法に関しては、「6.4 FL.NETの使い方」を参照してください。

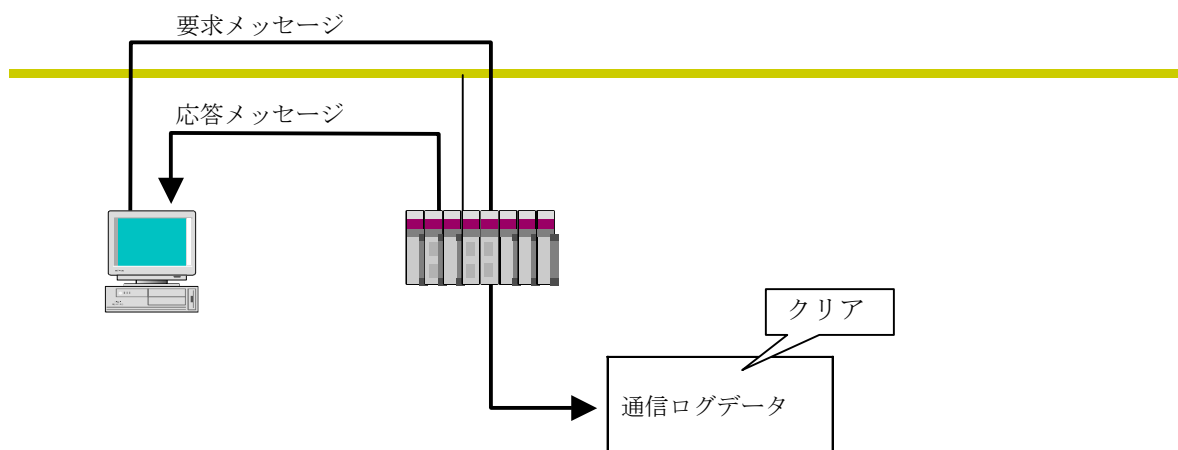


図6-35 通信ログデータのクリア

6 利用の手引き

● メッセージ折り返し

受信したメッセージを折り返す機能です。

折り返しは、FL.NET内で自動的に行います。

使用方法に関しては、「6.4 FL.NETの使い方」を参照してください。

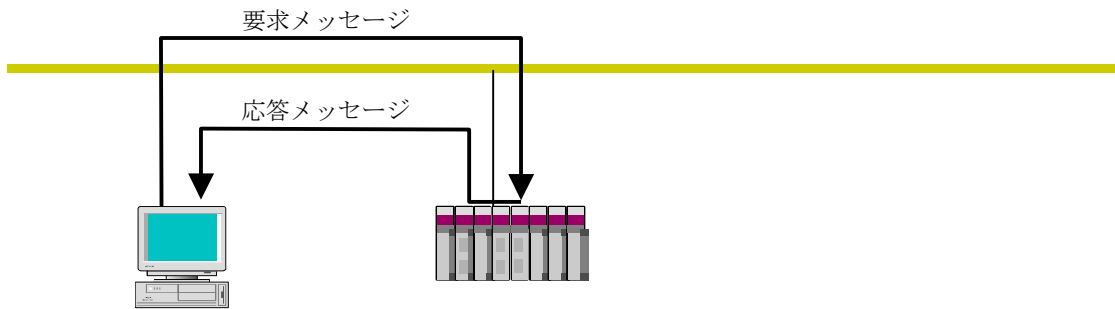


図6-36 メッセージ折り返し機能

● 透過形メッセージ伝送

FL-net上位層に透過形のサービスを提供する機能です。

このサービスは、受信メッセージをFL-net上位層へ通知します。通知を受けたFL-net上位層は、ユーザインタフェースレベルへそのまま通知します。ユーザインタフェースレベルへ通知された場合、対応する応答を作成し応答を返す必要があります。

使用方法に関しては、「6.4 FL.NETの使い方」を参照してください。

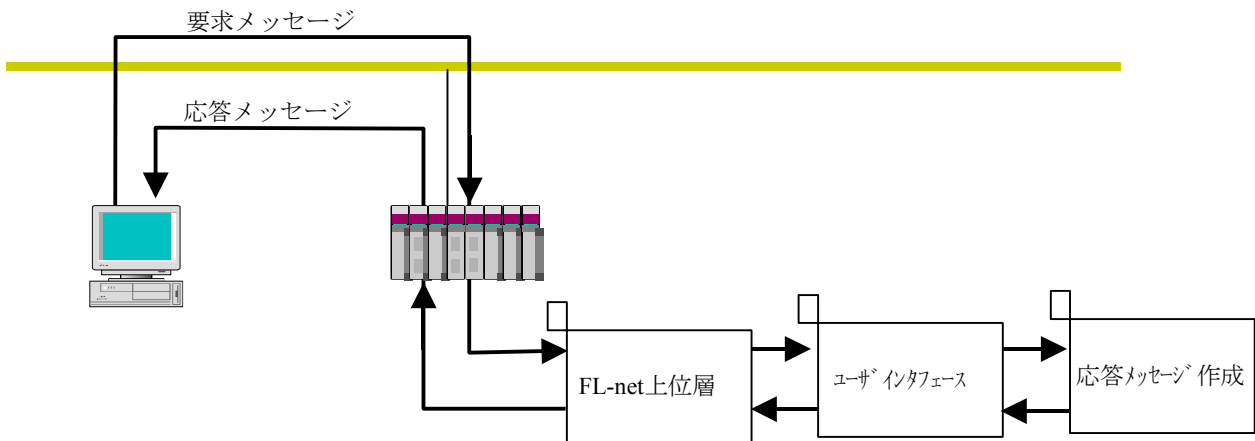
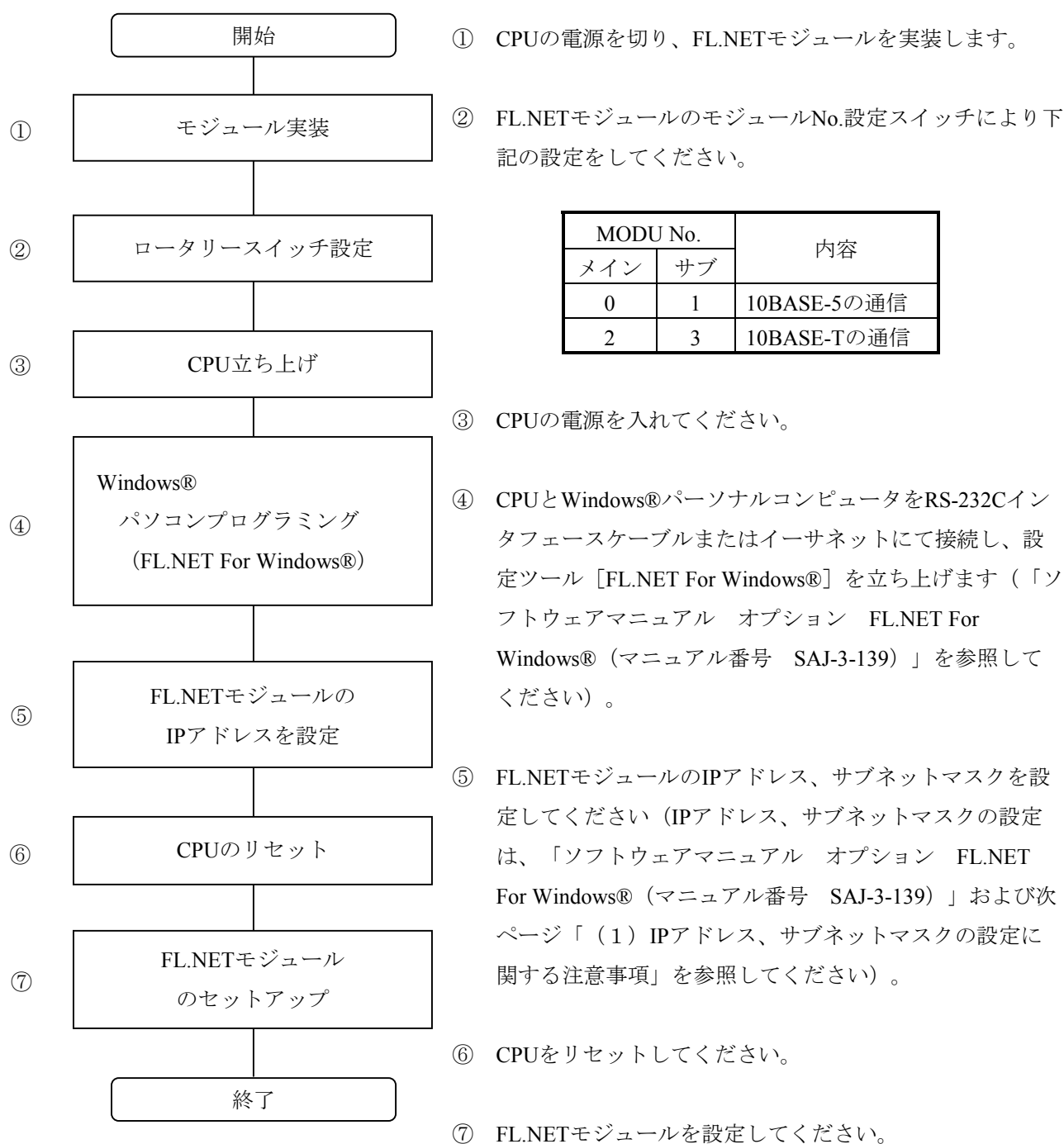


図6-37 透過形メッセージ伝送

6.3 FL.NETの設定方法

6.3.1 立ち上げ手順

このモジュールの立ち上げ手順を以下に示します。



6 利用の手引き

(1) IPアドレス、サブネットマスクの設定に関する注意事項

FL.NETモジュールでは、IPアドレスの設定、サブネットマスクの設定は自由に行えますが、FL-netプロトコルの規約に従う必要があります。

そのため、IPアドレスの設定はクラスCを使用し、ネットワークアドレスに192.168.250を使用することを推奨します。また、サブネットマスクの設定は必ず255.255.255.0としてください。

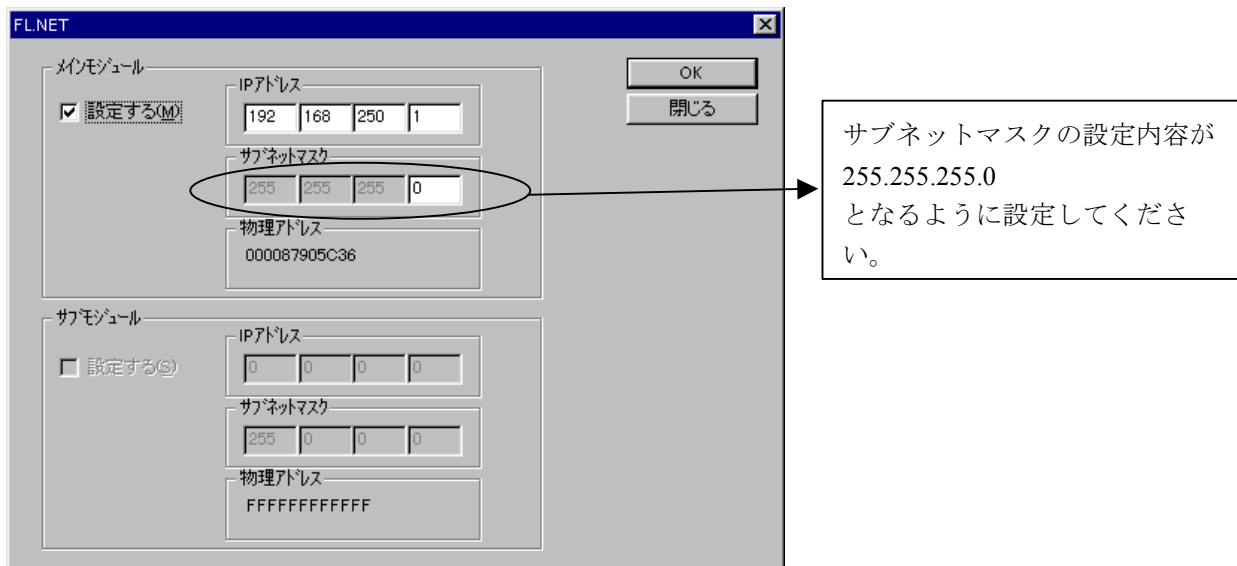


図6-38 IPアドレス設定画面

- (注1) IPアドレスの設定にて、IPアドレスをすべて/0、またはすべて/Fなどと設定した場合は、入力エラーとなります。
- (注2) ホスト番号部をすべて/0、またはすべて/Fと設定した場合、FL.NETモジュールはIPアドレス未設定と同様の処理を行います。
- (注3) FL.NETモジュール未実装状態では、設定ツール [FL.NET For Windows®] のIPアドレス設定の画面に切り替えると物理アドレス表示は/FFFFFFFFFFFFとなります。また、FL.NETモジュールが実装されていてもIPアドレス未設定状態では、物理アドレス表示は/000000000000となります。
- (注4) 物理アドレスを参照する場合は、FL.NETモジュールを実装し、IPアドレスを設定してください。なお、IPアドレス、サブネットマスクは、FL.NETモジュール未実装状態でも参照できます。
- (注5) IPアドレスが未設定またはOSロード時のメモリクリアなどによりIPアドレス設定がクリアされた場合は、FL.NETモジュールのLER LEDが点灯、CPUインディケータに下記のエラーを表示し、IPアドレス設定以外のすべての処理を停止します。
- メインモジュールのIPアドレスが未設定の場合：“FLNMIPNG”
- サブモジュールのIPアドレスが未設定の場合：“FLNSIPNG”
- (注6) 自ノード設定、他ノード受信設定を行う前に必ずFL.NETモジュールのIPアドレスを設定してください。FL.NETモジュールのIPアドレスが未設定のまま自ノード設定、他ノード受信設定を行ってもタイムアウトエラーが発生し設定が正常終了しません。

6.3.2 モジュールの実装とスイッチの設定

このモジュールの実装方法およびスイッチの設定に関しては、「4 FL.NETモジュールの実装」を参照してください。

6.3.3 ツールの接続方法

ツールの接続方法に関しては、「ソフトウェアマニュアル オプション FL.NET For Windows® (マニュアル番号 SAJ-3-139)」を参照してください。

6.3.4 ツールの立ち上げ方法

ツールの立ち上げ方法に関しては、「ソフトウェアマニュアル オプション FL.NET For Windows® (マニュアル番号 SAJ-3-139)」を参照してください。

6 利用の手引き

6.4 FL.NETの使い方

6.4.1 リンクパラメータの設定方法

リンクパラメータとは、FL.NETモジュールのコモンメモリ領域を各ノードごとに設定するパラメータ値です。設定ツール [FL.NET For Windows®] を使用してリンクパラメータの設定を行います。下記に自ノード設定画面を表示します。

「現在値」にFL.NETモジュールの現在の設定値が表示されます。

設定を変更したい場合には、「書き換え値」に変更したい内容を入力し、[OK] ボタンをクリックすれば、FL.NETモジュール内にリンクパラメータ設定が保存されます。



図 6-39 リンクパラメータ設定画面

以下に、各設定項目の入力可能範囲を示します。

入力項目	内 容	指定可能範囲値
ノード番号	FL-net通信を行うにあたり各ノードを識別するための番号です。	1～254
トークン監視 タイムアウト時間	自ノード宛のトークンを保持可能な時間です (ms)。	255固定 (ユーザは設定できません。)
最小許容フレーム 間隔	フレームを連続送信するときの間隔、および自ノード宛トークンを受信してからフレームを送信するまでの間隔です (100 μ s単位)。	0～50
領域1アドレス	領域1用送信エリアの開始アドレスを指定します。16進数で入力してください。	0～0x1FF (511)
領域1ワード数	領域1用送信エリアのサイズを指定します。16進数で入力してください。	0～0x200 (512)
領域2アドレス	領域2用送信エリアの開始アドレスを指定します。16進数で入力してください。	0～0x1FFF (8191)
領域2ワード数	領域2用送信エリアのサイズを指定します。16進数で入力してください。	0～0x2000 (8192)
ノード名称	自ノード名を指定します。	半角英数字10文字 以内

詳しい操作方法については、「ソフトウェアマニュアル オプション FL.NET For Windows® (マニュアル番号 SAJ-3-139)」を参照してください。

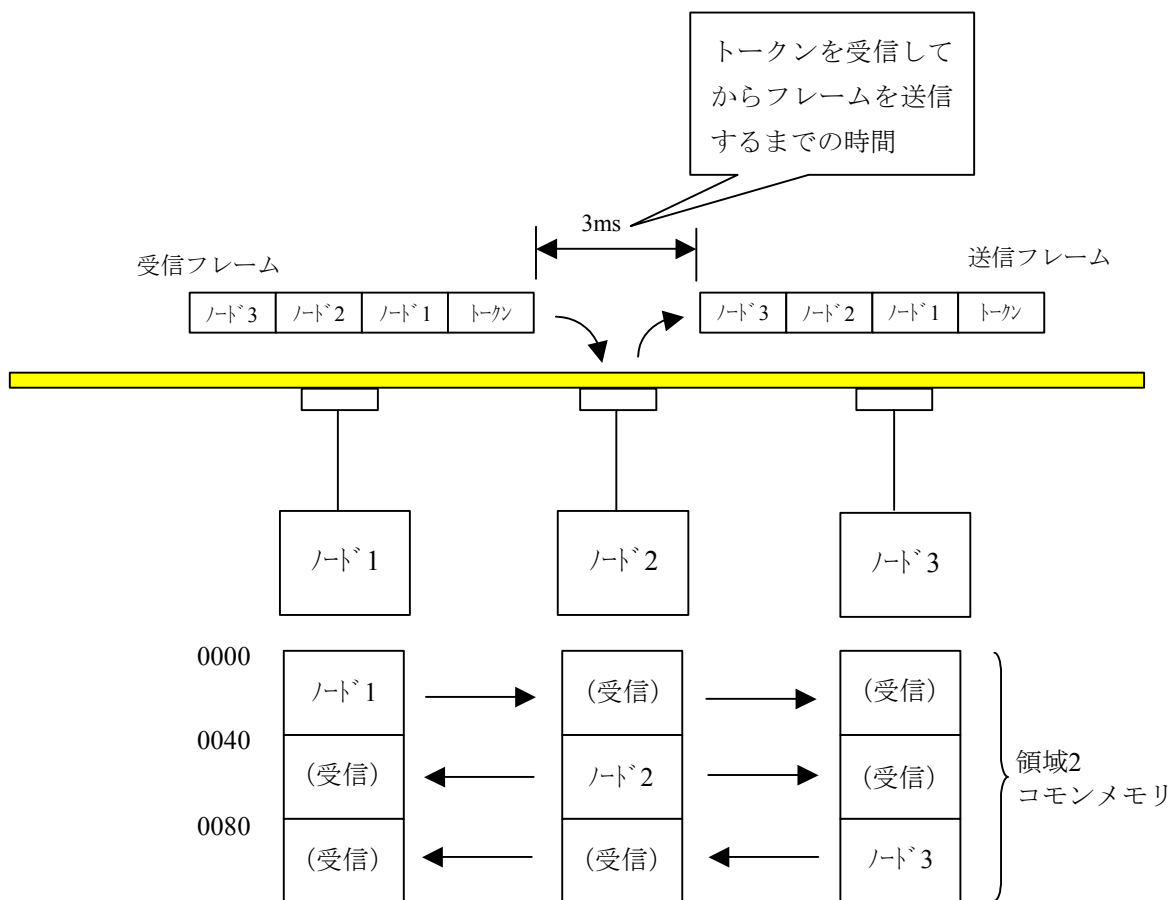
- (注1) 最小許容フレーム間隔に「0」を指定した場合には、フレームの送信間隔はありません。
- (注2) 自ノードからコモンメモリ送信をしない場合には、開始アドレス、サイズとも「0」を指定してください。
- (注3) 最小許容フレーム間隔の値は、20以上 (2ms以上) としてください。最小許容フレーム間隔の値が小さいと他ノードのと接続ができない可能性があります。

6 利用の手引き

(例) リンクパラメータ値を下記としたときのFL.NETモジュール動作を示します。

ノード番号	: 1	ノード番号	: 2
領域1アドレス	: 0x000	領域1アドレス	: 0x000
領域1ワード数	: 0x000	領域1ワード数	: 0x000
領域2アドレス	: 0x0000	領域2アドレス	: 0x0040
領域2ワード数	: 0x0040	領域2ワード数	: 0x0040
最小許容フレーム間隔	: 30	最小許容フレーム間隔	: 30
ノード名称	: CPU01	ノード名称	: CPU02

ノード番号	: 3
領域1アドレス	: 0x000
領域1ワード数	: 0x000
領域2アドレス	: 0x0080
領域2ワード数	: 0x0040
最小許容フレーム間隔	: 30
ノード名称	: CPU03



領域2コモンメモリのエリアを上記のようにしてデータの送受信を行います。

6. 4. 2 CPUメモリの割り付け方法

FL.NETモジュールでは、コモンメモリ領域をCPUメモリに割り付けます。

FL.NETモジュールで受信したデータは、割り付けしたCPUメモリに転写します。

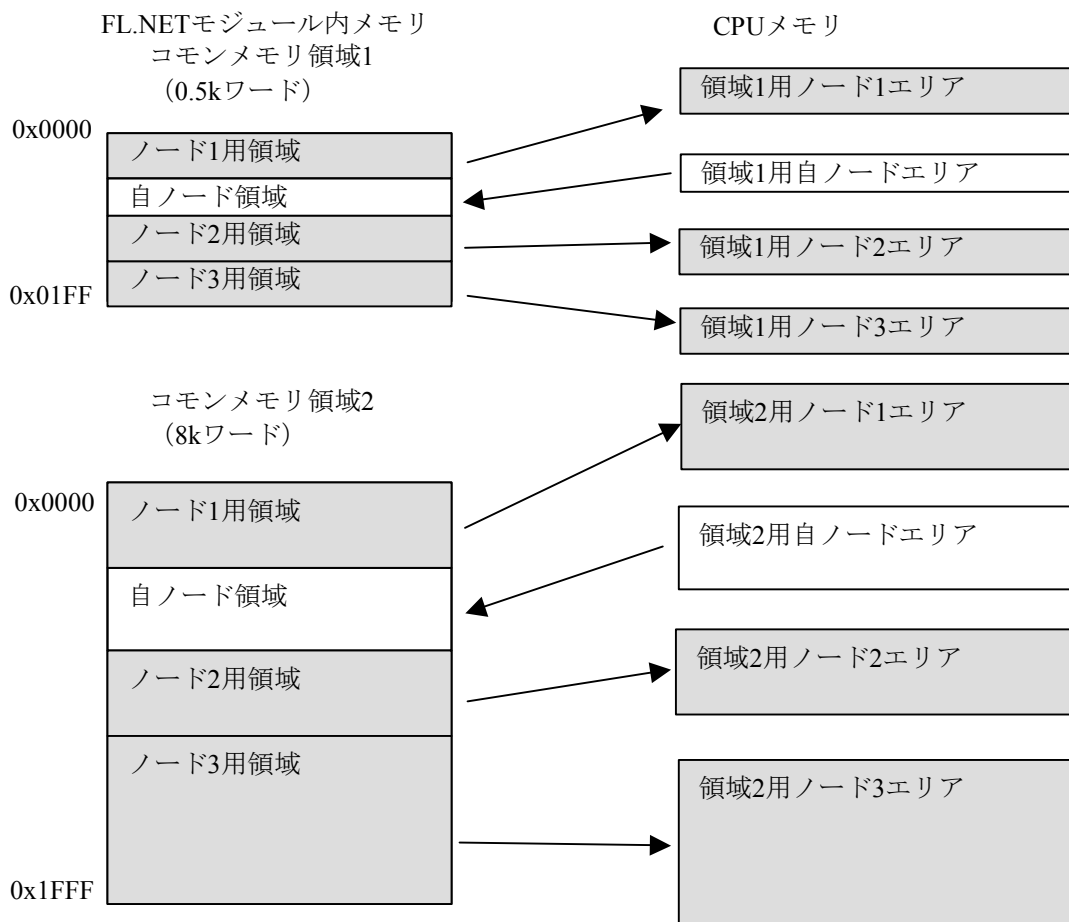


図6-40 FL.NETモジュールのコモンメモリデータ転写イメージ

コモンメモリ領域をCPUメモリへ割り付ける設定は、設定ツール [FL.NET For Windows®] にて行います。

各設定画面での詳しい操作方法は、「ソフトウェアマニュアル オプション FL.NET For Windows® (マニュアル番号 SAJ-3-139)」を参照してください。

6 利用の手引き

また、コモンメモリ領域をCPUメモリに割り付けできるアドレスを表6-7に示します。

表6-7 コモンメモリ領域をCPUメモリに割り付けできるアドレス範囲

指定可能レジスタ/アドレス	備 考
XW000~XWFF0	外部入力
YW000~YWFF0	外部出力
JW000~JWFF0	トランスファレジスタ
QW000~QWFF0	レシーブレジスタ
GW000~GWFF0	グローバルリンクレジスタ
RW000~RWFF0	内部レジスタ
EW400~EWFF0	イベントレジスタ
MW000~MWFF0	内部レジスタ
DW000~DWFFF	ファンクションデータレジスタ
FW000~FWBFF	ファンクションワークレジスタ
/100000~/4FFFFE (拡張メモリ)	CPUモジュールに拡張メモリが実装されている必要があります。

(1) 自ノード用CPUメモリの割り付け

設定ツール [FL.NET For Windows®] を使用して、自ノード設定画面を表示します。

PCs割り付けに表示されているエリアに、自ノードCPUメモリ領域として割り付けたいPI/Oまたは拡張メモリアドレスを入力してください。入力後、[OK] ボタンをクリックすれば、FL.NETモジュール内に設定が保存されます。

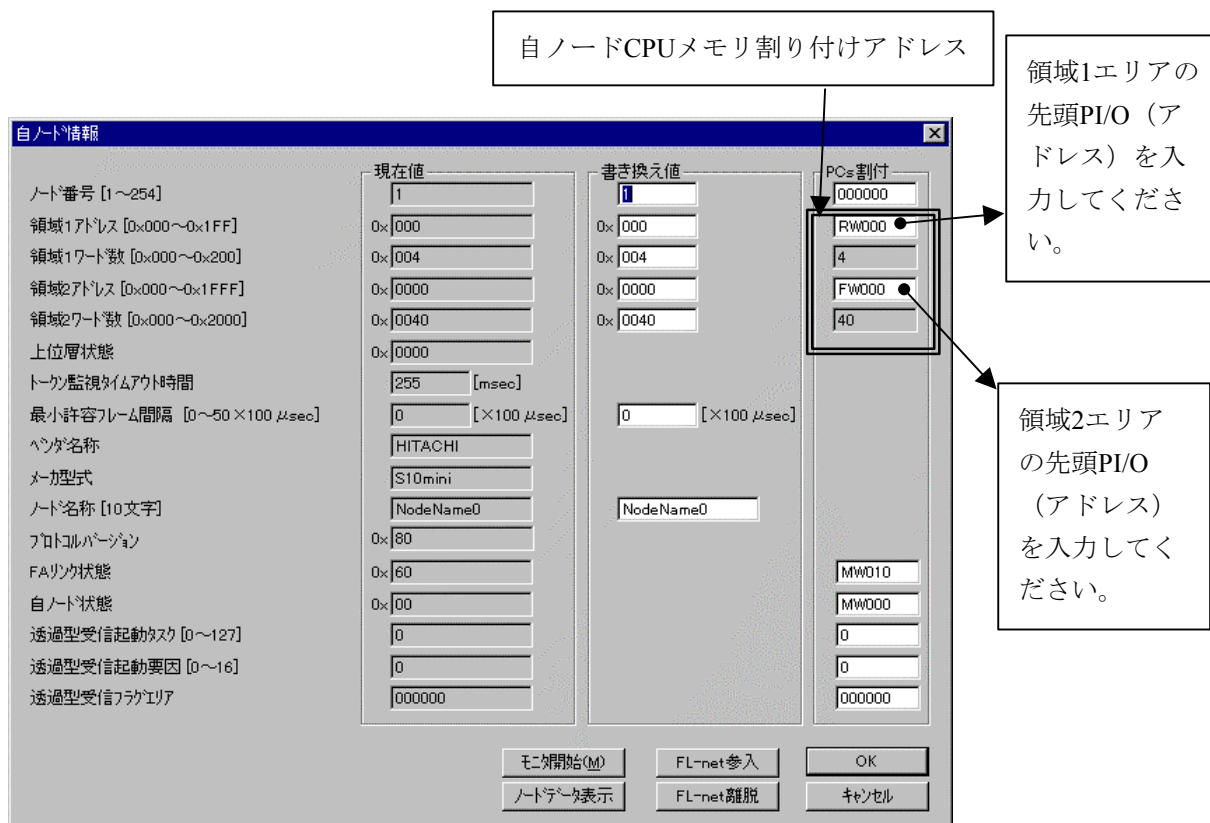


図 6-41 自ノードのCPUメモリ割り付け画面

入力項目	内 容	指定可能範囲
領域1アドレス (PCs割り付け)	領域1のコモンメモリとCPUメモリを割り付けする開始アドレスを設定します。	「表 6-7 コモンメモリ領域をCPUメモリに割り付けできるアドレス範囲」を参照してください。
領域2アドレス (PCs割り付け)	領域2のコモンメモリとCPUメモリを割り付けする開始アドレスを設定します。	「表 6-7 コモンメモリ領域をCPUメモリに割り付けできるアドレス範囲」を参照してください。

6 利用の手引き

- (注1) PCs割り付けに使用する領域1ワード数（領域2ワード数）は、リンクパラメータの領域1ワード数（領域2ワード数）と同じになります。
- (注2) 自ノード用CPUメモリの割り付けアドレスの設定は、各ノード用CPUメモリの割り付けアドレスと重複しないように設定してください。
- (注3) CPUメモリの割り付けは、PI/O範囲をまたがって使用できません。
例：以下のような設定はできません。
領域1側の設定でワード数（サイズ）が3ワードとなっているとき、CPUメモリの割り付けに“RWFF0”を指定できません（RWエリアの範囲を超えています）。
- (注4) リンクパラメータの領域1ワード数（領域2ワード数）を“0”に設定した場合、PCs割り付けの領域1アドレス（領域2アドレス）は設定できません。

(2) 他ノード用CPUメモリ領域の設定

設定ツール [FL.NET For Windows®] を使用して、他ノード受信設定をします。

各ノードから受信したデータをCPUメモリに割り付けしたエリアに転送します。CPUメモリの領域ワード数と、コモンメモリ設定サイズ（リンクパラメータの領域ワード数）が異なる場合、FL.NETモジュールは、CPUメモリの領域ワード数で指定の語数分のデータをコモンメモリから転写します。コモンメモリに受信したデータをすべて取得したい場合には、各ノードのコモンメモリ設定（リンクパラメータの設定値）と同じサイズを設定してください。

自ノード番号のCPUメモリ割り付け領域を表示しています。

領域1エリアの先頭PI/O（アドレス）、サイズを入力してください。

領域2エリアの先頭PI/O（アドレス）、サイズを入力してください。

参加他ノードCPUメモリ領域設定

ノード番号	領域1アドレス (16進数)	領域1ワード数 (16進数)	領域2アドレス (16進数)	領域2ワード数 (16進数)	FAリンク状態	上位層状態
1	RW000	4	FW000	40	MW010	000000
2	000000	0	000000	0	000000	000000
3	RW330	4	FW330	40	MW030	000000
4	RW440	4	FW440	40	MW040	000000
5	000000	0	000000	0	000000	000000
6	000000	0	000000	0	000000	000000
7	000000	0	000000	0	000000	000000
8	000000	0	000000	0	000000	000000
9	000000	0	000000	0	000000	000000
10	000000	0	000000	0	000000	000000
11	000000	0	000000	0	000000	000000
12	000000	0	000000	0	000000	000000
13	000000	0	000000	0	000000	000000
14	000000	0	000000	0	000000	000000
15	000000	0	000000	0	000000	000000
16	000000	0	000000	0	000000	000000

図 6-42 参加他ノード用コモンメモリ領域設定画面

6 利用の手引き

以下に、各項目での入力可能範囲を示します。

入力項目	内 容	指定可能範囲
領域1アドレス	指定ノード（左端に記されたノード番号）用領域1受信エリアの開始アドレスを指定します。	「表6-7 コモンメモリ領域をCPUメモリに割り付けできるアドレス範囲」を参照してください。
領域1ワード数	指定ノード（左端に記されたノード番号）用領域1受信エリアのサイズ（ワード単位）を指定します。16進数で入力してください。	0~0x200 (512)
領域2アドレス	指定ノード（左端に記されたノード番号）用領域2受信エリアの開始アドレスを指定します。	「表6-7 コモンメモリ領域をCPUメモリに割り付けできるアドレス範囲」を参照してください。
領域2ワード数	指定ノード（左端に記されたノード番号）用領域2受信エリアのサイズ（ワード単位）を指定します。16進数で入力してください。	0~0x2000 (8192)

(注1) 他ノード受信設定では、自ノードのCPUメモリ領域に割り付けしたエリアと重複しないように設定してください。

(注2) CPUメモリの割り付けは、各PI/O範囲をまたがって使用できません。

例：以下のような設定はできません。

領域1ワード数（サイズ）が3ワードとなっているとき、領域1アドレスに“RWFF0”を指定できません（RWエリアの範囲を超えています）。

- CPUメモリの領域ワード数と、コモンメモリ設定サイズ（リンクパラメータの領域ワード数）が異なる場合、FL.NETモジュールは、CPUメモリの領域ワード数で指定の語数分だけコモンメモリからデータを転写します。
- 設定されていないノードがあった場合、FL.NETモジュールは、そのノードからのコモンメモリに受信したデータをS10mini上に転写できません。

(3) 自ノード状態、FAリンクステータスを参照したい場合

FL.NETモジュールの自ノード状態/FAリンク状態、各ノードのFAリンクステータス、上位層状態フラグは、設定ツール [FL.NET For Windows®] にて参照できます。

S10mini上で各状態を参照したい場合、各フラグの転写エリアを設定することで、参加ノードの状態などを把握できるようになります。

自ノードの自ノード状態/FAリンク状態転写エリアは、自ノード情報画面にて設定します。

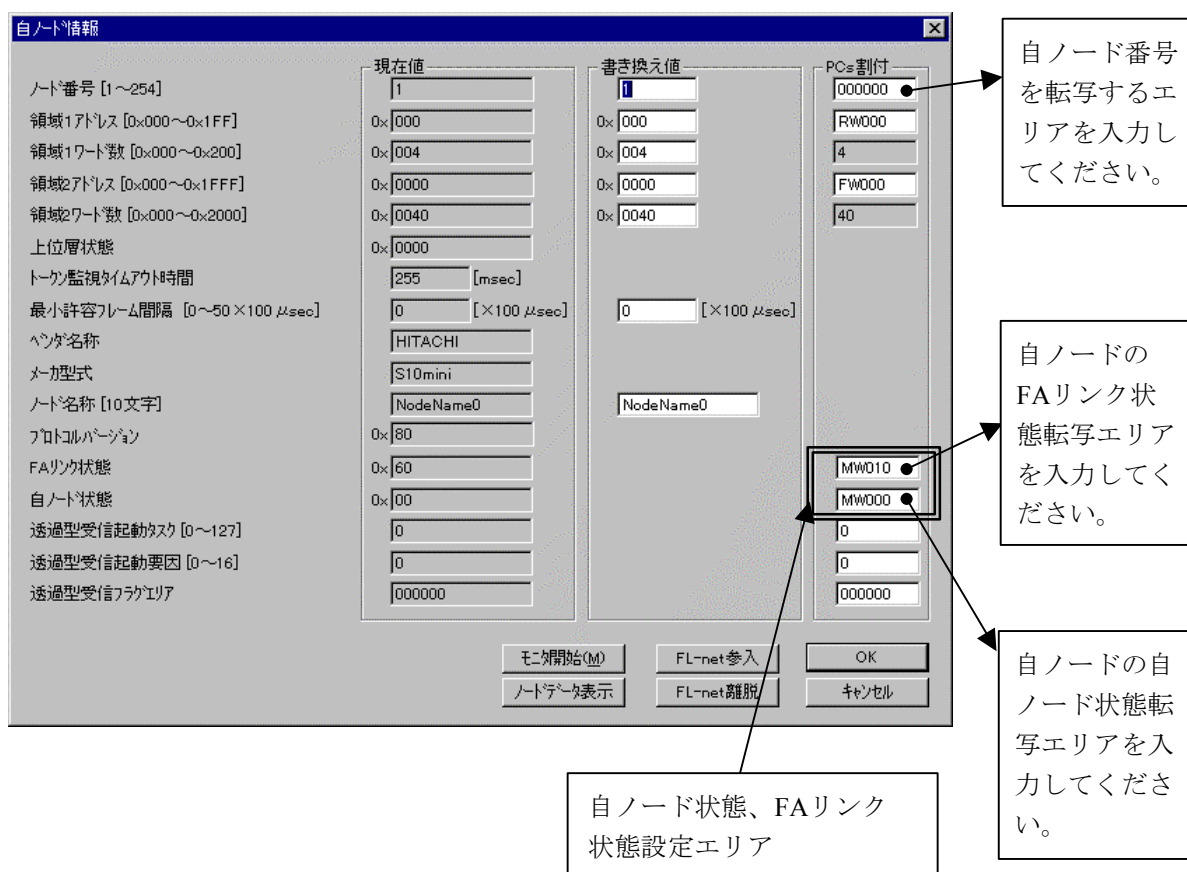


図 6-43 自ノードの各状態フラグエリア設定画面

自ノード状態、FAリンク状態の表示内容（ビット割り付け）に関しては、「6.4.6 管理テーブルの使い方」を参照してください。

6 利用の手引き

以下に、各項目での入力可能範囲を示します。

入力項目	内 容	指定可能範囲
ノード番号 (PCs割り付け)	自ノード番号を転写するエリアを指定します(エリアサイズは1ワードです)。	「表6-7 コモンメモリ領域をCPUメモリに割り付けできるアドレス範囲」を参照してください。
FAリンク状態 (PCs割り付け)	自ノードのFAリンク状態を転写するエリアを指定します(エリアサイズは1ワードです)。	「表6-7 コモンメモリ領域をCPUメモリに割り付けできるアドレス範囲」を参照してください。
自ノード状態 (PCs割り付け)	自ノード状態を転写するエリアを指定します(エリアサイズは1ワードです)。	「表6-7 コモンメモリ領域をCPUメモリに割り付けできるアドレス範囲」を参照してください。

(注1) 自ノード状態エリア、FAリンク状態エリアに指定できるレジスタ/拡張メモリアドレスの範囲は、CPUメモリ割り付け領域と同じです。

(注2) 自ノード状態エリア/FAリンク状態エリアは、参加他ノード用CPUメモリ割り付け領域の設定、自ノード用CPUメモリ割り付け領域と重複しないように設定してください。

- 自ノード状態エリアを設定すると、自ノード状態フラグが、1ワードのデータとして転写されます。
- FAリンク状態エリアを設定すると、指定エリアの下位バイトに、自ノードのFAリンク状態フラグが転写されます。
- 自ノード状態エリア、FAリンク状態エリアを指定されない場合、FL.NETモジュールは、これらの状態をS10mini上に転写できません。

各ノードのFAリンク状態／上位層状態フラグエリアは、参加他ノード設定画面にて設定します。

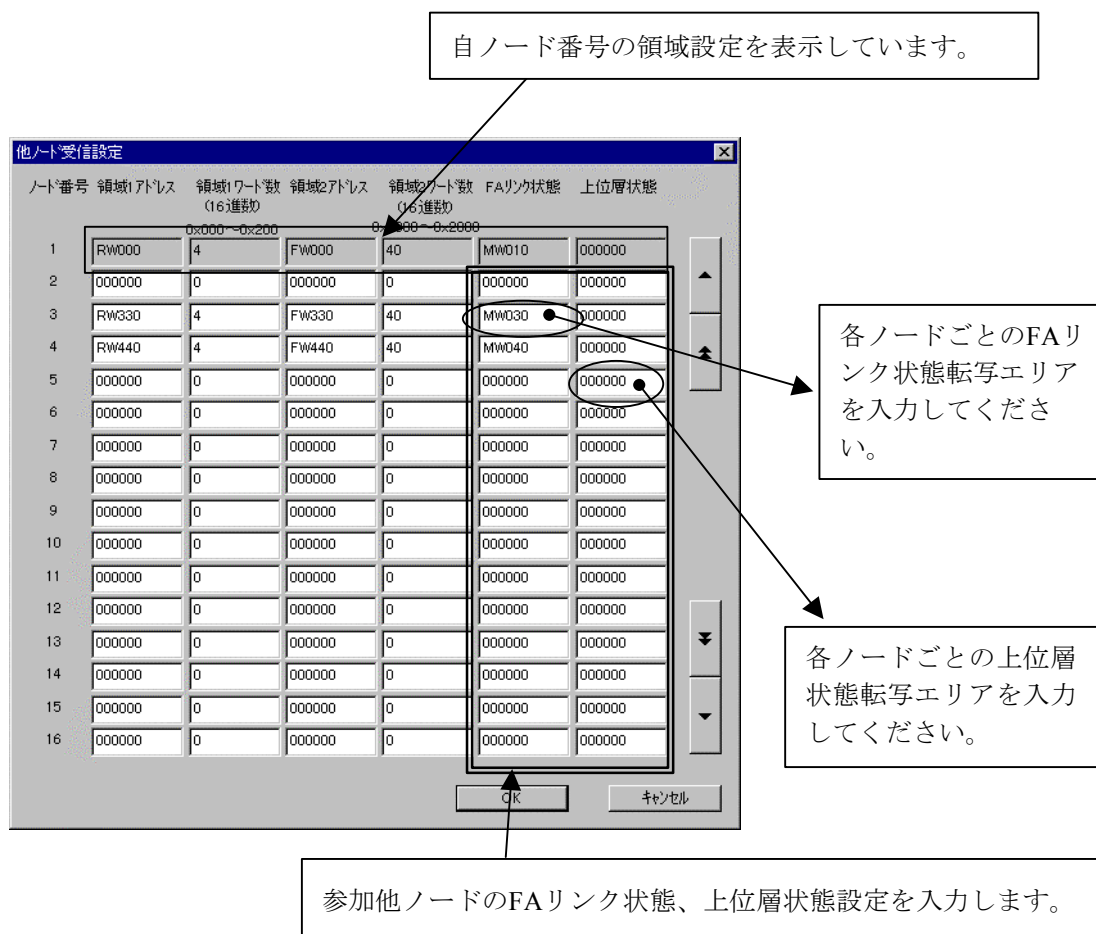


図6-44 参加他ノードの各状態フラグエリア設定画面

以下に、各項目での入力可能範囲を示します。

入力項目	内 容	指定可能範囲
FAリンク状態	左端に記されたノード番号のFAリンク状態を転写するエリアを指定します（エリアサイズは1ワードです）。	「表6-7 コモンメモリ領域をCPUメモリに割り付けできるアドレス範囲」を参照してください。
上位層状態	左端に記されたノード番号の上位層状態を転写するエリアを指定します（エリアサイズは1ワードです）。	「表6-7 コモンメモリ領域をCPUメモリに割り付けできるアドレス範囲」を参照してください。

FAリンク状態、上位層状態の表示内容（ビット割り付け）に関しては、「6.4.6 管理テーブルの使い方」を参照してください。

6 利用の手引き

- 上位層状態エリア、FAリンク状態エリアに指定可能なレジスタ／拡張メモリアドレスの範囲は、CPUメモリ割り付け領域と同じです。
- 上位層状態エリアを設定すると、上位層状態フラグの内容が、1ワードのデータとして転写されます。
- FAリンク状態エリアを設定すると、指定エリアの下位バイトに、そのノードのFAリンク状態フラグの内容が転写されます。
- 上位層状態エリア／FAリンク状態エリアは、他ノード受信設定でのCPUメモリ割り付け領域の設定、自ノード情報でのCPUメモリ割り付け領域の設定、その他のエリアと重複しないように設定してください。
- 設定されていないノードがあった場合、FL.NETモジュールは、そのノードの上位層状態、FAリンクステータスをS10mini上に転写できません。

6. 4. 3 ビットデータの使い方

FL.NETモジュールでは、コモンメモリ領域1のデータを送信／受信時にビット変換します。

例：領域1の1ワードデータ

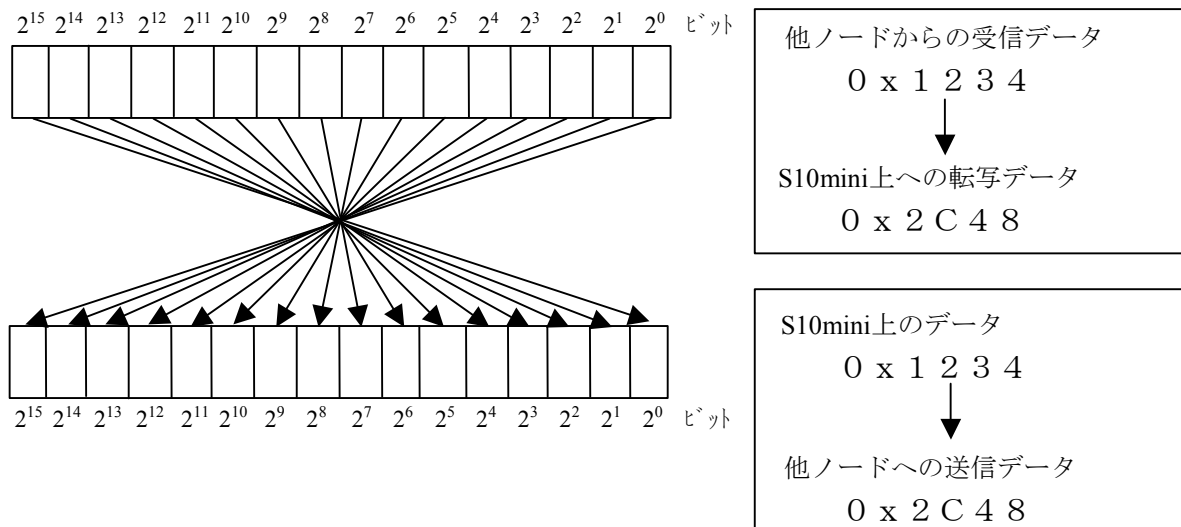


図 6-45 領域1データのビット変換

6. 4. 4 ワードデータの使い方

FL.NETモジュールでは、コモンメモリ領域2のデータを送信／受信時には、ビット変換をしません。

S10mini上のデータの並びそのままにネットワーク上に転送し、他ノードから受信したデータをそのまま転写します。

6. 4. 5 メッセージ通信の使い方

(1) メッセージ通信 (サーバ側)

FL.NETモジュールは、他ノードからのメッセージ要求を受信した場合、そのメッセージをFL.NETモジュール内部で処理します。ユーザが意識する必要はありません。

ただし、透過型メッセージを受信した場合には、自ノード状態フラグ、透過型受信フラグを通じて、ユーザに透過型メッセージを受信したことを通知します。詳しくは、「(4) 透過型メッセージの受信について」を参照してください。

また、透過型メッセージを受信した場合には、Cモードプログラムまたは演算ファンクションを使用して透過型メッセージをFL.NETモジュールから取り込んでください。

透過型メッセージがFL.NETモジュール内にたまっていると、他の要求メッセージを処理できなくなる場合があります。

(2) メッセージ通信 (クライアント側)

他ノードに対するメッセージを要求する (または透過型メッセージを受信する) 場合には、Cモードプログラムや演算ファンクションを使用して、FL.NETモジュールに対して要求を発行してください。

FL.NETモジュールでは、メッセージ要求を発行する手段として、Cモードハンドラと演算ファンクションを用意しています。

以下に、Cモードプログラムやラダープログラムで、メッセージ要求を発行する手順を示します。

(注) メッセージ通信を行うには、ユーザプログラムにて自ノード状態フラグが参照できる環境が必要です。

設定ツール [FL.NET For Windows®] の自ノード設定画面にて、自ノード状態フラグエリアを設定してください。詳しい操作方法に関しては、「ソフトウェアマニュアル オプション FL.NET For Windows® (マニュアル番号 SAJ-3-139)」を参照してください。

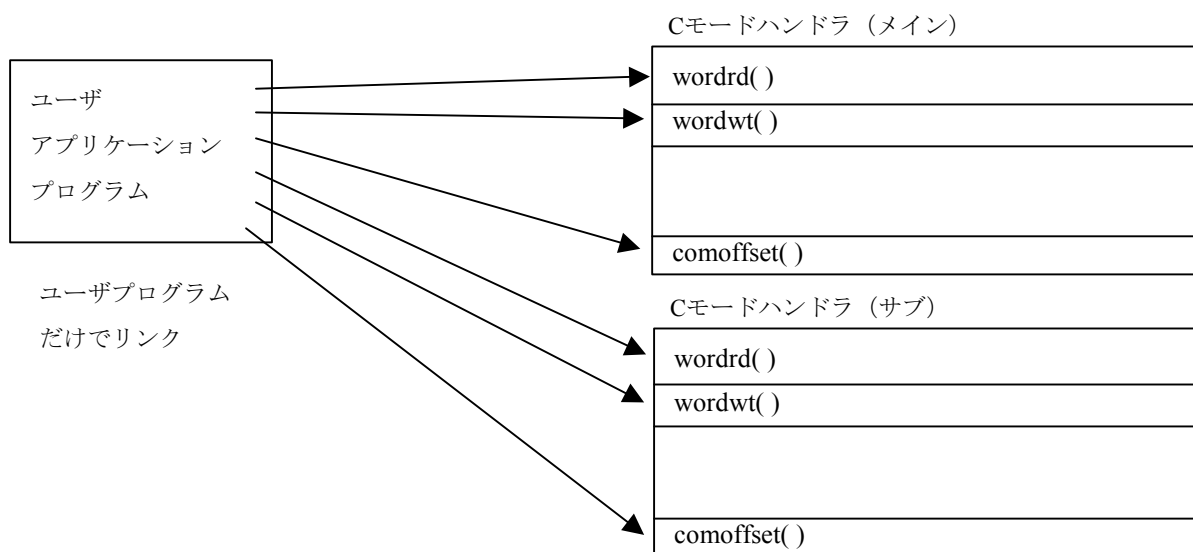
6 利用の手引き

(3) Cモードハンドラ、演算ファンクション

(a) Cモードハンドラによるメッセージ送信要求

CモードハンドラはC言語の関数として呼び出され、ユーザプログラムの代わりにFL.NETモジュールに対するメッセージ要求を行い、データの送受信を実行します。Cモードハンドラはメッセージの種類別に用意されています。

Cモードハンドラは、アドレス指定で呼び出してください。ユーザプログラムは、Cモードハンドラを含めた形では作成（リンク）できません。



また、FL.NETモジュールのCモードハンドラは、FL.NETモジュールに対して要求を発行するだけで処理を終了します。

Cモードハンドラからのリターンコードが `-1 (0xFFFFFFFF)` の場合には、パラメータ異常、または他メッセージ処理中であることを意味します。Cモードハンドラ起動時に指定したエラーコード格納アドレスの内容をチェックしてください。リターンコードが0の場合には、Cモードハンドラは処理の発行情が正常であることを意味します。

FL.NETモジュールは、要求受け付け後に、自ノード状態フラグ内のユーザ要求処理中ビットをONし、処理が終了するとユーザ要求処理中ビットをOFFします。

発行後、メッセージ処理終了を確認するには、自ノード状態フラグをチェックしてください。状態で処理終了を表します。

また、Cモードハンドラでの要求発行後は、ユーザ要求処理中ビットがOFFするまでメッセージ要求しないでください。要求を発行した場合、そのメッセージ要求は他メッセージ処理中として処理されません（エラーコード格納アドレス内に他メッセージ処理中エラーコードが設定されます）。

ユーザ要求処理中ビットがOFFした後、Cモードハンドラ起動時に指定したエラーコード格納アドレスに処理終了後の状態を入力します。

要求した処理が正常終了したかは、そのエラーコード格納アドレス内に設定されたコードにて確認してください。

表 6-8 Cモードハンドラー一覧

名称	サブルーチンコールアドレス		機能
	メイン	サブ	
wordrd()	/D74112	/DF4112	ワードブロック読み出し要求発行
wordwt()	/D74118	/DF4118	ワードブロック書き込み要求発行
parard()	/D7411E	/DF411E	ネットワークパラメータ読み出し要求発行
parawt()	/D74124	/DF4124	ネットワークパラメータ書き込み要求発行
reqstop()	/D7412A	/DF412A	停止要求発行
reqrun()	/D74130	/DF4130	運転要求発行
logrd()	/D7413C	/DF413C	ログ読み出し要求発行
logclr()	/D74142	/DF4142	ログクリア要求発行
mesret()	/D74148	/DF4148	メッセージ折り返し要求発行
reqmacro()	/D74160	/DF4160	指定タスク制御要求 (日立独自透過型サポート)
toukaread()	/D74178	/DF4178	透過型メッセージ受信要求 (日立独自透過型サポート)
toukasend()	/D7417E	/DF417E	透過型メッセージ送信要求 (日立独自透過型サポート)
comoffset()	/D74184	/DF4184	コモンメモリオフセット転写要求

(注) 自ノード状態フラグのビット割り付けに関しては、「6.4.6 管理テーブルの使い方」を参照してください。

Cモードハンドラと演算ファンクションは混在して使用しないでください。
必ずFL.NETモジュール単位で、Cモードハンドラが演算ファンクションに統一してください。

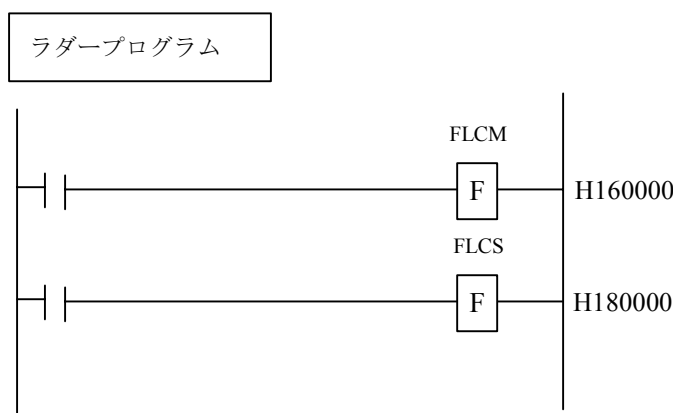
6 利用の手引き

(b) 演算ファンクションによるメッセージ送信要求

演算ファンクションはラダープログラムから呼び出され、FL.NETモジュールに対するメッセージ要求を行い、データの送受信を行います。演算ファンクションはメインモジュール/サブモジュール用が用意されていて、演算ファンクション起動時に指定するパラメータによって、要求したいメッセージ処理を指定します。

ラダープログラムにて演算ファンクションを使用するには、あらかじめ [FL.NET For Windows®] にて、メインモジュール/サブモジュール用の演算ファンクションを登録しておいてください。

FL.NETモジュール用演算ファンクションの登録手順に関しては、「ソフトウェアマニュアル オプション FL.NET For Windows® (マニュアル番号 SAJ-3-139)」を参照してください。



演算ファンクションもFL.NETモジュールへの要求を発行するだけで処理を終了します。

FL.NETモジュールは要求受け付け後、自ノード状態フラグ内のユーザ要求処理中ビットをONし、処理が終了するとユーザ要求処理中ビットをOFFします。

発行後のメッセージ処理終了を確認するには、自ノード状態フラグをチェックしてください。状態で処理終了を表します。

また、演算ファンクションでの要求発行後は、ユーザ要求処理中ビットがOFFするまでメッセージ要求しないでください。要求を発行した場合、そのメッセージ要求は他メッセージ処理中として処理されません（エラーコード格納アドレス内に他メッセージ処理中エラーコードが設定されます）。

ユーザ要求処理中ビットがOFFした後、演算ファンクション起動時に指定したエラーコード格納アドレスに処理終了後の状態を入力します。

要求した処理が正常終了したかは、そのエラーコード格納アドレス内に設定されたコードにて確認してください。

表 6 - 9 演算ファンクション一覧

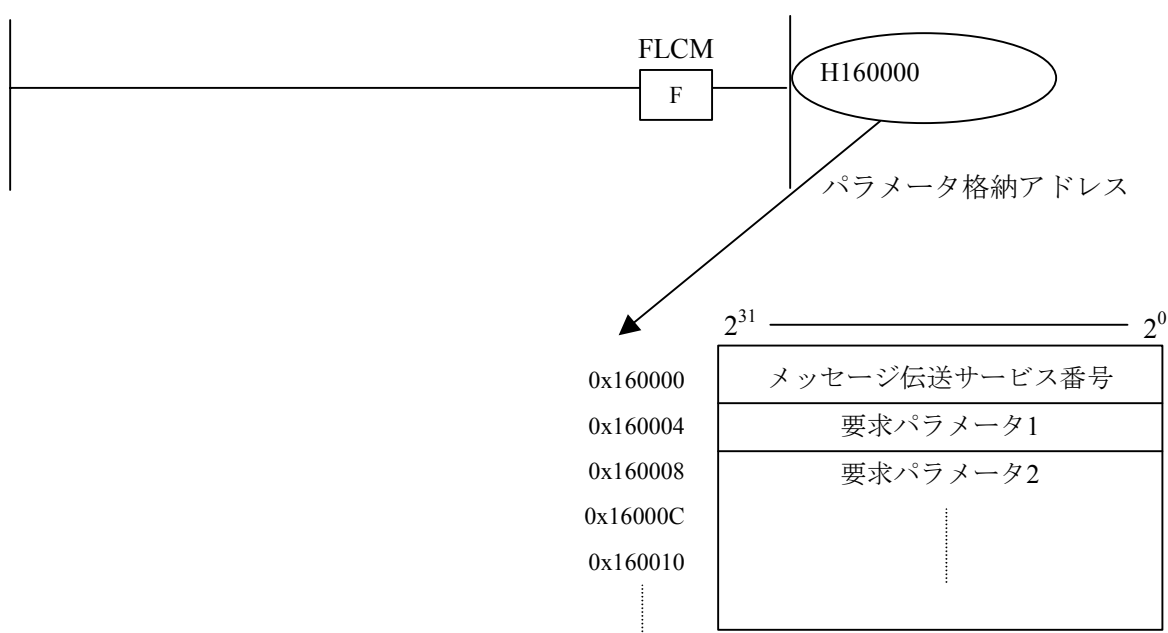
名称		演算ファンクション登録アドレス		機能
メイン	サブ	メイン	サブ	
FLCM	FLCS	/D74100	/DF4100	各種メッセージ伝送要求を行います。

<演算ファンクションによるFL.NETモジュールへの処理要求>

FL.NETモジュール用の演算ファンクションを使用して、ラダープログラムを作成します。

演算ファンクションに指定したアドレスに、要求パラメータを書き込み、演算ファンクションを実行してください。

演算ファンクションによるメッセージ送信要求を行う場合には、メッセージ伝送サービス番号にて、要求するメッセージの種類を指定する必要があります。各種パラメータについては、「(5) 各メッセージ要求時のパラメータ」を参照してください。



演算ファンクションにて使用するパラメータは、すべてロング長で指定してください。

また、演算ファンクションは、使用しているモジュールの設定に合わせて使い分けてください（メインモジュールに要求を発行したい場合には、メインモジュール用演算ファンクション（FLCM）を、サブモジュールに要求を発行したい場合には、サブモジュール用演算ファンクション（FLCS）を使用してください）。

（注1）自ノード状態フラグのビット割り付けに関しては、「6. 4. 6 管理テーブルの使い方」を参照してください。

（注2）演算ファンクションによる要求を行うには、設定ツール [FL.NET For Windows®] を使用して、FL.NET用の演算ファンクションをS10mini上に登録してください。

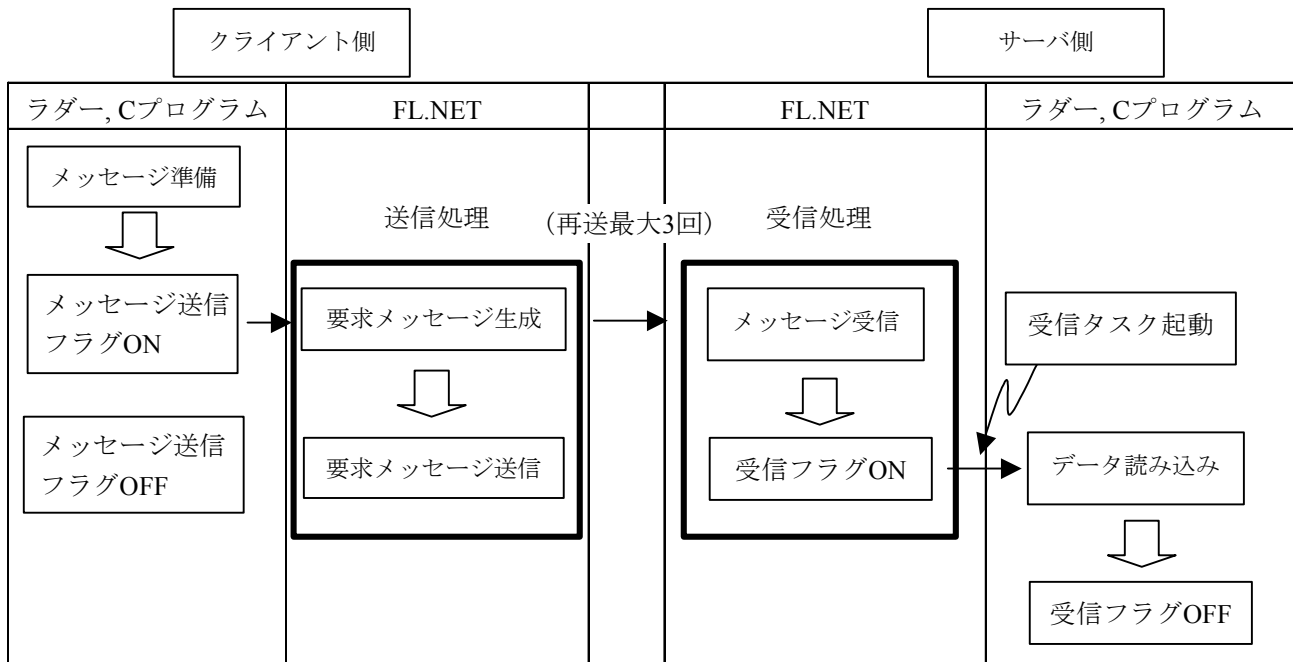
詳しい設定方法に関しては、「ソフトウェアマニュアル オプション FL.NET For Windows®（マニュアル番号 SAJ-3-139）」を参照してください。

FL.NETモジュールへ要求する際には、Cモードハンドラ、演算ファンクションを混在して使用しないでください。

6 利用の手引き

(4) 透過型メッセージの受信について

以下に透過型メッセージの概略動作説明図を示します。



FL.NETモジュールには、透過型メッセージの受信に関する特殊な機能があります。

<透過型受信起動タスク>

FL.NETモジュールに、透過型受信起動タスクが登録されていると、透過型メッセージを受信した際に、そのタスク番号を設定されている起動要因（透過型受信起動要因）でタスクキューします。

ただし、設定したタスクがリリースされていない場合には、設定したタスクは起動しません。

<透過型受信フラグ>

FL.NETモジュールに、透過型受信フラグが設定されていると、その設定アドレスから16ワードのエリアを透過型受信フラグエリアとして使用します。

透過型受信フラグエリアは、FL.NETモジュールが透過型メッセージを受信したとき、そのメッセージがどのノードから送信されたかを示します。

透過型メッセージの受信バッファは合計8ケース分用意されています。

透過型メッセージを受信すると、まず受信バッファに格納されます。そのとき、格納した透過型メッセージの送信元ノード番号をチェックし、透過型受信フラグエリアの送信元ノード番号に対応するビットをONします。

ユーザが透過型メッセージを取り込んだ後、ユーザに渡した透過型メッセージと同じ送信元ノード番号のメッセージが透過型メッセージ用受信バッファ内に残っていないかを確認します。

同じ送信元ノード番号のメッセージが存在しなくなったとき、透過型受信フラグエリアの送信元ノード番号に対応するビットをOFFします。

ただし、FL.NETモジュールにて独自にサポートしている機能（指定タスク制御および指定サブルーチン制御）に対応するTCDの透過型メッセージを受信した場合、これらのメッセージは透過型とみなしません。

透過型受信起動タスク、透過型受信フラグの設定は、設定ツール [FL.NET For Windows®] にて行うことができます。詳しい操作手順に関しては、「ソフトウェアマニュアル オプション FL.NET For Windows® (マニュアル番号 SAJ-3-139)」を参照してください。

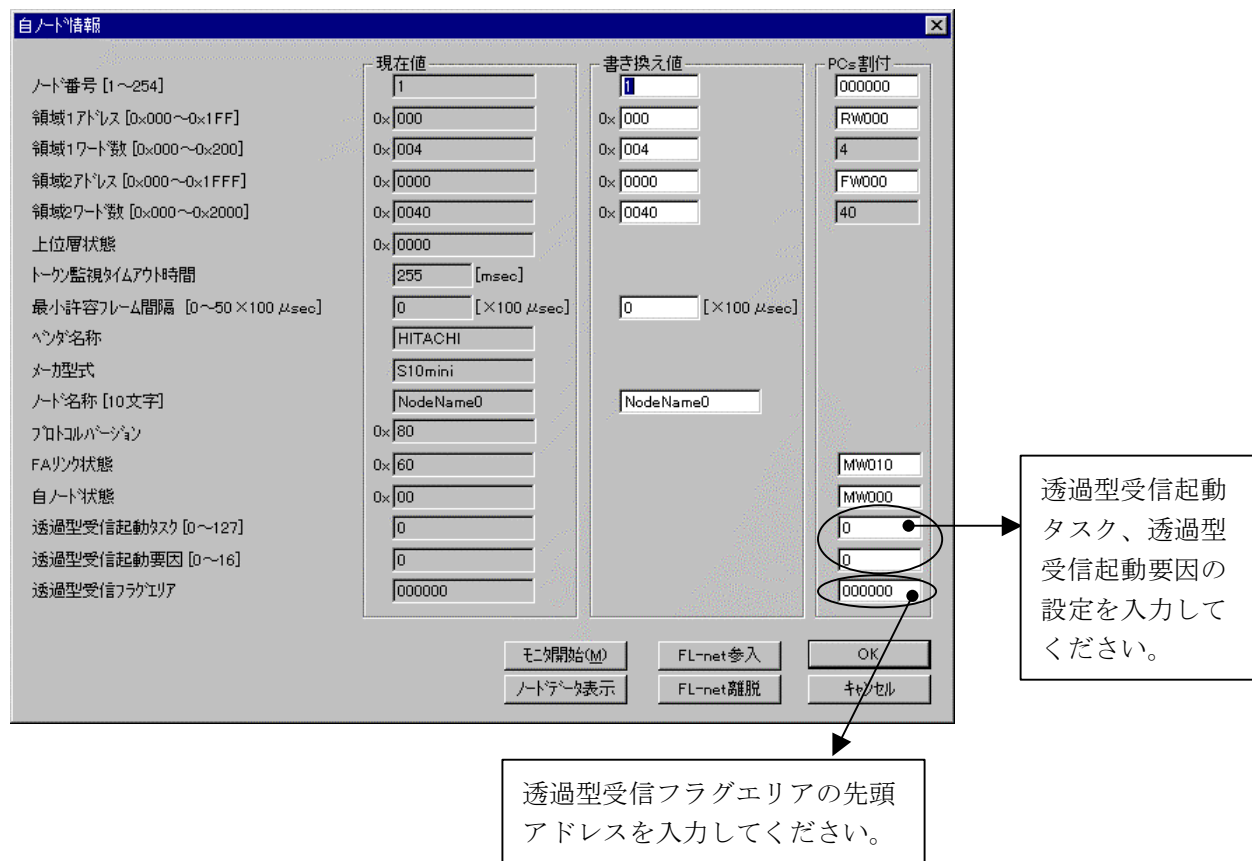


図 6-46 透過型受信フラグエリア、透過型受信起動タスク設定画面

以下に、各項目での入力可能範囲を示します。

入力項目	内容	指定可能範囲
透過型受信起動タスク	透過型メッセージを受信した際に起動するタスク番号	0～127の範囲で設定可能ですが、2～100の範囲で使用してください。
透過型受信起動要因	透過型受信起動タスクの起動要因	0～16
透過型受信フラグエリア	透過型メッセージ受信時に送信元ノード番号を通知するエリア (エリアサイズは16ワードです)。	「表 6-7 コモンメモリ領域をCPUメモリに割り付けできるアドレス範囲」を参照してください。

(注) 透過型受信フラグエリアに設定できるシンボル、拡張メモリアドレスは、CPUメモリ領域エリアの設定可能シンボルと同じです。また、透過型受信フラグエリアは、他の設定領域と重複しないように設定してください。

6 利用の手引き

表 6-10に、透過型受信フラグエリアのビット割り付けを示します。

表 6-10 透過型受信フラグ割り付け

アドレス	ビットNo.															
	2 ¹⁵	2 ¹⁴	2 ¹³	2 ¹²	2 ¹¹	2 ¹⁰	2 ⁹	2 ⁸	2 ⁷	2 ⁶	2 ⁵	2 ⁴	2 ³	2 ²	2 ¹	2 ⁰
0x0000		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0x0002	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0x0004	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
0x0006	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
0x0008	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79
0x000A	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95
0x000C	96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111
0x000E	112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127
0x0010	128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143
0x0012	144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159
0x0014	160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175
0x0016	176	177	178	179	180	181	182	183	184	185	186	187	188	189	190	191
0x0018	192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207
0x001A	208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223
0x001C	224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239
0x001E	240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	

表内の数値は、透過型メッセージの送信元ノード番号を示します。また、空白のビット位置は使用していません。

(表示例)

透過型受信フラグエリアをMW000に設定します。

自ノードが、ノード番号17から透過型メッセージを受信したとき、M011（ノード17用ビット）がONします。

PI/O	ビットデータ
MW000	0x0000
MW010	0x4000
⋮	⋮
MW0F0	0x0000

(5) 各メッセージ要求時のパラメータ

Cモードハンドラおよび演算ファンクションにて、各メッセージ要求を発行する際のパラメータを示します。

ワードブロック読み出し要求 : wordrd()

[リンク手順]

C言語	
メイン	サブ
<pre>Struct wordr_p { long node ; unsigned short *Erradr ; unsigned short *Setadr ; long Readadr ; long Readsiz ; } ; { long (*wordrd)() ; long rtn ; struct wordr_p *padr ; { wordrd = (long*)()0xD74112 ; { rtn = (*wordrd)(padr) ;</pre>	<pre>Struct wordr_p { long node ; unsigned short *Erradr ; unsigned short *Setadr ; long Readadr ; long Readsiz ; } ; { long (*wordrd)() ; long rtn ; struct wordr_p *padr ; { wordrd = (long*)()0xDF4112 ; { rtn = (*wordrd)(padr) ;</pre>

[演算ファンクションでのパラメータ]

+0x0000	メッセージ伝送サービス番号 (3)
+0x0004	相手先ノード番号 (1~254)
+0x0008	エラーコード格納アドレス
+0x000C	読み出しデータ格納アドレス
+0x0010	仮想アドレス
+0x0014	仮想サイズ

6 利用の手引き

[パラメータ]

```
padr                                : 入力パラメータの先頭アドレス
Struct wordr_p {
    long node ;                      : 相手先ノード番号 (1~254)
    unsigned short *Erradr ;         : エラーコード格納アドレス
    unsigned short *Setadr ;         : 読み出しデータ格納アドレス
    long Readadr ;                   : 仮想アドレス
    long Readsiz ;                   : 仮想サイズ (1~512)
};
```

パラメータ項目	入力範囲
メッセージ伝送サービス番号	3 (演算ファンクション使用時のみ)
相手先ノード番号	1~254
エラーコード格納アドレス	S10mini上の実アドレス (偶数アドレス) を指定してください。
読み出しデータ格納アドレス	
仮想アドレス	0~0xFFFFFFFF
仮想サイズ	1~512 (ワード単位)

[機能]

- ・相手先ノードに対し、指定された仮想アドレス、仮想サイズにてワードブロック読み出し要求を発行します。応答コード受信時、読み出しデータ格納アドレス内にデータを転写します。
- ・仮想アドレスは、各FL.NETモジュールによって異なります。

このFL.NETモジュールに対する仮想アドレスの仕様については、「表 3-4 仮想アドレス空間と物理メモリ」を参照してください。

FL.NETモジュールのサーバ機能では、ワードブロック読み出し要求を受信し仮想サイズが512ワードを超える場合には、異常応答を返します。そのときのエラーコードは、0xFFFFFFFFとなります。

ワードブロック書き込み要求 : wordwt()

[リンク手順]

C言語	
メイン	サブ
<pre>Struct wordw_p { long node ; unsigned short *Erradr ; unsigned short *Setadr ; long writeadr ; long writesz ; } } long (*wordwt)() ; long rtn ; struct wordw_p *padr ; } wordwt = (long*)()0xD74118 ; } rtn = (*wordwt)(padr) ;</pre>	<pre>Struct wordw_p { long node ; unsigned short *Erradr ; unsigned short *Setadr ; long writeadr ; long writesz ; } } long (*wordwt)() ; long rtn ; struct wordw_p *padr ; } wordwt = (long*)()0xDF4118 ; } rtn = (*wordwt)(padr) ;</pre>

[演算ファンクションでのパラメータ]

+0x0000	メッセージ伝送サービス番号 (4)
+0x0004	相手先ノード番号 (1~254)
+0x0008	エラーコード格納アドレス
+0x000C	書き込みデータ格納アドレス
+0x0010	仮想アドレス
+0x0014	仮想サイズ

[パラメータ]

padr	: 入力パラメータの先頭アドレス
Struct wordw_p {	
long node ;	: 相手先ノード番号 (1~254)
unsigned short *Erradr ;	: エラーコード格納アドレス
unsigned short *Setadr ;	: 書き込みデータ格納アドレス
long writeadr ;	: 仮想アドレス
long writesz ;	: 仮想サイズ (1~512)
};	

6 利用の手引き

パラメータ項目	入力範囲
メッセージ伝送サービス番号	4 (演算ファンクション使用時のみ)
相手先ノード番号	1~254
エラーコード格納アドレス	S10mini上の実アドレス (偶数アドレス) を指定してください。
書き込みデータ格納アドレス	
仮想アドレス	0~0xFFFFFFFF
仮想サイズ	1~512 (ワード単位)

[機能]

- ・相手先ノードに対し、指定された仮想アドレス、仮想サイズ、書き込みデータ格納アドレス内のデータを使用して、ワードブロック書き込み要求を発行します。
- ・仮想アドレスは、各FL.NETモジュールによって異なります。

このFL.NETモジュールに対する仮想アドレスの仕様については、「表 3-4 仮想アドレス空間と物理メモリ」を参照してください。

FL.NETモジュールのサーバ機能では、ワードブロック書き込み要求を受信し仮想サイズが512ワードを超える、または仮想サイズとデータサイズが一致しない場合、異常応答を返します。そのときのエラーコードは、0xFFFFFFFFとなります。

ネットワークパラメータ読み出し要求 : parard()

[リンク手順]

C言語	
メイン	サブ
<pre>Struct parar_p { long node ; unsigned short *Erradr ; unsigned char *Setadr ; }; } long (*parard) () ; long rtn ; struct parar_p *padr ; } parard = (long(*) ())0xD7411E ; } rtn = (*parard) (padr) ;</pre>	<pre>Struct parar_p { long node ; unsigned short *Erradr ; unsigned char *Setadr ; }; } long (*parard) () ; long rtn ; struct parar_p *padr ; } parard = (long(*) ())0xDF411E ; } rtn = (*parard) (padr) ;</pre>

[演算ファンクションでのパラメータ]

+0x0000	メッセージ伝送サービス番号 (5)
+0x0004	相手先ノード番号 (1~254)
+0x0008	エラーコード格納アドレス
+0x000C	読み出しパラメータデータ格納アドレス

[パラメータ]

padr	: 入力パラメータの先頭アドレス
Struct parar_p {	
long node ;	: 相手先ノード番号 (1~254)
unsigned short *Erradr ;	: エラーコード格納アドレス
unsigned char *Setadr ;	: 読み出しパラメータデータ格納アドレス
};	

パラメータ項目	入力範囲
メッセージ伝送サービス番号	5 (演算ファンクション使用時のみ)
相手先ノード番号	1~254
エラーコード格納アドレス	S10mini上の実アドレス (偶数アドレス)
読み出しパラメータデータ格納アドレス	ス) を指定してください。

6 利用の手引き

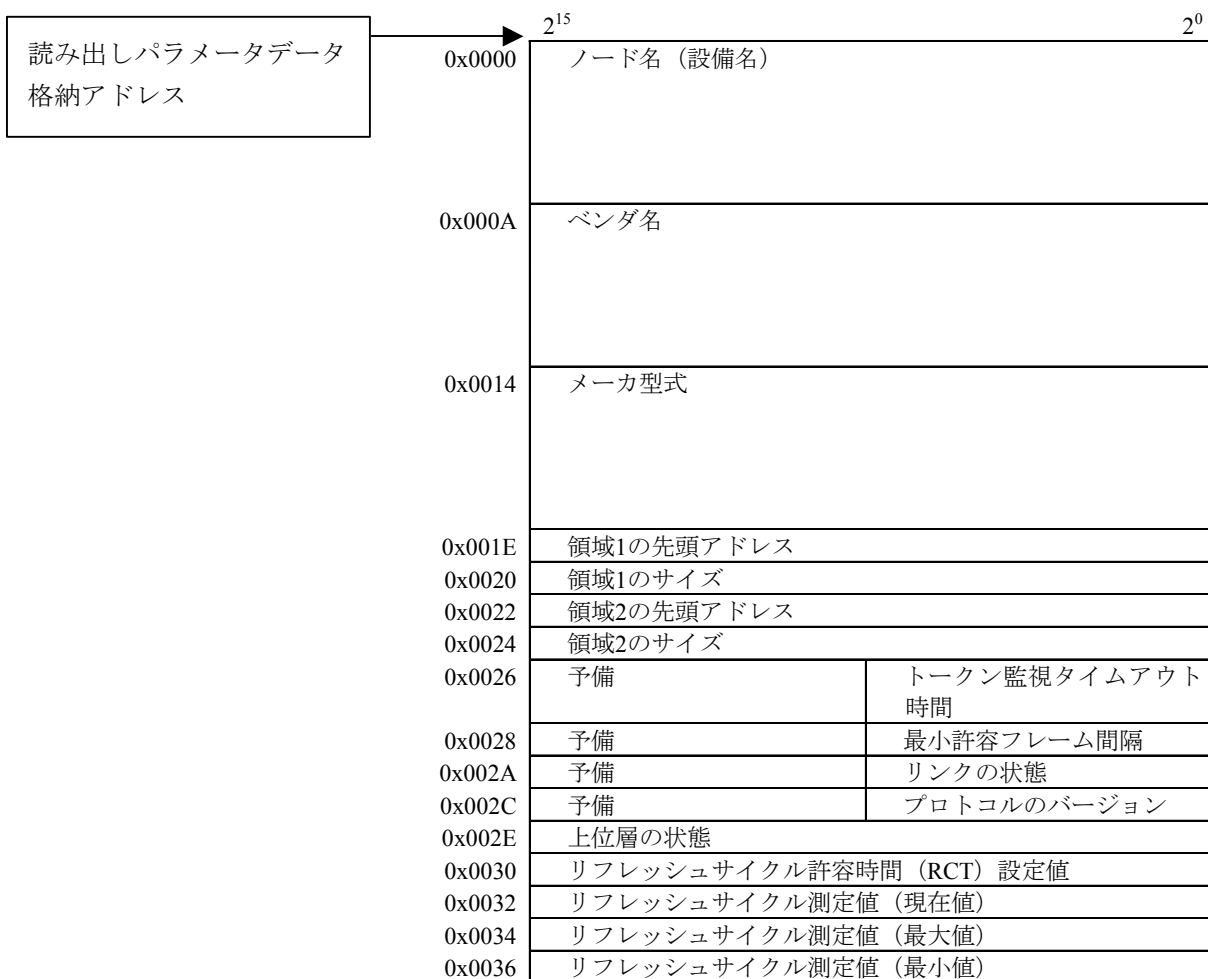
[機能]

相手先ノードに対し、ネットワークパラメータ読み込み要求を発行します。

応答コード受信時、読み出しパラメータデータ格納アドレス内に、指定ノードのネットワークパラメータ情報を転写します。

読み出しパラメータデータ格納アドレスには、次のようなフォーマットでデータを転写します。

ノード番号は、S10mini上には転写しません。



ネットワークパラメータ書き込み要求 : parawt()

[リンク手順]

C言語	
メイン	サブ
<pre>Struct paraw_p { long node ; unsigned short *Erradr ; unsigned char *Dataadr ; } ; { long (*parawt)() ; long rtn ; struct paraw_p *padr ; { parawt = (long(*) ())0xD74124 ; { rtn = (*parawt)(padr) ;</pre>	<pre>Struct paraw_p { long node ; unsigned short *Erradr ; unsigned char *Dataadr ; } ; { long (*parawt)() ; long rtn ; struct paraw_p *padr ; { parawt = (long(*) ())0xDF4124 ; { rtn = (*parawt)(padr) ;</pre>

[演算ファンクションでのパラメータ]

+0x0000	メッセージ伝送サービス番号 (6)
+0x0004	相手先ノード番号 (1~254)
+0x0008	エラーコード格納アドレス
+0x000C	書き込みパラメータデータ格納アドレス

[パラメータ]

padr	: 入力パラメータの先頭アドレス
Struct paraw_p {	
long node ;	: 相手先ノード番号 (1~254)
unsigned short *Erradr ;	: エラーコード格納アドレス
unsigned char *Dataadr ;	: 書き込みパラメータデータ格納アドレス
};	

パラメータ項目	入力範囲
メッセージ伝送サービス番号	6 (演算ファンクション使用時のみ)
相手先ノード番号	1~254
エラーコード格納アドレス	S10mini上の実アドレス (偶数アドレス) を指定してください。
書き込みパラメータデータ格納アドレス	

6 利用の手引き

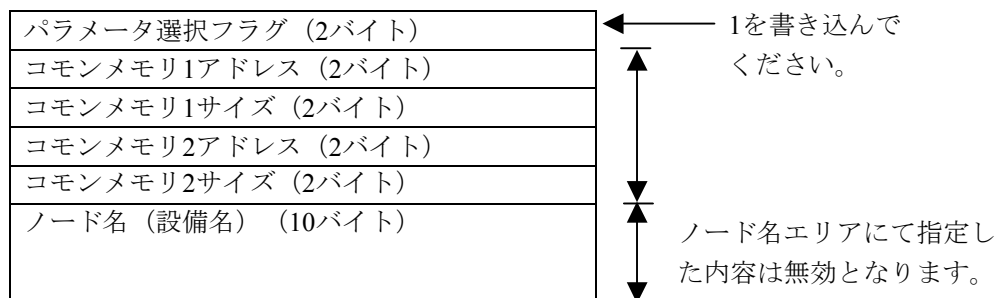
[機能]

相手先ノードに対し、書き込みパラメータデータ格納アドレス内のパラメータにて、ネットワークパラメータ書き込み要求を発行します。

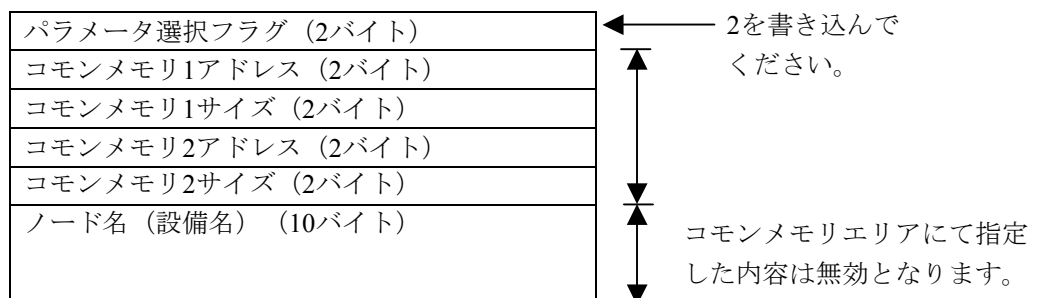
[書き込みパラメータデータ構成] (連続エリア)

書き込みパラメータデータ格納アドレスに指定したエリアに、以下のようなデータを作成してください。

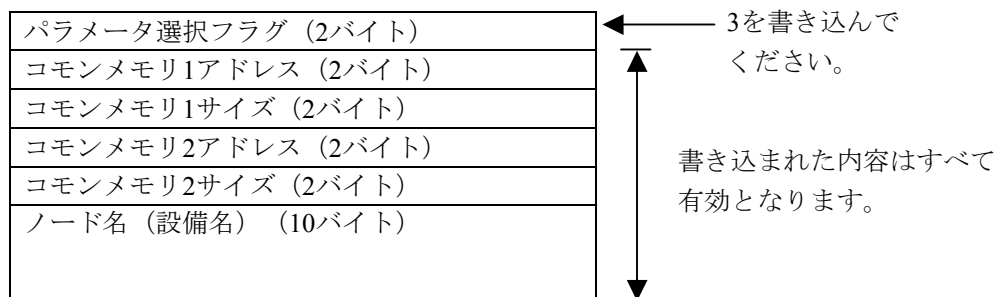
1. コモンメモリアドレス、コモンメモリサイズのみ書き換えたい場合



2. ノード名 (設備名) のみ書き換えたい場合



3. コモンメモリアドレス、コモンメモリサイズ、ノード名 (設備名) すべてを書き換えたい場合



- ネットワークパラメータは、コモンメモリなどの設定が重複しないように十分注意して書き換えてください。特にこのモジュールに対して実行し、S10mini上のコモンメモリアreaが重複するとFL.NETモジュールの再設定が必要となる可能性があります。
- FL.NETモジュールのサーバ機能では、ネットワークパラメータの書き込み要求を受信しパラメータ選択フラグが1~3以外の場合は、異常応答を返します。その場合のエラーコードは0xFFFFFFFFとなります。

停止指令要求 : reqstop()

[リンク手順]

C言語	
メイン	サブ
<pre>Struct reqs_p { long node ; unsigned short *Erradr ; } ; { long (*ReqStop) () ; long rtn ; struct reqs_p *padr ; { ReqStop = (long(*) ())0xD7412A ; { rtn = (*ReqStop) (padr) ;</pre>	<pre>Struct reqs_p { long node ; unsigned short *Erradr ; } ; { long (*ReqStop) () ; long rtn ; struct reqs_p *padr ; { ReqStop = (long(*) ())0xDF412A ; { rtn = (*ReqStop) (padr) ;</pre>

[演算ファンクションでのパラメータ]

+0x0000	メッセージ伝送サービス番号 (7)
+0x0004	相手先ノード番号 (1~254)
+0x0008	エラーコード格納アドレス

6 利用の手引き

[パラメータ]

```
padr : 入力パラメータの先頭アドレス  
Struct reqs_p {  
    long node ; : 相手先ノード番号 (1~254)  
    unsigned short *Erradr ; : エラーコード格納アドレス  
};
```

パラメータ項目	入力範囲
メッセージ伝送サービス番号	7 (演算ファンクション使用時のみ)
相手先ノード番号	1~254
エラーコード格納アドレス	S10mini上の実アドレス (偶数アドレス) を指定してください。

[機能]

相手先ノードに対し、停止指令要求を発行します。

FL.NETモジュールでは、停止指令に対するサーバ機能をサポートしていません。FL.NETモジュールに対して、停止指令要求を発行した場合、未サポート応答が返ってきます。

運転指令要求 : reqrun()

[リンク手順]

C言語	
メイン	サブ
<pre>Struct reqr_p { long node ; unsigned short *Erradr ; } ; { long (*ReqRun)() ; long rtn ; struct reqr_p *padr ; { ReqRun = (long(*) ())0xD74130 ; { rtn = (*ReqRun) (padr) ;</pre>	<pre>Struct reqr_p { long node ; unsigned short *Erradr ; } ; { long (*ReqRun)() ; long rtn ; struct reqr_p *padr ; { ReqRun = (long(*) ())0xDF4130 ; { rtn = (*ReqRun) (padr) ;</pre>

[演算ファンクションでのパラメータ]

+0x0000	メッセージ伝送サービス番号 (8)
+0x0004	相手先ノード番号 (1~254)
+0x0008	エラーコード格納アドレス

[パラメータ]

padr	: 入力パラメータの先頭アドレス
Struct reqr_p {	
long node ;	: 相手先ノード番号 (1~254)
unsigned short *Erradr ;	: エラーコード格納アドレス
} ;	

パラメータ項目	入力範囲
メッセージ伝送サービス番号	8 (演算ファンクション使用時のみ)
相手先ノード番号	1~254
エラーコード格納アドレス	S10mini上の実アドレス (偶数アドレス) を指定してください。

6 利用の手引き

[機能]

相手先ノードに対し、運転指令要求を発行します。

FL.NETモジュールでは、運転指令に対するサーバ機能をサポートしていません。FL.NETモジュールに対して、運転指令要求を発行した場合、未サポート応答が返ってきます。

プロファイルの読み出し

このモジュールでは、他ノードに対してプロファイルの読み出しのクライアント機能はサポートしていませんが他ノード要求に対する応答は返します。

以下に、他ノードからのプロファイルの読み出し要求に対するFL.NETモジュールの応答内容を示します。

識別子	長さ			
0x30	0x816A			
		識別子	長さ	
		0x30	0x68	
		識別子	長さ	内容
		0x13	0x06	“COMVER”
		0x02	0x01	“0x01”
		0x13	0x02	“ID”
		0x13	0x07	“SYSPARA”
		0x13	0x03	“REV”
		0x02	0x01	“0x01”
		0x13	0x07	“REVDATA”
		0x30	0x0A	
		0x02	0x02	“0x07CF (1999)”
		0x02	0x01	“0x0B(11)”
		0x02	0x01	“0x0F(15)”
		0x13	0x0A	“DVCATEGORY”
		0x13	0x03	“PLC”
		0x13	0x06	“VENDOR”
		0x13	0x07	“HITACHI”
		0x13	0x07	“DVMODEL”
		0x13	0x06	“LQE000”

6 利用の手引き

通信ログデータの読み出し要求 : logrd()

[リンク手順]

C言語	
メイン	サブ
<pre> Struct logr_p { long node ; unsigned short *Erradr ; unsigned char *logadr ; } ; { long (*Logrd)() ; long rtn ; struct logr_p *padr ; } Logrd = (long(*) ())0xD7413C ; } rtn = (*Logrd)(padr) ; </pre>	<pre> Struct logr_p { long node ; unsigned short *Erradr ; unsigned char *logadr ; } ; { long (*Logrd)() ; long rtn ; struct logr_p *padr ; } Logrd = (long(*) ())0xDF413C ; } rtn = (*Logrd)(padr) ; </pre>

[演算ファンクションでのパラメータ]

+0x0000	メッセージ伝送サービス番号 (10)
+0x0004	相手先ノード番号 (1~254)
+0x0008	エラーコード格納アドレス
+0x000C	ログデータ格納アドレス

[パラメータ]

```

padr                                : 入力パラメータの先頭アドレス
Struct logr_p {
    long node ;                      : 相手先ノード番号 (1~254)
    unsigned short *Erradr ;        : エラーコード格納アドレス
    unsigned char *logadr ;         : ログデータ格納アドレス
};

```

パラメータ項目	入力範囲
メッセージ伝送サービス番号	10 (演算ファンクション使用時のみ)
相手先ノード番号	1~254
エラーコード格納アドレス	S10mini上の実アドレス (偶数アドレス) を指定してください。
ログデータ格納アドレス	

[機能]

- ・相手先ノードに対し、通信ログデータの読み出し要求を発行します。
応答コードが返ってきた場合、応答コード内の通信ログデータをログデータ格納アドレスに転写します。
- ・転写される通信ログデータは512バイトです。各データは、4バイトデータです。
通信ログデータの詳細に関しては、「6. 4. 8 通信ログの使い方」を参照してください。

6 利用の手引き

通信ログデータのクリア要求 : logclr()

[リンク手順]

C言語	
メイン	サブ
<pre>Struct logclr_p { long node ; unsigned short *Erradr ; } ; { long (*Logclr)() ; long rtn ; struct logclr_p *padr ; { Logclr = (long(*) ()) 0xD74142 ; { rtn = (*Logclr)(padr) ;</pre>	<pre>Struct logclr_p { long node ; unsigned short *Erradr ; } ; { long (*Logclr)() ; long rtn ; struct logclr_p *padr ; { Logclr = (long(*) ())0xDF4142 ; { rtn = (*Logclr)(padr) ;</pre>

[演算ファンクションでのパラメータ]

+0x0000	メッセージ伝送サービス番号 (11)
+0x0004	相手先ノード番号 (1~255)
+0x0008	エラーコード格納アドレス

[パラメータ]

padr : 入力パラメータの先頭アドレス

Struct logclr_p {

 long node ; : 相手先ノード番号 (1~255)

 unsigned short *Erradr ; : エラーコード格納アドレス

} ;

パラメータ項目	入力範囲
メッセージ伝送サービス番号	11 (演算ファンクション使用時のみ)
相手先ノード番号	1~255
エラーコード格納アドレス	S10mini上の実アドレス (偶数アドレス) を指定してください。

[機能]

- ・相手先ノードに対し、通信ログデータのクリア要求を発行します。
- ・相手先ノード番号を255に指定すると、1対N伝送となります。

6 利用の手引き

メッセージ折り返し要求 : mesret()

[リンク手順]

C言語	
メイン	サブ
<pre>Struct mesreq_p { long node ; unsigned short *Erradr ; unsigned char *SendData ; long Sendsiz ; unsigned char *Recvadr ; }; { long (*Mesret)() ; long rtn ; struct mesreq_p *padr ; { Mesret = (long*)()0xD74148 ; { rtn = (*Mesret)(padr) ;</pre>	<pre>Struct mesreq_p { long node ; unsigned short *Erradr ; unsigned char *SendData ; long Sendsiz ; unsigned char *Recvadr ; }; { long (*Mesret)() ; long rtn ; struct mesreq_p *padr ; { Mesret = (long*)()0xDF4148 ; { rtn = (*Mesret)(padr) ;</pre>

[演算ファンクションでのパラメータ]

+0x0000	メッセージ伝送サービス番号 (12)
+0x0004	相手先ノード番号 (1~254)
+0x0008	エラーコード格納アドレス
+0x000C	メッセージデータ格納先頭アドレス
+0x0010	メッセージデータサイズ
+0x0014	折り返しメッセージ格納アドレス

[パラメータ]

```

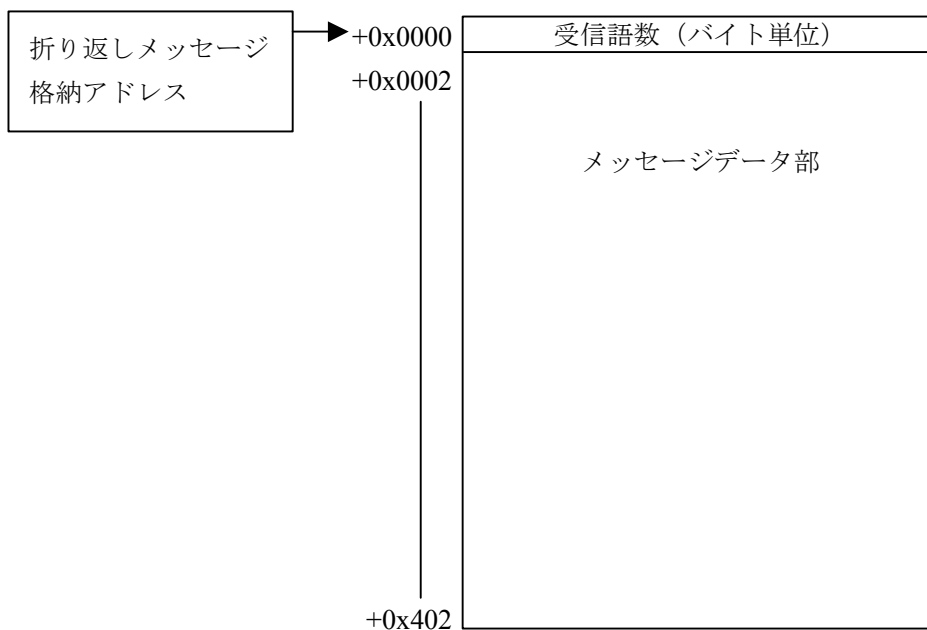
padr                               : 入力パラメータの先頭アドレス
Struct mesreq_p {
    long node ;                     : 相手先ノード番号 (1~254)
    unsigned short *Erradr ;        : エラーコード格納アドレス
    unsigned char *SendData ;       : メッセージデータ格納先頭アドレス
    long Sendsiz ;                  : メッセージデータサイズ (1~1024)
    unsigned char *Recvadr ;        : 折り返しメッセージ格納アドレス
};

```

パラメータ項目	入力範囲
メッセージ伝送サービス番号	12 (演算ファンクション使用時のみ)
相手先ノード番号	1~254
エラーコード格納アドレス	S10mini上の実アドレス (偶数アドレス) を指定してください。
メッセージデータ格納先頭アドレス	
メッセージデータサイズ	1~1024
折り返しメッセージ格納アドレス	S10mini上の実アドレス (偶数アドレス) を指定してください。

[機能]

- ・相手先ノードに対し、メッセージデータ格納先頭アドレス内のデータにて、メッセージ折り返し要求を発行します。応答メッセージが返ってきた場合、応答コード内のメッセージデータを折り返しメッセージ格納アドレスに転写します。
- ・折り返しメッセージ格納アドレスには、次のようなフォーマットでデータを転写します。



6 利用の手引き

指定タスク制御（日立独自透過型サポート）：reqmacro()

[動作内容]

ユーザが指定したタスク番号に対する制御（アボート、リリース、キュー）を実行します。

[リンク手順]

C言語	
メイン	サブ
<pre> Struct Reqmacro_p { long node unsigned short *Erradr ; unsigned long *Retadr ; long *ParaCnt ; unsigned long Para[3] ; } ; { long (*ReqMacro) () ; long rtn ; struct Reqmacro_p *padr ; { ReqMacro = (long(*) ())0xD74160 ; { rtn = (*ReqMacro) (padr) ; </pre>	<pre> Struct Reqmacro_p { long node unsigned short *Erradr ; unsigned long *Retadr ; long *ParaCnt ; unsigned long Para[3] ; } ; { long (*ReqMacro) () ; long rtn ; struct Reqmacro_p *padr ; { ReqMacro = (long(*) ())0xDF4160 ; { rtn = (*ReqMacro) (padr) ; </pre>

[演算ファンクションでのパラメータ]

+0x0000	メッセージ伝送サービス番号 (13)
+0x0004	相手先ノード番号 (1~254)
+0x0008	エラーコード格納アドレス
+0x000C	CPMSマクロ実行結果格納アドレス
+0x0010	CPMSマクロパラメータ数
+0x0014	CPMSマクロパラメータ1
+0x0018	CPMSマクロパラメータ2
+0x001C	CPMSマクロパラメータ3

[パラメータ]

```

padr                                : 入力パラメータの先頭アドレス
Struct Reqmacro_p {
    long node ;                       : 相手先ノード番号 (1~254)
    unsigned short *Erradr ;          : エラーコード格納アドレス
    unsigned long *Retadr ;           : CPMSマクロ実行結果格納アドレス
    long *ParaCnt ;                   : CPMSマクロパラメータ数
    unsigned long Para[3] ;           : CPMSマクロパラメータ
};

```

パラメータ項目	入力範囲
メッセージ伝送サービス番号	13 (演算ファンクション使用時のみ)
相手先ノード番号	1~254
エラーコード格納アドレス	S10mini上の実アドレス (偶数アドレス) を指定してください。
CPMSマクロ実行結果格納アドレス	
CPMSマクロパラメータ数	[機能] を参照してください。
CPMSマクロパラメータ	

[機能]

相手先ノードに対して、タスクアポート/リリース/キューのいずれかの実行を要求できます。

応答メッセージを受信した際、CPMSマクロ実行結果格納アドレスへタスク制御の実行結果 (ロング値) が書き込まれます。

各パラメータを以下のように指定してください。

<タスクアポートを実行させたい場合>

CPMSマクロパラメータ数=2

CPMSマクロパラメータ [0] =1

CPMSマクロパラメータ [1] =タスク番号 (1~128)

<タスクリリースを実行させたい場合>

CPMSマクロパラメータ数=2

CPMSマクロパラメータ [0] =2

CPMSマクロパラメータ [1] =タスク番号 (1~128)

<タスクキューを実行させたい場合>

CPMSマクロパラメータ数=3

CPMSマクロパラメータ [0] =3

CPMSマクロパラメータ [1] =タスク番号 (1~128)

CPMSマクロパラメータ [2] =タスク起動要因 (0~16)

6 利用の手引き

透過型メッセージ要求は、十分注意して使用してください。
タスク番号の指定を間違えると、相手ノード（S10mini）のCPUダウンを引き起こす可能性があります。

透過型メッセージ受信（日立独自透過型サポート）：toukaread()

[リンク手順]

C言語	
メイン	サブ
<pre>Struct ToukaRead_p { long node ; unsigned short *Erradr ; unsigned char *dataadr ; unsigned long datasiz ; } ; { long (*ToukaRead)() ; long rtn ; struct ToukaRead_p *padr ; { ToukaRead = (long*)()0XD74178 ; { rtn = (*ToukaRead)(padr) ;</pre>	<pre>Struct ToukaRead_p { long node ; unsigned short *Erradr ; unsigned char *dataadr ; unsigned long datasiz ; } ; { long (*ToukaRead)() ; long rtn ; struct ToukaRead_p *padr ; { ToukaRead = (long*)()0XDF4178 ; { rtn = (*ToukaRead)(padr) ;</pre>

[演算ファンクションでのパラメータ]

+0x0000	メッセージ伝送サービス番号 (17)
+0x0004	受信先ノード番号 (0~254)
+0x0008	エラーコード格納アドレス
+0x000C	受信データ格納アドレス
+0x0010	データ語数

[パラメータ]

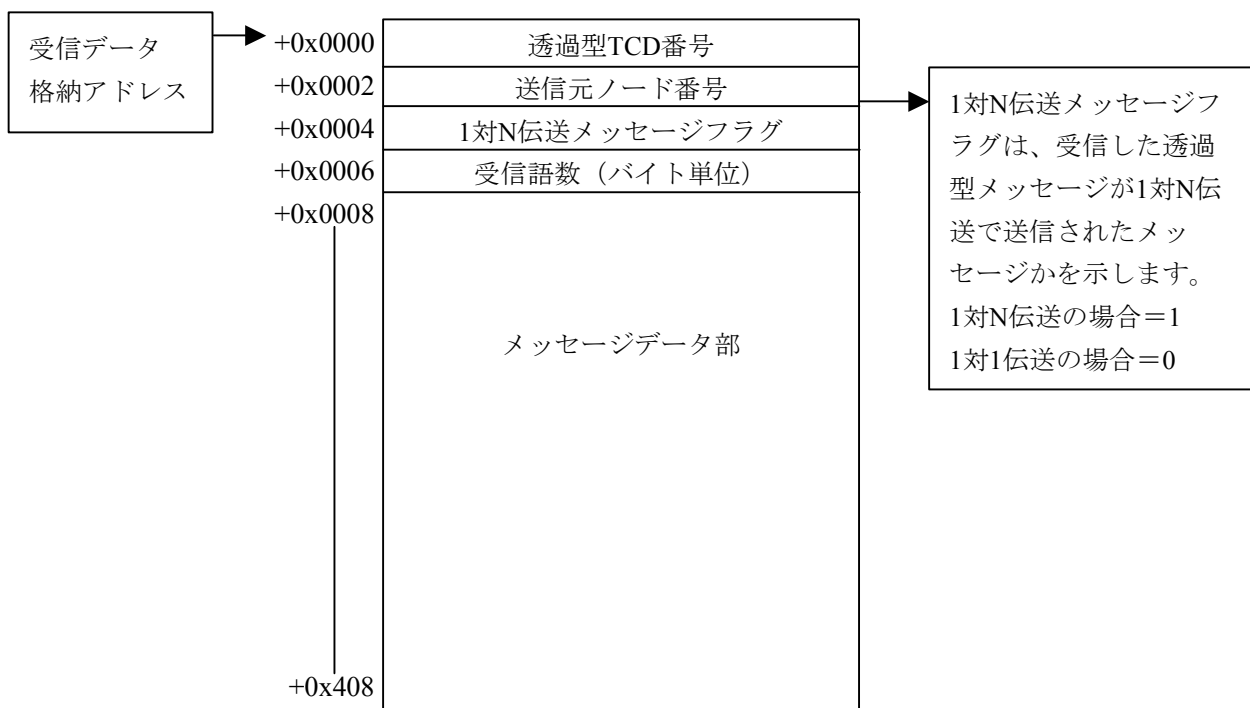
padr	: 入力パラメータの先頭アドレス
Struct ToukaRead_p {	
long node ;	: 受信先ノード番号 (0~254)
unsigned short *Erradr ;	: エラーコード格納アドレス
unsigned char *dataadr ;	: 受信データ格納アドレス
unsigned long datasiz ;	: データ語数 (0~1024)
} ;	

6 利用の手引き

パラメータ項目	入力範囲
メッセージ伝送サービス番号	17 (演算ファンクション使用時のみ)
受信先ノード番号	0~254
エラーコード格納アドレス	S10mini上の実アドレス (偶数アドレス) を指定してください。
受信データ格納アドレス	
データ語数	0~1024 (バイト単位)

[機能]

- ・受信済みの透過型メッセージを、受信データ格納アドレスにて指定されたエリアへ、受信語数分転写します。
- ・受信先ノード番号を指定すると、そのノード番号からの透過型メッセージが存在しない場合、パラメータエラーとなります。
- ・受信先ノード番号に0を指定すると、送信元ノード番号を気にせずに、透過型メッセージキュー内で最初に検索されたデータを受信データ格納アドレスに転写します (受信順ではありません)。透過型メッセージの受信が全くない場合、パラメータエラーとなります。
- ・ユーザから指定されたデータ語数が実際の受信語数より小さい場合、メッセージデータはユーザ指定語数分のみ転写されます。
- ・受信データ格納アドレスには、次のようなフォーマットで透過型メッセージを転写します。



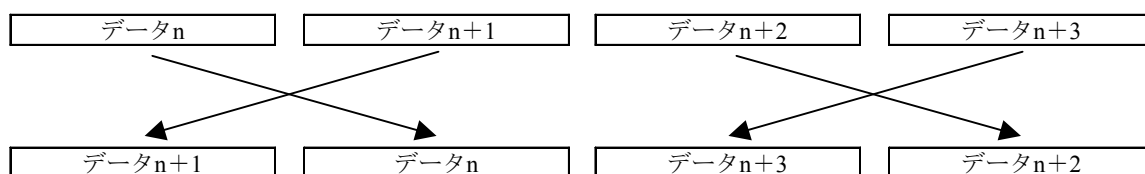
(注) ユーザから指定されたデータ語数が奇数バイトである場合、最終転写受信データに1バイトのデータ「0x00」を付加します。

● FL.NETモジュール独自機能

受信した透過型メッセージTCD番号が0～999である場合、受信メッセージデータをワードデータのリトルエンディアン形式のデータとして扱います。ただし、受信メッセージのデータ語数が奇数バイトだった場合、リトルエンディアン形式のデータ変換が正常に行われません。それ以外のTCD番号が指定された場合には、受信データをそのまま転写します。

TCD番号0～999の透過型メッセージを受信した場合のデータ変換

例：4バイトのデータを受信



ワード単位でバイトスワップされて、受信します。

6 利用の手引き

透過型メッセージ送信（日立独自透過型サポート）：toukasend()

[リンク手順]

C言語	
メイン	サブ
<pre>Struct ToukaSend_p { long node ; unsigned short *Erradr ; unsigned char *dataadr ; unsigned long datasiz ; unsigned long TcdNo ; }; { long (*ToukaSend) () ; long rtn ; struct ToukaSend_p *padr ; { ToukaSend = (long(*) ())0XD7417E ; { rtn = (*ToukaSend) (padr) ;</pre>	<pre>Struct ToukaSend_p { long node ; unsigned short *Erradr ; unsigned char *dataadr ; unsigned long datasiz ; unsigned long TcdNo ; }; { long (*ToukaSend) () ; long rtn ; struct ToukaSend_p *padr ; { ToukaSend = (long(*) ())0xDF417E ; { rtn = (*ToukaSend) (padr) ;</pre>

[演算ファンクションでのパラメータ]

+0x0000	メッセージ伝送サービス番号 (18)
+0x0004	送信先ノード番号 (1~255)
+0x0008	エラーコード格納アドレス
+0x000C	送信データ格納アドレス
+0x0010	データ語数 (0~1024)
+0x0014	送信メッセージTCD

[パラメータ]

padr	: 入力パラメータの先頭アドレス
Struct ToukaSend_p {	
long node ;	: 送信先ノード番号 (1~255)
unsigned short *Erradr ;	: エラーコード格納アドレス
unsigned char *dataadr ;	: 送信データ格納アドレス
unsigned long datasiz ;	: データ語数 (0~1024バイト)
unsigned long TcdNo ;	: 送信メッセージTCD (0~59999)
};	

パラメータ項目	入力範囲
メッセージ伝送サービス番号	18 (演算ファンクション使用時のみ)
送信先ノード番号	1~255
エラーコード格納アドレス	S10mini上の実アドレス (偶数アドレス) を指定してください。
送信データ格納アドレス	
データ語数	0~1024 (バイト単位)
送信メッセージTCD	0~59999 (ただし、10000, 10001, 10200, 10201は除く)

[機能]

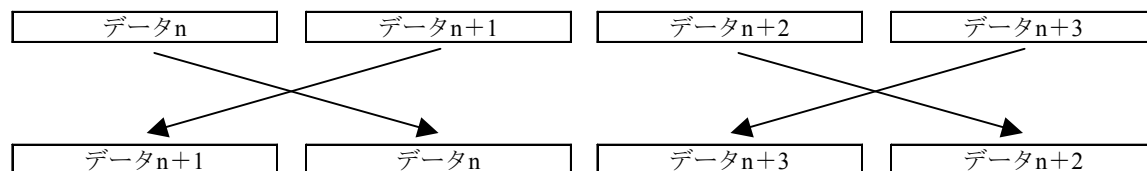
- 送信データ格納アドレスにて指定されたエリアのデータを語数分 (バイト数指定)、送信メッセージTCDにて指定されたTCD番号の透過型メッセージとして送信先ノード番号に転送します。
- 送信先ノード番号に255を指定すると、1対N伝送になります。
- データ語数に0を指定すると、データを付加しないで送信します。
- 送信メッセージTCDには、透過型メッセージ独自サポート機能 (指定タスク制御/指定サブルーチン制御) にて使用しているTCD番号 (10000, 10001, 10200, 10201) を指定できません。

● FL.NETモジュール独自機能

TCD番号に0~999を指定した場合、送信データをリトルエンディアン形式のデータ扱いで送信します。また、送信語数が奇数バイトだった場合には、送信データに1バイトのデータ「0x00」を付加し、リトルエンディアン形式に変換します。それ以外のTCD番号が指定された場合には、送信データをそのまま送信します。

TCD番号0~999に指定された場合のデータ変換

例：4バイトのデータを受信



ワード単位でバイトスワップされて、送信します。

6 利用の手引き

コモンメモリオフセット機能 : comoffset()

[動作内容]

このハンドラを実行することにより、データ転写時の開始アドレスを指定ノードのコモンメモリ領域の先頭から指定オフセットだけずれた位置に設定することができます。

ノードごとにコモンメモリ領域1,2それぞれ別々のオフセットサイズを指定できます。

このハンドラは、受信したデータのうち1部のみ使用する場合に有効です。

以下に通常動作（オフセットなし）とオフセット指定したときの動作の違いを示します。

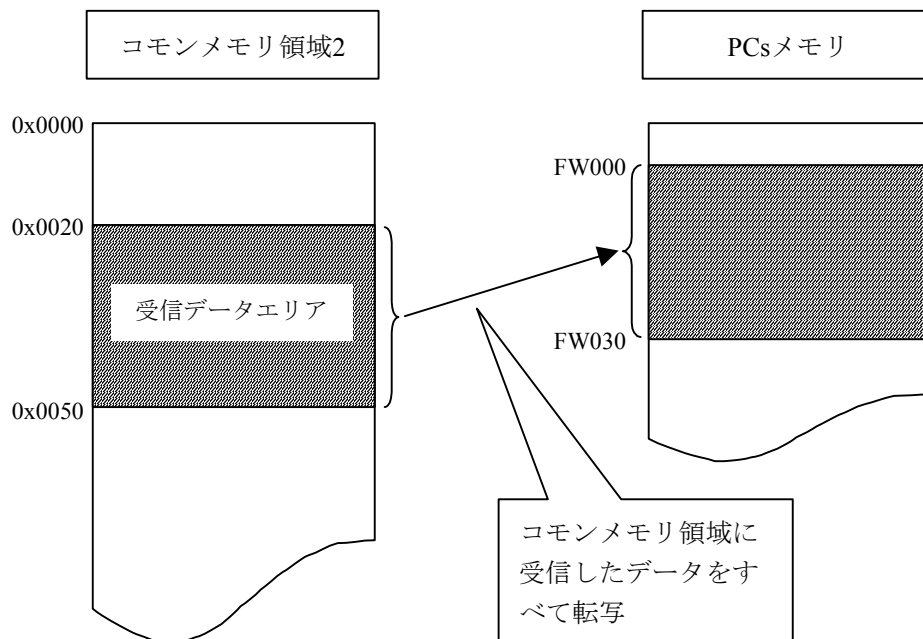
例：通常動作時（オフセット指定なし）

コモンメモリ領域2アドレス：0x0020

コモンメモリ領域2ワード数：0x0030

PCsメモリアドレス：FW000

PCsメモリワード数：30



オフセット指定あり時

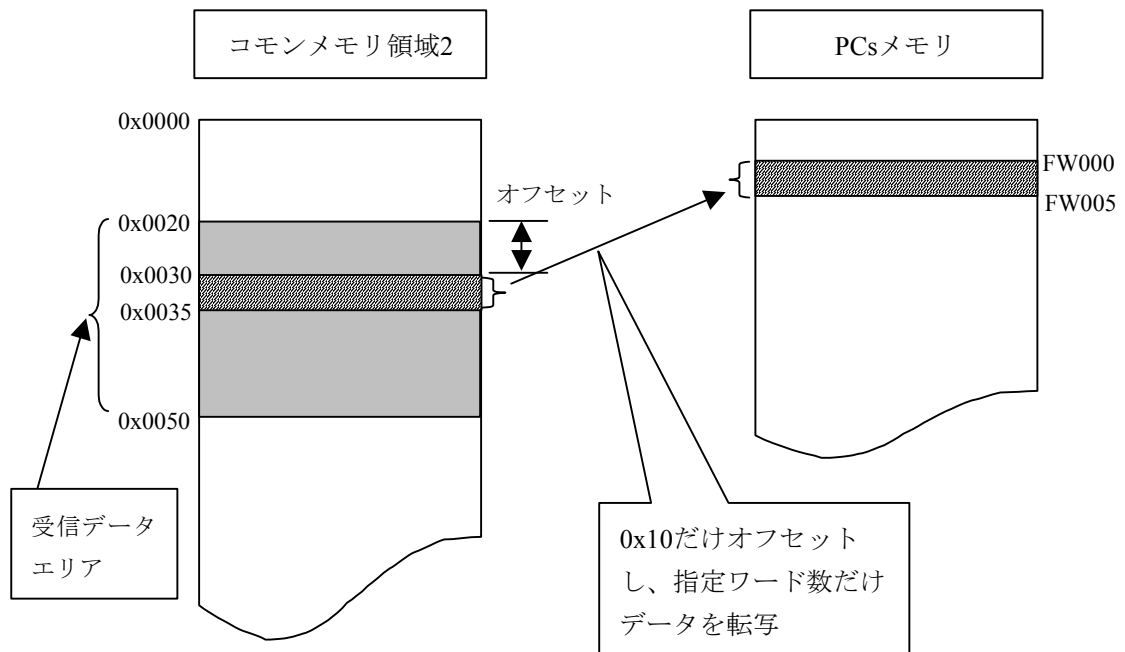
コモンメモリ領域2アドレス : 0x0020

コモンメモリ領域2ワード数 : 0x0030

PCsメモリアドレス : FW000

PCsメモリワード数 : 5

オフセットサイズ : 0x10



6 利用の手引き

[リンク手順]

C言語	
メイン	サブ
<pre>Struct CommonOffset_p { long node ; unsigned short *Erradr ; unsigned long comloffset ; unsigned long com2offset ; } ; { long (*comoffset)() ; long rtn ; struct CommonOffset_p *padr ; { comoffset= (long(*) ())0xD74184 ; { rtn = (*comoffset)(padr) ;</pre>	<pre>Struct CommonOffset_p { long node ; unsigned short *Erradr ; unsigned long comloffset ; unsigned long com2offset ; } ; { long (*comoffset)() ; long rtn ; struct CommonOffset_p *padr ; { comoffset= (long(*) ())0xDF4184 ; { rtn = (*comoffset)(padr) ;</pre>

[演算ファンクションでのパラメータ]

+0x0000	メッセージ伝送サービス番号 (19)
+0x0004	指定ノード番号 (1~254)
+0x0008	エラーコード格納エリア
+0x000C	コモンメモリ領域1オフセットサイズ
+0x0010	コモンメモリ領域2オフセットサイズ

[パラメータ]

padr	: 入力パラメータの先頭アドレス
Struct CommonOffset_p {	
long node ;	: 指定ノード番号 (1~254)
unsigned short *Erradr ;	: エラーコード格納アドレス
unsigned long comloffset ;	: コモンメモリ領域1オフセットサイズ
unsigned long com2offset ;	: コモンメモリ領域2オフセットサイズ
};	

パラメータ項目	入力範囲
メッセージ伝送サービス番号	19 (演算ファンクション使用時のみ)
指定ノード番号	1~254
エラーコード格納アドレス	S10mini上の実アドレス (偶数アドレス) を指定してください。
COMMONメモリ領域1オフセットサイズ	0~0x1FF (ワード単位)
COMMONメモリ領域2オフセットサイズ	0~0x1FFF (ワード単位)

[機能]

- ・指定ノード番号のCOMMONメモリ転写時、オフセットサイズをFL.NETモジュールに通知します。
- ・指定ノード番号に、自ノード番号を指定すると、パラメータエラーとなります。
- ・COMMONメモリ領域1オフセットサイズにCOMMONメモリ領域1最大サイズ (0x200) 以上のサイズを指定するとパラメータエラーとなります。
- ・COMMONメモリ領域2オフセットサイズにCOMMONメモリ領域2最大サイズ (0x2000) 以上のサイズを指定するとパラメータエラーとなります。
- ・メッセージ処理中にオフセットサイズの指定を行うと、他メッセージ処理中となります。

- 指定ノード番号に対するオフセットサイズの指定を行うと、FL.NETモジュールが電源断またはリセットされない限り、オフセットサイズを記憶します。しかし、オフセット設定後、ユーザがFL.NETモジュール内のオフセット設定を参照することはできません。オフセット設定はユーザにて管理してください。
- この機能は、S10mini上のメモリ領域が不足している場合に使用されることを想定した機能です。S10mini上のメモリ領域が不足していない場合は、使用しないでください。
- オフセットをすることにより各COMMONメモリ領域 (領域1、領域2) のアドレスを超えてしまった場合、その超えてしまったアドレスのデータはS10miniのメモリに転写されません。COMMONメモリの領域を超えないよう使用してください。

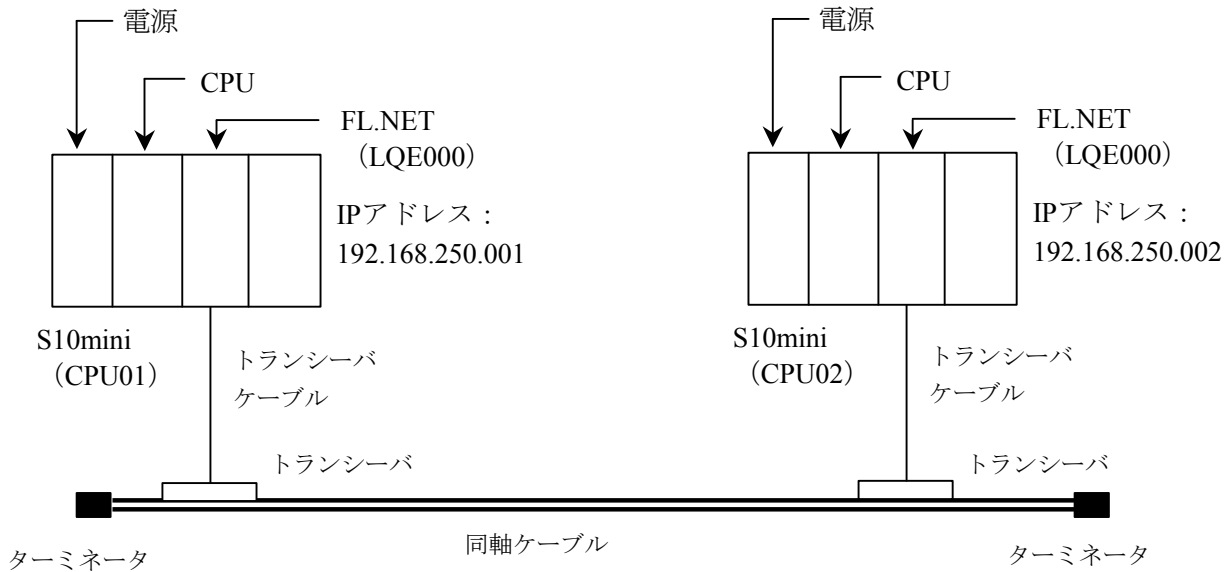
6 利用の手引き

(6) プログラム例

以下に2台のFL.NETモジュールにて透過型メッセージの送信/受信を行うプログラムを示します。

(a) Cモードハンドラによる送受信

[システム構成]



CPU01, CPU02のFL.NETモジュールのMODU No.スイッチは0に設定してください。

システム構成部品一覧

品名	形式	数量	備考
電源	LQV000	2	
CPU	LQP010	2	
FL.NET	LQE000	2	
トランシーバケーブル	HBN-TC-100	2	メーカー：日立電線（株）
トランシーバ	HLT-200TB	2	メーカー：日立電線（株）
同軸ケーブル	HBN-CX-100	1	メーカー：日立電線（株）
ターミネータ	HBN-T-NJ	2	メーカー：日立電線（株）
12V電源	HK-25A-12	2	メーカー：デジテック（株）

[FL.NETモジュール設定]

CPU01, CPU02の各FL.NETモジュールの設定を以下に示します。

各FL.NETモジュールの自ノード設定

設定項目	CPU01設定内容	CPU02設定内容
ノード番号	1	2
領域1アドレス (設定値)	0x000	0x004
領域1ワード数	0x004	0x004
領域1アドレス (PCs割り付け)	RW000	RW040
領域2アドレス (設定値)	0x0000	0x0040
領域2ワード数	0x0040	0x0040
領域2アドレス (PCs割り付け)	DW000	DW040
自ノード状態 (PCs割り付け)	RW080	RW080
透過型受信起動タスク	—————	—————
透過型受信起動要因	—————	—————
透過型受信フラグ	—————	RW100

各FL.NETモジュールの他ノード設定

設定項目	CPU01設定内容		CPU02設定内容	
	ノード1	ノード2	ノード1	ノード2
領域1アドレス	自ノード番号 のため、設定 する必要はあ りません。	RW040	RW000	自ノード番号 のため、設定 する必要はあ りません。
領域1ワード数		0x004	0x004	
領域2アドレス		DW040	DW000	
領域2ワード数		0x0040	0x0040	
FAリンク状態		—————	—————	
上位層状態		—————	—————	

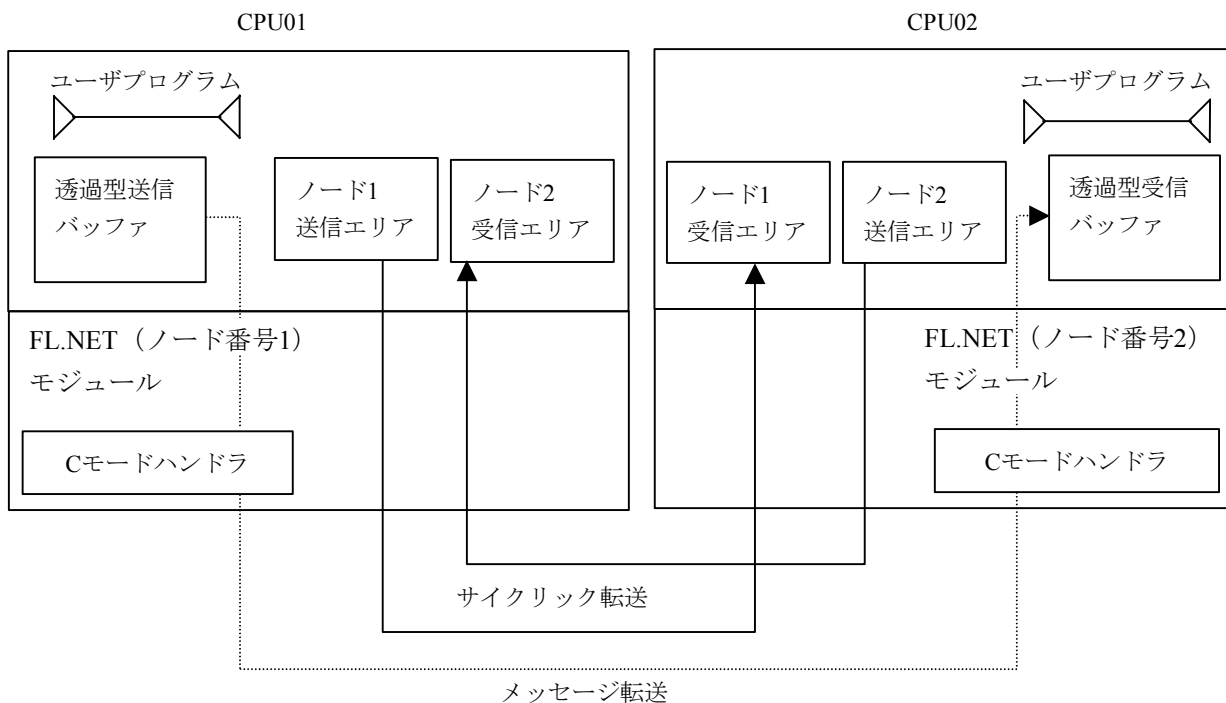
6 利用の手引き

[プログラム構成]

プログラム構成を以下に示します。ノード番号1モジュール（CPU01のFL.NETモジュール）とノード番号2モジュール（CPU02のFL.NETモジュール）を接続し、FL.NETモジュール間でサイクリック転送を実施させます（サイクリック転送はFL.NETモジュールが実施します。ユーザは意識する必要はありません）。

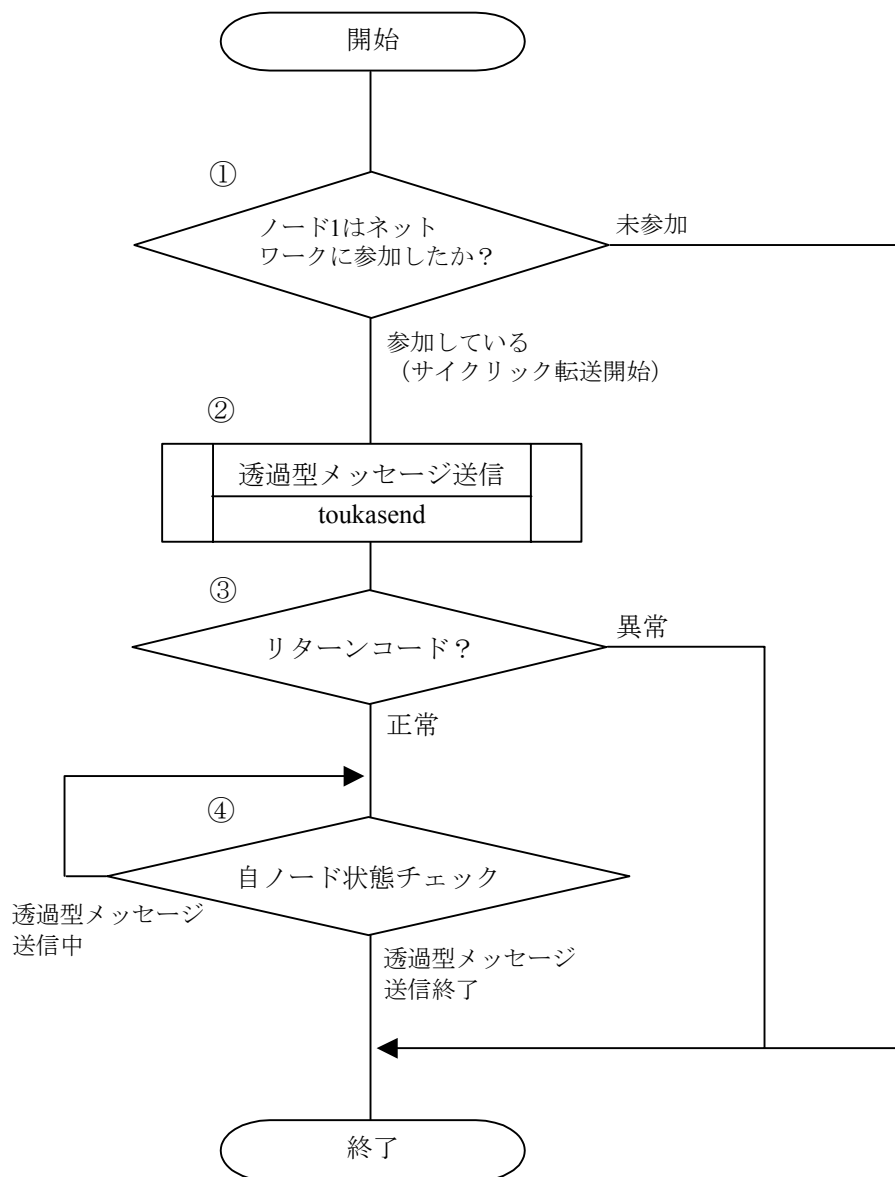
その後、ノード番号1モジュールからノード番号2モジュールに対して、1024バイトの透過型メッセージ（TCD番号=1000）を送信します。ノード番号2モジュールはノード番号1からの透過型メッセージを受信します。

このプログラムを動作させる場合、必ずCPU01からユーザプログラムを起動させてください。



項目 \ CPU		CPU01	CPU02
機能		透過型メッセージ送信	透過型メッセージ受信
送信バッファ	アドレス	0x1E6000	—
	バイト数	1024	—
受信バッファ	アドレス	—	0x1E6000
	バイト数	—	1024
Cモードハンドラ	toukaread()	0XD74178	0XD74178
先頭アドレス	toukasend()	0XD7417E	0XD7417E

[CPU01側プログラムフローチャート]



- ① 自ノード状態転写エリアをチェックし、その内容が0x0080になっていた場合、自ノードがネットワークに参加しているとみなします。
- ② ノード番号2モジュールに対して、透過型メッセージの送信要求を発行します。
- ③ Cモードハンドラからリターンコードをチェックし、正常に要求を受け付けられたかを判定します（リターンコードが0の場合は正常、-1の場合は異常とみなします）。
- ④ 自ノード状態転写エリアをチェックし、 2^{15} ビット（0x8000）がOFFするまで待ちます（自ノード状態フラグの 2^{15} ビットはユーザ要求処理中ビットです）。

6 利用の手引き

[CPU01側のC言語プログラム例]

```
#define TOUKA_SEND      0xD7417EL      /* toukasend() 先頭アドレス(メイン) */
#define SBUFADR        0x1E6000L      /* 送信バッファのアドレス */
#define PARADDR        0x1E5000L      /* 入力パラメータの先頭アドレス */
#define RW080          0x0E0C10L      /* 自ノード状態転写エリア(RW080) */
#define RW090          0x0E0C12L      /* 透過型メッセージ送信エラーコードエリア */

struct ToukaSend_p {
    long          node;                /* 送信先ノード番号 */
    unsigned short *Erradr;           /* エラーコード格納アドレス */
    unsigned char *dataadr;          /* 送信データ先頭アドレス */
    unsigned long  datasiz;           /* 送信データバイト数 */
    unsigned long  TcdNo;             /* 透過型メッセージTCD番号 */
};

/*****
/* task2:送信(CPU01) */
*****/

main()
{
    register long   ( *toukasend )();
    long           rtn;
    struct          ToukaSend_p   *send;
    unsigned short *nodeflg;

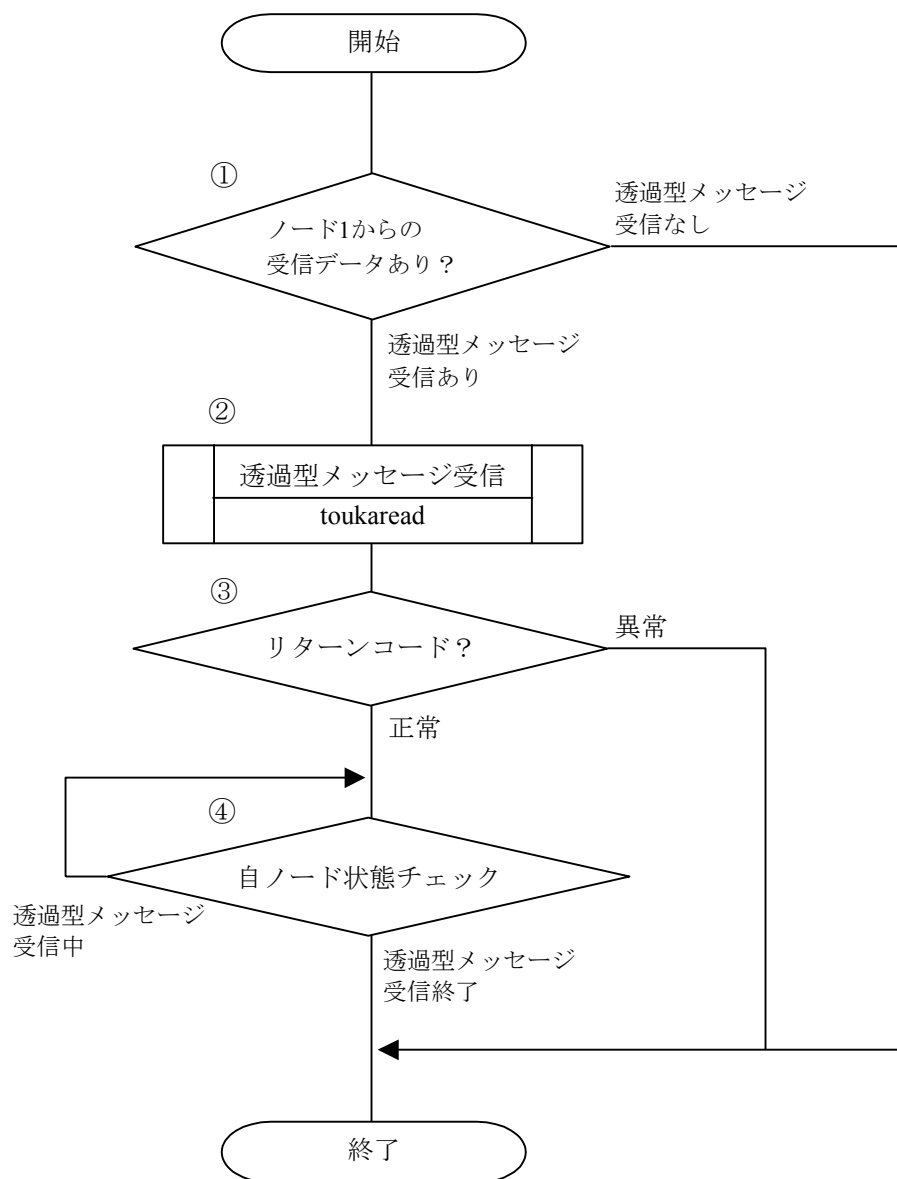
    nodeflg = ( unsigned short *)RW080;
    toukasend = ( long(*) ( ))TOUKA_SEND;
    send      = ( struct ToukaSend_p *)PARADDR;
    if( !( *nodeflg & 0x0080)){                /* 自ノード状態チェック */
        return;
    }

    send->node      = 0x00000002;                /* 送信先ノード番号 */
    send->Erradr    = (unsigned short*)RW090;    /* エラーコード格納アドレス */
    send->dataadr   = (unsigned char*)SBUFADR;   /* 送信データ先頭アドレス */
    send->datasiz   = 1024;                      /* 送信データバイト数 */
    send->TcdNo    = 1000;                      /* 透過型メッセージTCD番号 */
    rtn = ( toukasend )( send );                /* 透過型メッセージ送信 */

    if( rtn != 0){                              /* リターンコードチェック */
        return;
    }

    while( 1 ){
        if( !( *nodeflg & 0x8000)){            /* メッセージ処理終了待ち */
            break;
        }
    }
}
```

[CPU02側プログラムフローチャート]



- ① 透過型受信フラグエリアをチェックし、その内容が0x4000であり、かつ、自ノード状態転写エリア内の 2^{14} ビットがONしていた場合、ノード1からの透過型メッセージありとみなします（透過型受信フラグの先頭アドレス 2^{14} ビットがノード1用受信ビットです）。
- ② 透過型メッセージの受信要求を発行します。
- ③ Cモードハンドラからリターンコードをチェックし、正常に要求を受け付けられたかを判定します（リターンコードが0の場合は正常、-1の場合は異常とみなします）。
- ④ 自ノード状態転写エリアをチェックし、 2^{15} ビット（0x8000）がOFFするまで待ちます（自ノード状態フラグの 2^{15} ビットはユーザ要求処理中ビットです）。

6 利用の手引き

[CPU02側のC言語プログラム例]

```
#define TOUKA_READ      0xD74178L      /* toukaread() 先頭アドレス(メイン) */
#define RBUFADR        0x1E6000L      /* 受信バッファのアドレス */
#define PARADDR        0x1E5000L      /* 入力パラメータの先頭アドレス */
#define RW080          0x0E0C10L      /* 自ノード状態転写エリア(RW080) */
#define RW090          0x0E0C12L      /* 透過型メッセージ送信エラーコードエリア */
#define RW100          0x0E0C20L      /* 透過型受信フラグエリア */

struct ToukaRead_p {
    long      node;                    /* 送信先ノード番号 */
    unsigned short *Erradr;           /* エラーコード格納アドレス */
    unsigned char *dataadr;           /* 受信データ先頭アドレス */
    unsigned long datasiz;            /* 受信データバイト数 */
};

/*****
/* task3:受信(CPU02) */
*****/

main()
{
    register long  (*toukaread)();
    long          rtn;
    struct        ToukaRead_p  *read;
    unsigned short *nodeflg, *rcvarea;

    rcvarea = ( unsigned short *)RW100;      /* 透過型受信フラグエリア */
    nodeflg = ( unsigned short *)RW080;
    toukaread = ( long(*)())TOUKA_READ;
    read      = ( struct ToukaRead_p *)PARADDR;
    if( ( *rcvarea != 0x4000) ||            /* 透過型受信フラグエリアおよび */
        !( *nodeflg & 0x4000)){           /* 自ノード状態をチェック */
        return;
    }

    read->node      = 0x00000001;           /* 送信元ノード番号 */
    read->Erradr     = ( unsigned short *)RW090; /* エラーコード格納アドレス */
    read->dataadr    = ( unsigned char *)RBUFADR; /* 受信データ先頭アドレス */
    read->datasiz    = 1024;                /* 受信データバイト数 */
    rtn = ( toukaread )( read );           /* 透過型メッセージ受信 */

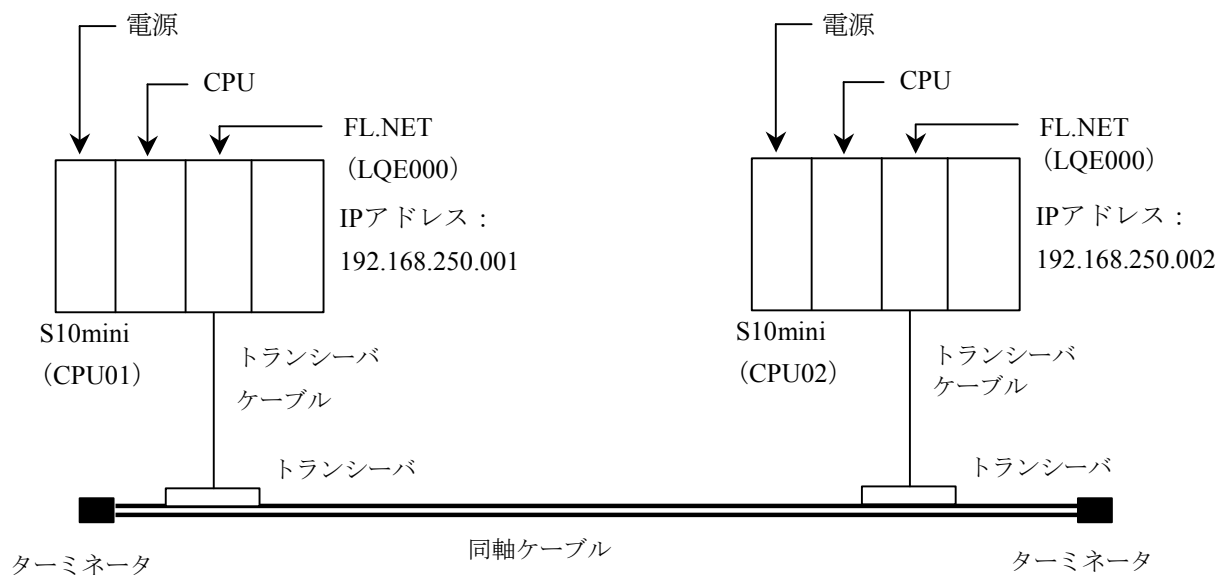
    if( rtn != 0){                          /* リターンコードチェック */
        return;
    }

    while( 1 ){
        if( !( *nodeflg & 0x8000)){         /* メッセージ処理終了待ち */
            break;
        }
    }
}
```

(注) ここに記述したユーザプログラムは、理解しやすいことを目的に書かれています。
実用プログラムでは、メッセージ発行後、ユーザ要求処理中ビットがOFFした後で、エラーコード格納アドレス内に書き込まれるエラーコードをチェックしてください。

(b) 演算ファンクションによる送受信

[システム構成]



CPU01, CPU02のFL.NETモジュールのMODU No.スイッチは0に設定してください。

システム構成品一覧

品名	形式	数量	備考
電源	LQV000	2	
CPU	LQP010	2	
FL.NET	LQE000	2	
トランシーバケーブル	HBN-TC-100	2	メーカー：日立電線（株）
トランシーバ	HLT-200TB	2	メーカー：日立電線（株）
同軸ケーブル	HBN-CX-100	1	メーカー：日立電線（株）
ターミネータ	HBN-T-NJ	2	メーカー：日立電線（株）
12V電源	HK-25A-12	2	メーカー：デンソーエムダ（株）

6 利用の手引き

[FL.NETモジュール設定]

CPU01, CPU02の各FL.NETモジュールの設定を以下に示します。

各FL.NETモジュールの自ノード設定

設定項目	CPU01設定内容	CPU02設定内容
ノード番号	1	2
領域1アドレス (設定値)	0x000	0x004
領域1ワード数	0x004	0x004
領域1アドレス (PCs割り付け)	RW000	RW040
領域2アドレス (設定値)	0x0000	0x0040
領域2ワード数	0x0040	0x0040
領域2アドレス (PCs割り付け)	DW000	DW040
自ノード状態 (PCs割り付け)	RW080	RW080
透過型受信起動タスク	_____	_____
透過型受信起動要因	_____	_____
透過型受信フラグ	_____	RW100

各FL.NETモジュールの他ノード設定

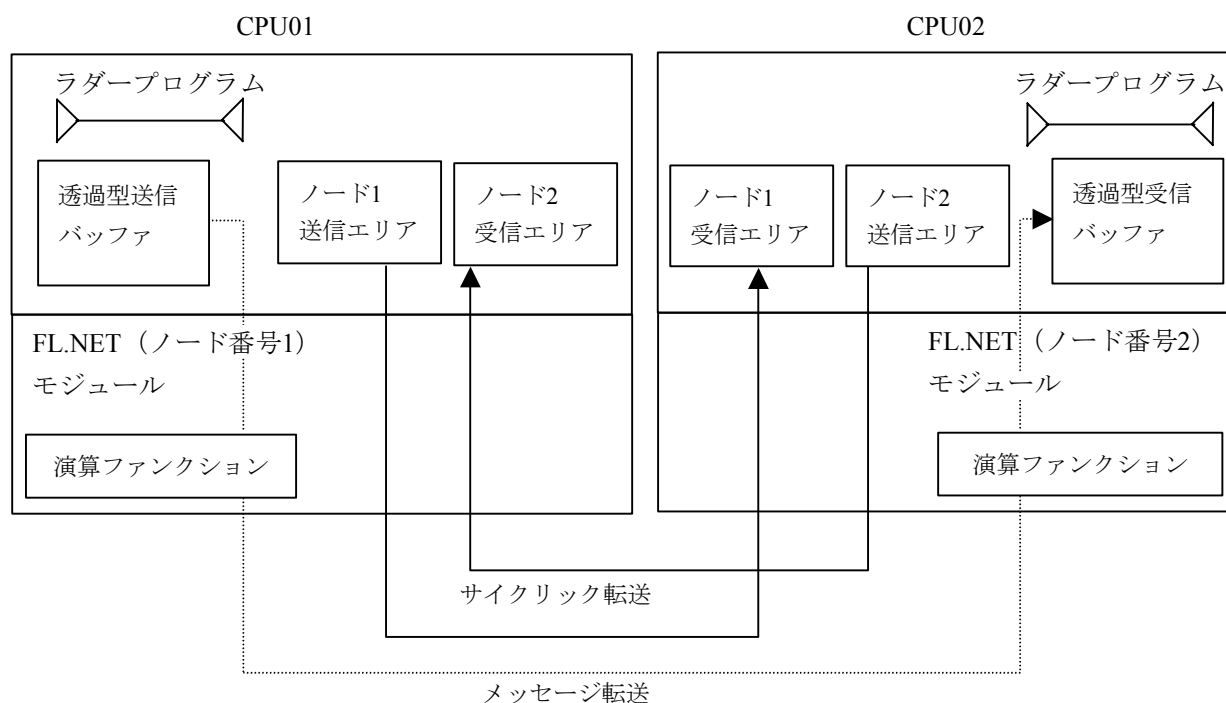
設定項目	CPU01設定内容		CPU02設定内容	
	ノード1	ノード2	ノード1	ノード2
領域1アドレス	自ノード番号 のため、設定 する必要はあ りません。	RW040	RW000	自ノード番号 のため、設定 する必要はあ りません。
領域1ワード数		0x004	0x004	
領域2アドレス		DW040	DW000	
領域2ワード数		0x0040	0x0040	
FAリンク状態		_____	_____	
上位層状態		_____	_____	

[プログラム構成]

プログラム構成を以下に示します。ノード番号1モジュール（CPU01のFL.NETモジュール）とノード番号2モジュール（CPU02のFL.NETモジュール）を接続し、FL.NETモジュール間でサイクリック転送を実施させます（サイクリック転送はFL.NETモジュールが実施します。ユーザは意識する必要はありません）。

その後、CPU01側のラダープログラムにてR0A0の立ち上がりを検出したとき、ノード番号1モジュールからノード番号2モジュールに対して、1024バイトの透過型メッセージ（TCD番号=1000）を送信します。ノード番号2モジュールはノード番号1からの透過型メッセージを受信します。

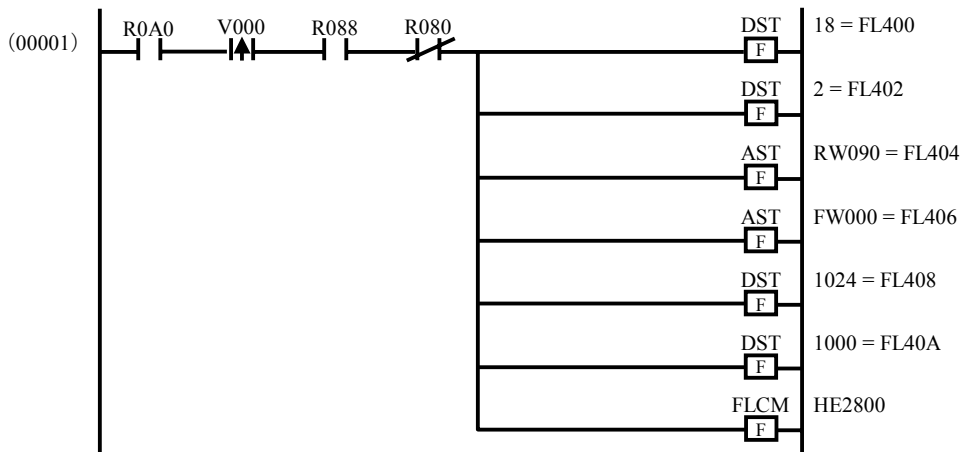
このラダープログラムを使用する場合には、S10miniをRUN状態にしてください。



項目		CPU	
		CPU01	CPU02
機能		透過型メッセージ送信	透過型メッセージ受信
送信バッファ	アドレス	FW000～FW1FF	—
	バイト数	1024バイト	—
受信バッファ	アドレス	—	FW000～FW203
	バイト数	—	1024バイト+4ワード
演算ファンクション	FLCM	0xD74100	0xD74100

6 利用の手引き

[CPU01側ラダープログラム]



R0A0がONしたタイミングで、動作します。

そのとき、自ノード状態フラグ内のネットワーク参加状態ビット (R088) とユーザ要求処理中ビット (R080) をチェックしています (ネットワーク参加状態ビットがONし、かつユーザ要求処理中ビットがOFFのとき、メッセージ処理が可能となります)。

各演算ファンクションにて、それぞれ透過型メッセージ送信を行うためのパラメータを指定しています。

- DST 18 = FL400

メッセージ伝送サービス番号に18を指定し、透過型メッセージ送信要求を指定しています。

- DST 2 = FL402

相手局 (CPU02) を指定しています。

- AST RW090 = FL404

エラーコード格納アドレス (RW090の実アドレス) を指定しています。

- AST FW000 = FL406

送信バッファアドレス (FW000の実アドレス) を指定しています。

- DST 1024 = FL408

送信語数を指定しています。

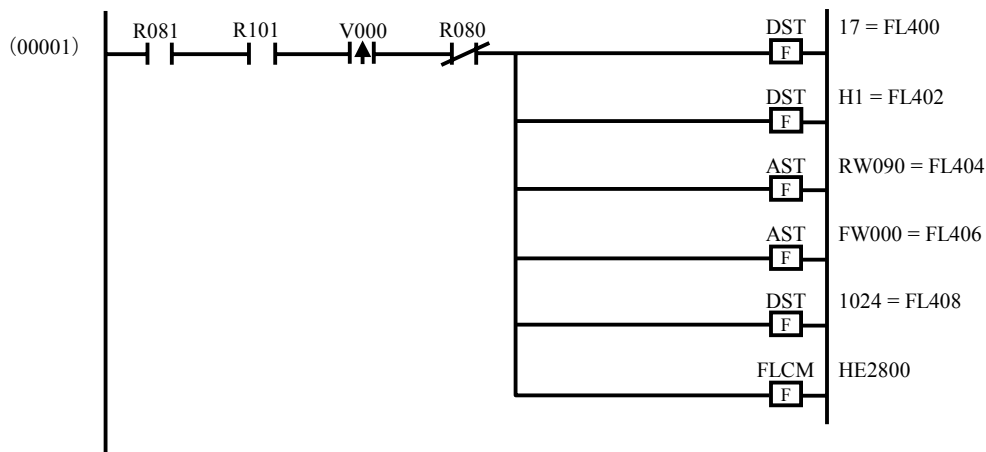
- DST 1000 = FL40A

送信する透過型メッセージのTCD番号 (1000) を指定しています。

- FLCM HE2800

演算ファンクション (FLCM) に、パラメータを格納したアドレスを指定しています (HE2800 = FW400の実アドレス)。FL.NETモジュールへのメッセージ要求 (透過型メッセージ送信) を発行します。

[CPU02側ラダープログラム]



自ノード状態フラグの透過型メッセージ受信ビット（R081）がONし、透過型受信フラグエリア内のノード1用受信ビット（R101）がそれぞれONしたタイミングで動作します。

そのとき、自ノード状態フラグ内のユーザ要求処理中ビット（R080）をチェックしています（ユーザ要求処理中ビットがOFFのとき、メッセージ処理ができるようになります）。

各演算ファンクションにて、それぞれ透過型メッセージ送信するためのパラメータを指定しています。

- DST 17 = FL400
メッセージ伝送サービス番号に17を指定し、透過型メッセージ受信要求を指定しています。
- DST 1 = FL402
相手局（CPU01）からの透過型メッセージを指定しています。
- AST RW090 = FL404
エラーコード格納アドレス（RW090の実アドレス）を指定しています。
- AST FW000 = FL406
受信バッファアドレス（FW000の実アドレス）を指定しています。
- DST 1024 = FL408
受信語数を指定しています。
- FLCM HE2800
演算ファンクション（FLCM）に、パラメータを格納したアドレスを指定しています（HE2800 = FW400の実アドレス）。FL.NETモジュールへのメッセージ要求（透過型メッセージ送信）を発行します。

(注) ここに記述したラダープログラムは、理解しやすいことを目的に書かれています。

実用プログラムでは、メッセージ発行後、ユーザ要求処理中ビットがOFFした後で、エラーコード格納アドレス内に書き込まれるエラーコードをチェックしてください。

6 利用の手引き

6.4.6 管理テーブルの使い方

FL.NETモジュールでは、他ノードとの通信状態を、各種管理テーブルで管理しています。

この管理テーブルを参照することで、他ノードとの通信状態を確認できます。

ただし、FL.NETモジュール内の管理テーブルを参照するには、設定ツール [FL.NET For Windows®] を使用してください。

各表示での詳しい操作方法は、「ソフトウェアマニュアル オプション FL.NET For Windows® (マニュアル番号 SAJ-3-139)」を参照してください。

(1) 自ノード管理テーブルを参照するには

自ノード管理テーブルを参照するには、設定ツール [FL.NET For Windows®] にて、参加ノード情報を表示させます。その際、ネットワークに参加中のノード番号一覧が表示されますので、自ノードの表示を指定してください。

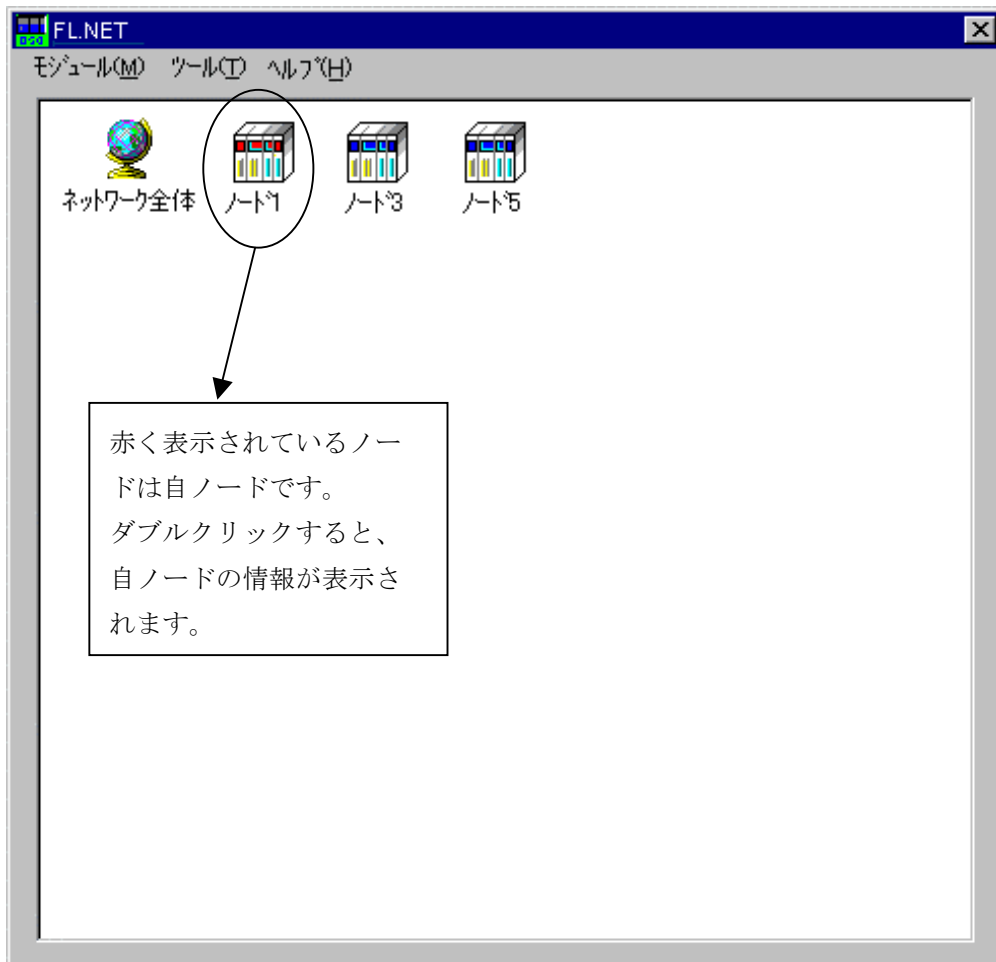


図 6-47 参加ノード番号一覧画面1

上位層を示すフラグが入ります。フラグの内容は次ページを参照してください。

指定のエリアにFAリンク状態、自ノード状態を示すフラグが入ります。フラグの内容は次ページを参照してください。

自ノード情報

ノード番号 [1~254]

領域1アドレス [0x000~0x1FF]

領域1ワート数 [0x000~0x200]

領域2アドレス [0x000~0x1FFF]

領域2ワート数 [0x000~0x2000]

上位層状態

トークン監視タイムアウト時間 [msec]

最小許容フレーム間隔 [0~50×100 μsec] [×100 μsec]

ヘンダ名称

メカ型式

ノード名称 [10文字]

プロトコルバージョン

FAリンク状態

自ノード状態

透過型受信起動タスク [0~127]

透過型受信起動要因 [0~16]

透過型受信フラグエリア

現在値

[msec]

[×100 μsec]

書き換え値

[×100 μsec]

PCs割付

モニタ開始(M)

FL-net参入

OK

ノードデータ表示

FL-net離脱

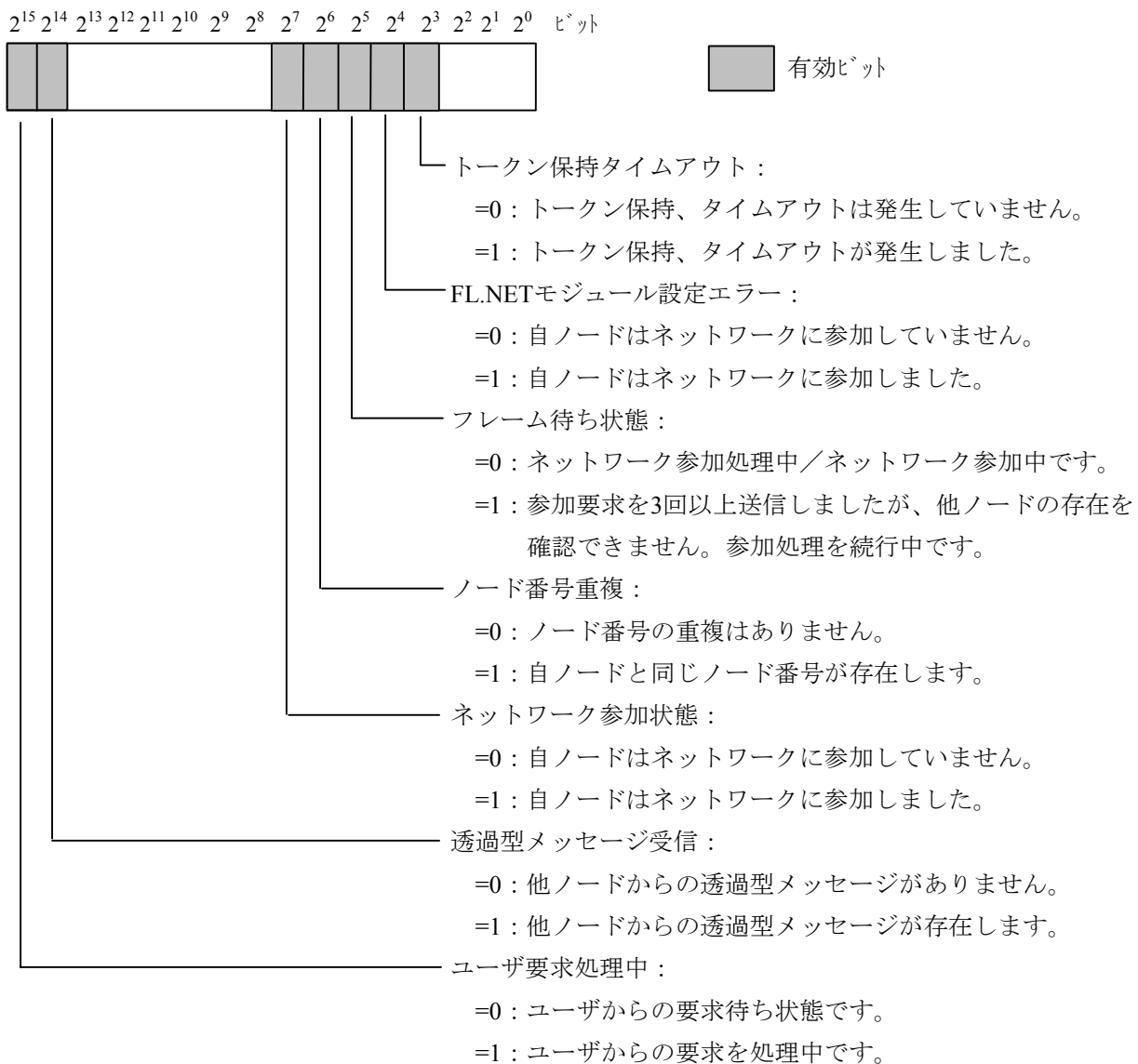
キャンセル

図6-48 自ノード情報（自ノード管理テーブル）表示画面

6 利用の手引き

[自ノード状態フラグの内容]

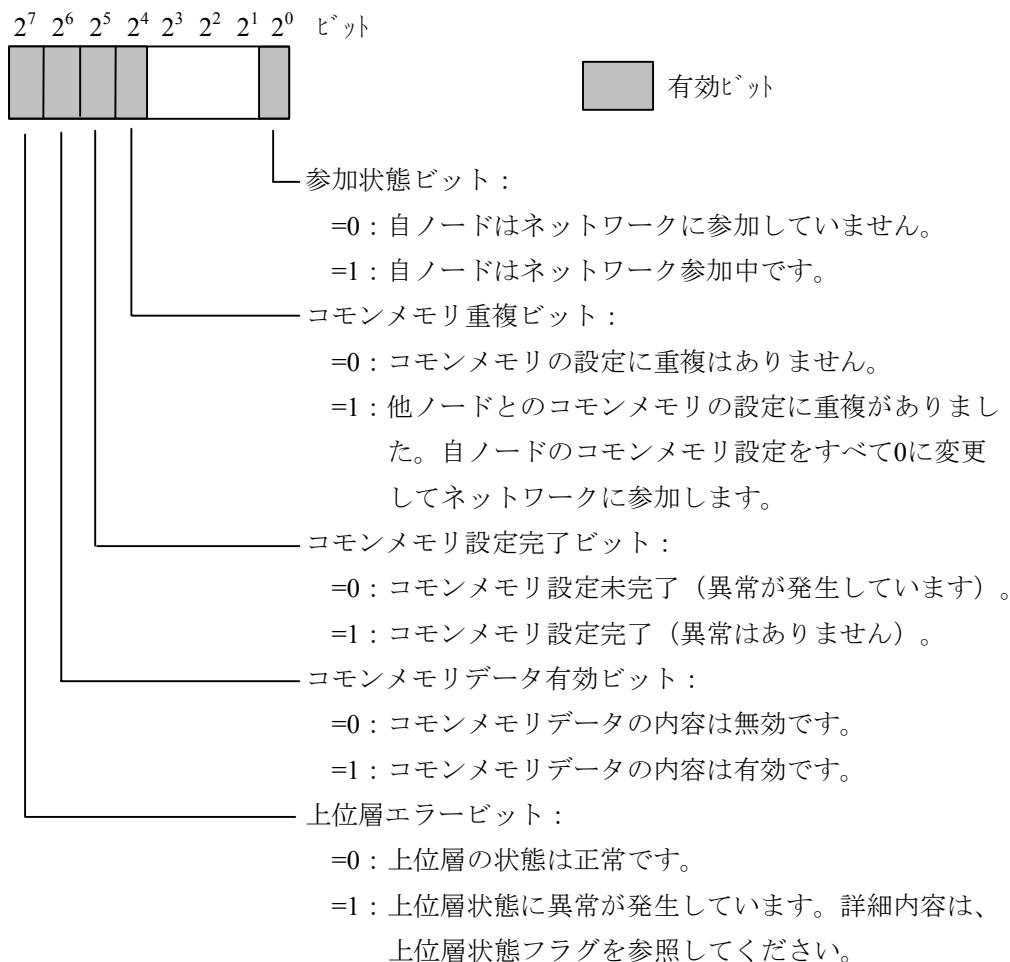
FL.NETモジュールでは、自ノード状態フラグのビット割り付けは次のようになっています。



(注) 上記の自ノード状態は、S10mini上に転写された場合の内容です。設定ツール [FL.NET For Windows®] の自ノード情報ダイアログボックスでは、下位バイト (2⁷~2⁰ビット) のみ参照できます。

[FAリンク状態フラグ（FAリンクステータス）の内容]

FAリンクステータスフラグのビット割り付けは次のようになっています。



6 利用の手引き

[上位層状態フラグの内容]

上位層状態フラグのビット割り付けは次のようになっています。

2¹⁵ 2¹⁴ 2¹³ 2¹² 2¹¹ 2¹⁰ 2⁹ 2⁸ 2⁷ 2⁶ 2⁵ 2⁴ 2³ 2² 2¹ 2⁰ ビット



■ 有効ビット

上位層エラーコード (12ビット構成) :

上位層にて発生しているエラーのエラーコードが表示されます。
このFL.NETモジュールでのエラーコード表を以下に示します。

エラーコード	内容
0x000	上位層 (S10mini) は正常です。
0x0FF	上位層 (S10mini) にてCPUダウンが発生しています。

上位層エラー状態ビット (2ビット構成) :

上位層エラー発生時、その状態を示します。

2 ¹⁴ ビット	2 ¹³ ビット	内容
OFF	OFF	上位層は正常です。
OFF	ON	上位層に継続可能な異常が発生しています (ARARM状態)。 サイクリックデータとメッセージデータの内容は保証されます。
ON	OFF	上位層に継続不可能な異常が発生しています (WARNING状態)。
ON	ON	サイクリックデータとメッセージデータの内容は保証されません。

このFL.NETモジュールでは、上位層異常発生時には、2¹⁴ビットと2¹³ビットをONします。

上位層動作状態ビット :

=0 : 上位層 (S10mini) はSTOP中 (停止中) です。

=1 : 上位層 (S10mini) はRUN中 (動作中) です。

(2) 参加ノード管理テーブルを参照するには

設定ツール [FL.NET For Windows®] にて参加ノード情報を表示させることで、ネットワークに参加中のノード番号一覧と、各参加ノード番号の参加ノード管理テーブルの内容を表示する参加他ノード情報ダイアログボックスを参照できます。

ただし、設定ツール [FL.NET For Windows®] は、ネットワークに参加しているノード番号の参加ノード管理テーブルの内容しか参照できません。

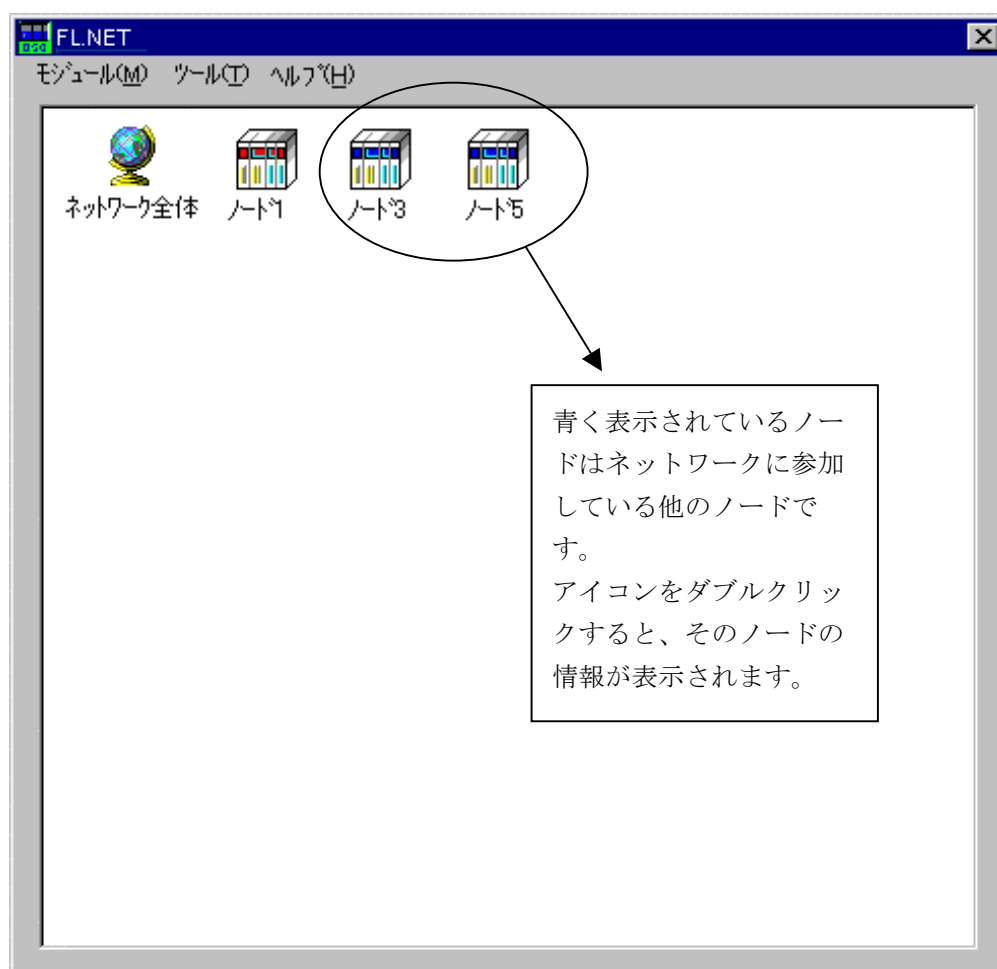


図 6-49 参加ノード番号一覧画面2

6 利用の手引き

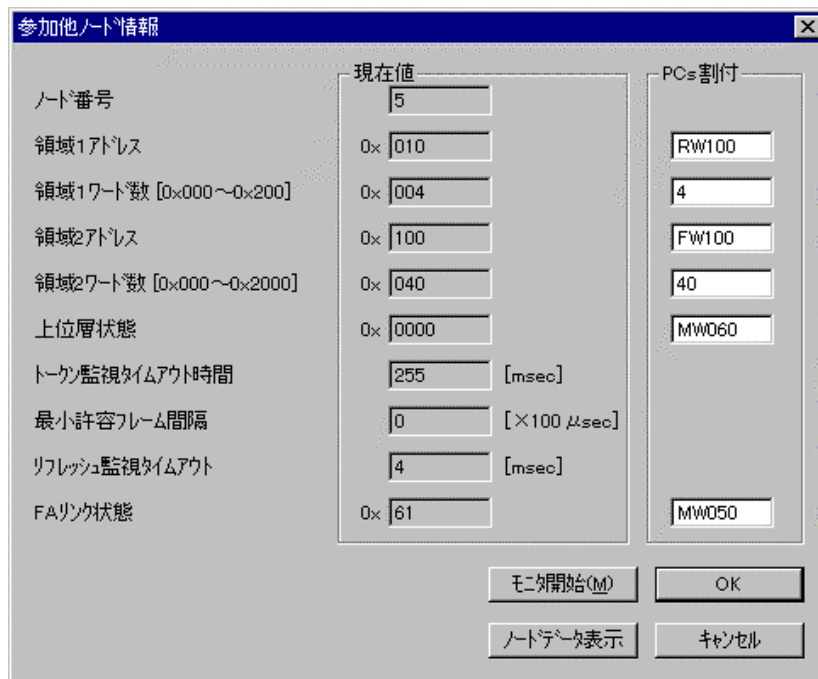


図 6-50 参加他ノードリンク情報表示画面

各ノードの上位層状態、FAリンク状態（FAリンクステータス）のビット割り付けに関しては、各ノードのマニュアルを参照してください。

(3) ネットワーク管理テーブルを参照するには

設定ツール [FL.NET For Windows®] のネットワーク状態画面にて、ネットワーク管理テーブルの内容を参照できます。

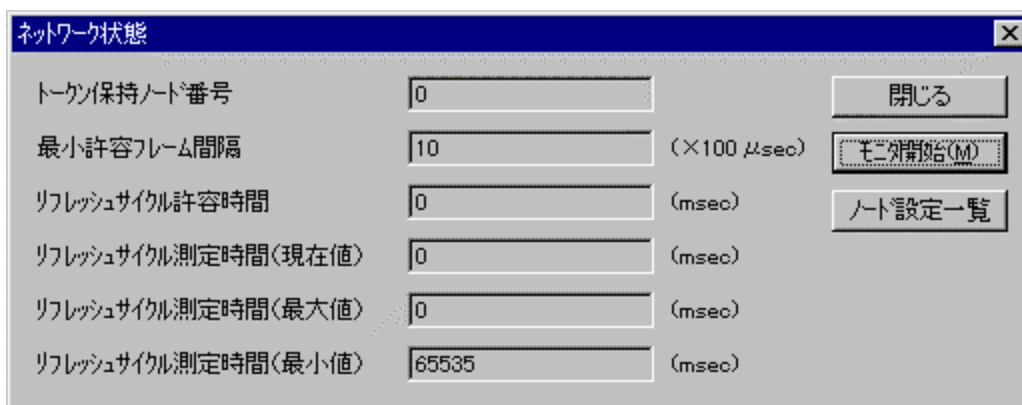


図 6-51 ネットワーク状態（ネットワーク管理テーブル）表示画面

各項目の内容を以下に示します。

表示項目	表示内容
トークン保持ノード番号	現在トークンを保持しているノード番号です。
最小許容フレーム間隔	ネットワーク上の最小許容フレーム間隔です。 ネットワークを構成しているノード内での最大値となります。
リフレッシュサイクル許容時間	トークンがネットワークを1周する時間を1.2倍した値です。
リフレッシュサイクル測定時間 (現在値)	トークンがネットワークを1周する時間の現在の時間、過去の最大時間、過去の最小時間を表示します。
リフレッシュサイクル測定時間 (最大値)	
リフレッシュサイクル測定時間 (最小値)	
リフレッシュサイクル測定時間 (最小値)	

6. 4. 7 FL.NETモジュールの通信時間

S10miniのFL.NETの通信性能を以下に示します。ただし、接続する相手や語数により変わってきますので下記条件を前提としています。

- (条件) ・ S10mini FL.NET同士を接続した際の通信性能です。
- ・ 接続したFL.NETの設定 (通信語数) はすべて同じ語数C (C1+C2)

なお、計算式で使用している記号の意味を以下に示します。

C : 領域1、2の通信語数 (ワード)
 C1 : 領域1通信語数 (ワード)
 C2 : 領域2通信語数 (ワード)
 T : 通信語数を512で割った余り。
 ただし、割り切れた場合はT=512とします (T=1~512ワード)。
 RCT : リフレッシュサイクル (ms)
 SCT : シーケンスサイクル (ms)
 MS : メッセージ送信語数 (バイト)
 MR : メッセージ受信語数 (バイト)

- トークン保持時間 (ms)
 トークン保持時間 (ms) > 最小許容フレーム間隔 (ms) の場合 (最小許容フレーム間隔の方が大きい場合は、最小許容フレーム間隔がトークン保持時間となります)。

C ≤ 512の場合 ---- $1.51 + 0.00371 * C$

C > 512の場合 ---- $1.51 + 0.01164 * C1 + 0.01006 * C2 - 0.01055 * T$

6 利用の手引き

- リフレッシュサイクル測定時間 (ms)

各ノードのトークン保持時間の合計です。ノード1から3までの3台が接続されている場合は、ノード1のトークン保持時間+ノード2のトークン保持時間+ノード3のトークン保持時間（トークン保持時間の計算は前項目を参照してください。）

- サイクリック転送スループット (ms)

$2 \times RCT + 2 \times SCT$ （ラダーの場合のワーストケースです。）

- ワードブロックリード (ms)

演算ファンクション（サブルーチン）を起動してから、データを受信するまでの時間です。

$C \leq 512$ の場合

$3.1 + 0.0177 \times C1 + 0.0169 \times C2 + 2RTC +$ （自ノードから送信先ノード手前までのトークン保持時間の累計）

$C > 512$ の場合

$3.1 + 0.0217 \times C1 + 0.0201 \times C2 - 0.0032 \times T + 2RTC +$ （自ノードから送信先ノード手前までのトークン保持時間の累計）

- ワードブロックライト (ms)

演算ファンクション（サブルーチン）を起動してから、相手からACKを受信するまでの時間です。

$C \leq 512$ の場合

$4.32 + 0.0106 \times C1 + 0.0118 \times C2 + 2RTC +$ （自ノードから送信先ノード手前までのトークン保持時間の累計）

$C > 512$ の場合

$4.32 + 0.0138 \times C1 + 0.0148 \times C2 - 0.0032 \times T + 2RTC +$ （自ノードから送信先ノード手前までのトークン保持時間の累計）

- 透過型メッセージ (ms)

メッセージ作成時間 (ms) = $0.02 + (0.0007 \times MS)$ > 最小許容フレーム間隔 (ms) の場合（最小許容フレーム間隔の方が大きい場合は、最小許容フレーム間隔がメッセージ作成時間となります）。

1対1送信要求処理時間

リフレッシュサイクル測定時間 $\times 2$ + 自ノードのトークン保持時間 + メッセージ作成時間 + $1.272 + (0.00096 \times MS) + (0.0013 \times MR)$

1対N送信要求処理時間

リフレッシュサイクル測定時間 + メッセージ作成時間 + $0.742 + (0.00096 \times MS)$

（リフレッシュサイクル測定時間、トークン保持時間の計算は前項目を参照してください。）

6. 4. 8 通信ログの使い方

設定ツール [FL.NET For Windows®] で、FL.NETモジュール内のRAS情報（通信ログ）を参照できます。

The screenshot shows a dialog box titled "RAS情報" (RAS Information) with a close button (X) in the top right corner. The dialog is divided into several sections, each containing a list of error counters with their respective values (all are 0).

ログデータ	値
送受信に関するログ	
送信回数	0
ソケット以下の送信エラー回数	0
イーサネットの送信エラー回数	0
受信回数	0
ソケット以下の受信エラー回数	0
イーサネットの受信エラー回数	0
フレーム送受信回数	
トークン送信回数	0
サイクリック送信回数	0
1:1メッセージ送信回数	0
1:Nメッセージ送信回数	0
トークン受信回数	0
サイクリック受信回数	0
1:1メッセージ受信回数	0
1:Nメッセージ受信回数	0
サイクリック伝送のエラー回数	
サイクリックエラー・トル数	0
アドレス・サイズエラー	0
CBNIエラー	0
TBNIエラー	0
BSIZEエラー	0
メッセージ伝送のエラー回数	
再送回数	0
再送オーバー回数	0
受信エラー回数	0
通番バージョンエラー回数	0
通番再認識回数	0
ACK関連のエラー回数	
ACKエラー回数	0
バージョンエラー回数	0
通番エラー回数	0
ノード番号エラー回数	0
TCDエラー回数	0
トークン関連のエラー回数	
トークン多重化認識回数	0
トークン破棄回数	0
トークン再発行回数	0
トークン保持タイムアウト回数	0
トークン監視タイムアウト回数	0
ノードの状態	
トータル稼働時間	0
フレーム待ち状態の回数	0
加入回数	0
自己離脱回数	0
スキップ離脱回数	0
他ノード離脱認識回数	0

On the right side of the dialog, there are four buttons: "閉じる" (Close), "モニタ開始(M)" (Start Monitor), "クリア(C)" (Clear), and a small "X" button in the top right corner.

図 6-52 RAS情報ダイアログボックス

6 利用の手引き

以下に、ログ情報（RAS情報）の表示内容を示します。

	項 目	表 示 内 容
送受信に関するログ	送信回数	送信要求を行ったフレーム数
	ソケット以下の送信エラー回数	ソケット部における送信エラー回数
	イーサネット以下の送信エラー回数	未使用
	受信回数	ソケット部から受けたフレーム数
	ソケット以下の受信エラー回数	ソケット部における受信エラーおよび異常フレーム受信回数
	イーサネット以下の受信エラー回数	未使用
フレーム送受信回数	トークン送信回数	トークンを送信した回数
	サイクリック送信回数	トークンを含まないサイクリックフレームを送信した回数
	1:1メッセージ送信回数	1対1のメッセージを送信した回数
	1:Nメッセージ送信回数	1対Nのメッセージを送信した回数
	トークン受信回数	トークンを受信した回数
	サイクリック受信回数	トークンを含まないサイクリックフレームを受信した回数
	1:1メッセージ受信回数	1対1のメッセージを受信した回数
	1:Nメッセージ受信回数	1対Nのメッセージを受信した回数
サイクリック伝送のエラー回数	サイクリックエラートータル数	サイクリック伝送受信におけるエラー回数
	アドレス・サイズエラー	未使用
	CBNエラー	フレームの並びに関するエラー回数
	TBNエラー	フレームの分割数に関するエラー回数
	BFSIZEエラー	フレームのサイズに関するエラー回数
メッセージ伝送のエラー回数	再送回数	メッセージを再送した回数
	再送オーバー回数	メッセージが再送オーバーになった回数
	受信エラー回数	メッセージ伝送受信におけるエラー回数
	通番バージョンエラー回数	メッセージを受信し、通番バージョンエラーと認識した回数
	通番再送認識回数	メッセージを受信し、再送メッセージと認識した回数
ACK関連のエラー回数	ACKエラー回数	ACKに関するエラー回数
	バージョンエラー回数	ACKの通番バージョンの不整合の回数
	通番エラー回数	ACKの通番番号の不整合の回数
	ノード番号エラー回数	未使用
	TCDエラー回数	未使用
トークン関連のエラー回数	トークン多重化認識回数	トークンの多重化を認識した回数
	トークン破棄回数	トークン破棄回数
	トークン再発行回数	トークンの再発行回数
	トークン保持タイムアウト回数	トークン保持タイムアウトが発生した回数
	トークン監視タイムアウト回数	トークン監視タイムアウトが発生した回数
ノードの状態	トータル稼働時間	未使用
	フレーム待ち状態の回数	ネットワーク上に他ノードがなくフレーム待ちになった回数
	加入回数	ネットワークへの加入回数
	自己離脱回数	トークン保持タイムアウトが連続3回または、ネットワーク上に他ノードがいなくなったための離脱回数
	スキップ離脱回数	自ノード宛トークンがスキップされたことによる離脱回数

7 保守・点検

7 保守・点検

7. 1 保守点検項目

S10miniを最適な状態で使用するため、以下に示す点検をしてください。点検は日常または定期的（年2回以上）に行ってください。

- モジュール外観
モジュールのケースにひび、割れなどがいないか点検してください。ケースに異常があると内部回路が破損している場合があります、システム誤動作の原因になります。
- インディケータの点灯状態と表示内容
表示器の状態から特に異常がないか点検してください。
- 取り付けねじ、端子台ねじの緩み
モジュールの取り付けねじ、端子台ねじなどねじ類に緩みがないか点検してください。
緩みがある場合は、増し締めをしてください。ねじに緩みがあるとシステムの誤動作、さらには過熱による焼損の原因になります。
- モジュールの交換
活線状態での交換は、ハードウェア、ソフトウェアの破損につながります。必ず電源を切った状態で交換してください。
- ケーブル被覆の状態
ケーブル被覆に異常がないか点検してください。被覆が剥がれているとシステムの誤動作、感電、さらにはショートによる焼損の原因になります。
- ほこり類の付着状態
モジュールにほこり類が付着していないか点検してください。ほこりが付着しているときには、掃除機などで清掃してください。ほこりが付着していると内部回路がショートし、焼損の原因になります。
- 電源電圧の状態
モジュールの電源、外部供給電源が規定値の範囲内であるか点検してください。電源電圧が定格を外れると、システム誤動作の原因になります。



注 意

静電気によりモジュールが破損する恐れがあります。作業する前に、人体の静電気を放電してください。

7. 1. 1 モジュールの交換、増設

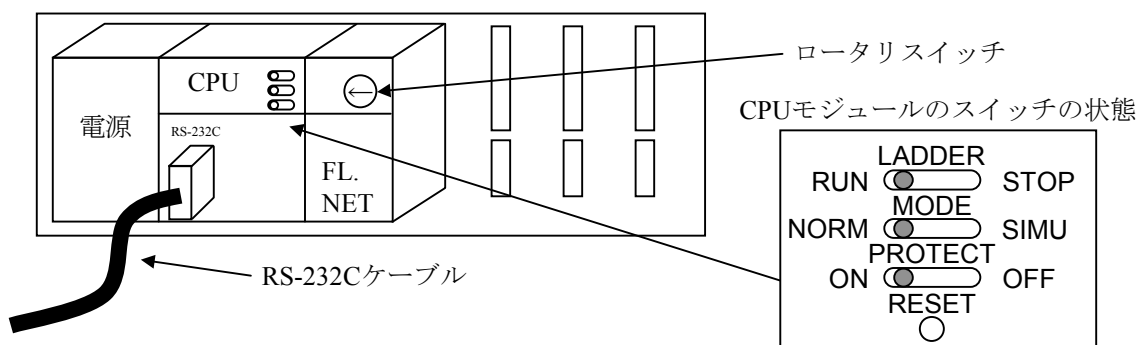
● 交換前準備品

- ① パソコン (Hitachi S10 FL.NETシステムツール組み込み済み)
- ② RS-232Cケーブル (ET.NET使用の場合、10BASE-T)
- ③ FL.NETモジュール (LQE000)
- ④ 交換対象モジュールのパラメータ値 (パラメータが読み出せない場合に使用します。)
- ⑤ オプションモジュールにET.NETが実装されている場合は、通信種類をET.NETにすることができます。

「ユーザーズマニュアル オプション ET.NET (LQE520) (マニュアル番号 SVJ-1-103)」の「2. 1 各部の名称と機能」、「3. 2 モジュールの実装」を参照してください。

● 交換手順

- ① 実装されているFL.NETモジュール前面のロータリスイッチの設定を記録します (MODU No.)。
- ② CPUモジュール前面のスイッチの状態を記録します (LADDER, MODE, PROTECT)。



- ③ パソコンとCPUモジュールをRS-232Cケーブルで接続します。
- ④ Hitachi S10 FL.NETシステムツールを立ち上げ、設定されているIPアドレスを記録し、設定を保存します (読み出せない場合は、交換前準備品の④を使用してください)。
- ⑤ CPUモジュール前面のLADDERスイッチをSTOPにし、ユニットの電源をOFFにします。
- ⑥ FL.NETモジュールに接続されているケーブルを外します。
- ⑦ 新しいモジュールと交換し、ロータリスイッチを①で記録した状態に設定します。
- ⑧ ユニットの電源をONにし、Hitachi S10 FL.NETシステムツールから④で記録したIPアドレスを入力し、設定を送信します。
- ⑨ 設定を保存したパラメータと一致しているかを確認してください。
- ⑩ ユニットの電源をOFFにします。
- ⑪ ③で接続したRS-232Cケーブルを外します。

7 保守・点検

- ⑫ ⑥で外したケーブルを元に戻します。
- ⑬ CPUモジュールのスイッチを②で記録した状態に設定します。
- ⑭ ユニットの電源をONにし、正常に動作していることを確認してください。

● 増設手順

- ① CPUモジュール前面のスイッチの設定状態を記録します。
- ② システムの停止を確認後、CPUモジュールのLADDERスイッチをSTOPにし、ユニットの電源をOFFにします。
- ③ 「4. 1 モジュールの実装」を参照のうえ、FL.NETモジュールを実装します。
- ④ メイン側のモジュールと重複しないようにロータリスイッチをサブ側のNo.に設定してください。
- ⑤ パソコンとCPUモジュールをRS-232Cケーブルで接続し、ユニットの電源をONにした後、Hitachi S10 FL.NETシステムツールから増設したFL.NETモジュールにパラメータを設定します。
- ⑥ ユニットの電源をOFFにし、増設したFL.NETモジュールにケーブルを接続します。
- ⑦ CPUモジュール前面のスイッチを①で記録した状態に設定します。
- ⑧ ⑤で接続したRS-232Cケーブルを外します。
- ⑨ ユニットの電源をONにし、正常に動作していることを確認してください。

8 トラブルシューティング

8. 1 故障かなと思ったら

故障かなと思ったら、以下の項目について確認してください。

- ・モジュールは、正しく実装されていますか？
- ・モジュールのスイッチは、正しく設定されていますか？
- ・ネットワークのIPアドレスは、正しく設定されていますか？
- ・コモンメモリ領域は、正しく設定されていますか？
- ・モジュールの接続コネクタなどに緩みがないですか？
- ・通信ケーブルは、正しく接続されていますか？
- ・10BASE-5同軸ケーブルの終端抵抗は、正しく接続されていますか？
- ・10BASE-5同軸ケーブルにアース接地は、接続されていますか？
- ・10BASE-Tケーブルにクロスケーブルを使用していませんか？
- ・10BASE-Tケーブルは、カテゴリ5仕様のケーブルですか？
- ・イーサネットのハブやリピータの電源が入っていますか？

8. 2 ネットワークの不具合とその対策

(1) ネットワークに関する不具合と対策（通信ができない場合）

表 8-1 ネットワークに関する不具合と対策（通信ができない場合）

現象	点検箇所	確認事項	対応方法
通信ができない	電源	機器のメイン電源ランプは、点灯していますか？	電源、電源ケーブルの抜け、電圧を確認してください。
		AUIの電源ユニットの電源ランプは点灯していますか？	電源、電源ケーブルの抜け、電圧を確認してください。
		AUIの電源ユニットの電源出力は規定の電圧（12V）ですか？	電源、電源ケーブルの抜け、電圧を確認してください。
		ハブの電源ランプは点灯していますか？	電源、電源ケーブルの抜け、電圧を確認してください。
		AUI用の電源ケーブルは正しく機器に接続されていますか？	電源、電源ケーブルの抜け、電圧を確認してください。
	通信ケーブルとトランシーバ接続	トランシーバの取り付け部にぐらつきはありませんか？	「9. 6 FL-netのネットワーク施工方法」に従って施工をやり直してください。
		トランシーバの施工状態チェック器で異常はありませんか？	正常になるまで調整してください。連続して異常発生するときは別の箇所に施工してください。
		トランシーバは、正しく絶縁されていますか？	「9. 6 FL-netのネットワーク施工方法」に従って施工をやり直してください。
		トランシーバは、通信ケーブルのマーカ部に正しく取り付けられていますか？	「9. 6 FL-netのネットワーク施工方法」に従って施工位置を見直してください。
	トランシーバケーブルとトランシーバ接続	トランシーバケーブルの取り付け部にぐらつきはありませんか？	「9. 6 FL-netのネットワーク施工方法」に従って施工を見直してください。必要に応じて増し締めしてください。
		トランシーバの施工状態チェック器で異常はありませんか？	チェック器取説に従って施工をチェックしてください。
		トランシーバは、正しくロックされていますか？	「9. 6 FL-netのネットワーク施工方法」に従って正しくロックしてください。
		トランシーバのLEDは正常に点灯していますか？	電源、電源ケーブルの抜け、電圧を確認してください。
	トランシーバケーブルと機器接続	トランシーバケーブルの取り付け部にぐらつきはありませんか？	「9. 6 FL-netのネットワーク施工方法」に従って施工を見直してください。必要に応じて増し締めしてください。
		機器のLEDのTX（送信）、RX（受信）は正常に点灯していますか？	「8 トラブルシューティング」に従って異常内容を確認してください。
メディア切り替えスイッチ（SQEなど）は正しく設定されていますか？		「9. 6 FL-netのネットワーク施工方法」に従って設定を見直してください。	

8 トラブルシューティング

(2) ネットワークに関する不具合と対策（通信が不安定な場合）

表 8-2 ネットワークに関する不具合と対策（通信が不安定な場合）

現象	点検箇所	確認事項	対応方法
通信が全くできないまたは不安定	伝送路の確認	同軸ケーブルの外部導体は一点接地ですか？	「9.6 FL-netのネットワーク施工方法」に従って正しく接地してください。
		AUIケーブルのシールド線は正しくグラウンドに接続されていますか？	メーカーの取扱説明書に従って接地してください。
		Pingコマンドに正しく各局が返答していますか？	返答を返さない局の電源、ケーブルなどをチェックしてください。
		コリジョンランプが頻繁に点灯していませんか？	ケーブル、コネクタの接触を確認してください。アナライザで異常内容を確認してください。
		リピータは、4段以内ですか？	「9.6 FL-netのネットワーク施工方法」に従って構成を見直してください。
		各セグメントは、規定長以内ですか？	「9.6 FL-netのネットワーク施工方法」に従って構成を見直してください。
		終端抵抗は、両端に2個設置されていますか？	「9.6 FL-netのネットワーク施工方法」に従って構成を見直してください。
		各セグメント内の接続機器数は、規定数以内ですか？	「9.6 FL-netのネットワーク施工方法」に従って構成を見直してください。
		機器が接続されているセグメントは、3セグメント以内ですか？	「9.6 FL-netのネットワーク施工方法」に従って構成を見直してください。
		リピータの電源は、入っていますか？	電源、電源ケーブルの抜け、電圧を確認してください。
通信参加局の機器設定確認	通信参加局の機器設定確認	ネットワークのIPアドレスは正しく設定していますか？	設定したIPアドレスをサポートツールとアナライザで再確認してください。
		機器の局番は正しく設定していますか？	設定した局番をサポートツールやアナライザで再確認してください。
		機器のパラメータは正しく設定していますか？	設定した機器のパラメータをサポートツールで再確認してください。
		CD（キャリア検出）ランプは、連続的にまたは断続的に点灯していますか？	通信ケーブル、AUIの電源などを再確認してください。
		TX（送信）ランプは、連続的にまたは断続的に点灯していますか？	機器側の設定を再確認してください。
		LK（リンク）ランプは、連続的に点灯していますか？	機器側のパラメータ設定を再確認してください。

(3) パーソナルコンピュータの“Ping機能”によるIPアドレスの確認方法

FL-netネットワークアナライザなどの専用ツールを使用しなくても汎用のWindows® 95パーソナルコンピュータなどを使用して、対象となるFL-net機器の接続およびIPアドレス設定の確認ができます。以下に、“Ping”機能を使用したIPアドレスの確認方法を示します。

IP接続の場合、“Ping”（ピング）コマンドを使って接続が問題ないか確認してください。

- ① Windows® 95の場合は [スタート] ボタン内 [プログラム] - [MS-DOSプロンプト] を選択し、[MS-DOSプロンプト] を表示します。
- ② “Ping” コマンドを入力し、リンクユニットとパーソナルコンピュータ間の基本となる通信テストを実行します。PingコマンドはPing [IPアドレス] またはPing [ホスト名] と入力します。

<例：IPアドレス> Ping 192.168.250.13

対象のFL-net機器が正しく設定されている場合は、以下のメッセージが表示されます。

```
Pinging 192.168.250.13 with 32 bytes of data
Reply from 192.168.250.13: bytes=32 time=2ms TTL=32
Reply from 192.168.250.13: bytes=32 time=1ms TTL=32
Reply from 192.168.250.13: bytes=32 time=1ms TTL=32
Reply from JEMA 192.168.250.13 : bytes=32 time=1ms TTL=32
C:\WINDOWS>
```

- ③ NG（未接続）の場合、下記のような表示（タイムアウト）になります。

```
Pinging 192.168.250.13 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.
C:\WINDOWS>
```

8 トラブルシューティング

8.3 FL-netを使用するうえでの注意事項

FL-netの伝送路の規格については、前述の節またはIEEE802.3を参照してください。FL-netを使用するうえでの注意事項について以下に示します。

- FL-netの通信ケーブルに他のイーサネットの通信データを流さないでください。
- FL-netをルータに接続しないようにしてください。
- FL-netにスイッチングハブを用いても効果はありません。
- 赤外線や無線などのメディアを使用すると、通信のリアルタイム性が大幅に低下することがあります。
- パーソナルコンピュータを使用した場合、パーソナルコンピュータ本体の能力や使用するOSおよびアプリケーションによって通信のリアルタイム性が大幅に変化することがあります。
- IPアドレスは、決められたアドレスを使用してください。ネットワークアドレスについては揃える必要があります（標準ネットワークアドレスは、192.168.250.です）。また、IPアドレスのノード番号（局番）については入力範囲が推奨されています。ノード番号は、初期設定時には番号の重複チェックはできず、通信して初めてノード番号重複エラーとなりますので、十分注意して設定してください。

ネットワークアドレス	ノード番号
192.168.250.	1～249

- アースは確実に接続してください。また、アース線は十分な太さを確保してください。
- ノイズ源からは十分に隔離してください。また、電源ケーブルとの並設は避けてください。
- サイクリックデータとメッセージデータを同時に通信するときは、データ量などによりリアルタイム性が低下することがあります。
- サイクリックデータ通信の領域（コモンメモリ領域）は連続して確保する必要はありません。
- トランシーバにSQEスイッチが装着されている場合は、取扱説明書に従って正しく設定してください。
- 接続される機器の処理能力によって、システム全体の定時通信性が影響を受けます。最も遅い機器の通信処理能力（最小許容フレーム間隔）とネットワークに接続されるすべての機器が通信処理速度を合わせて通信します。このため、1台の機器接続または追加により、システム全体のリアルタイム性が大幅に低下することがあります。
- メッセージデータ通信のヘッダ部はビッグエンディアンですが、データ部はリトルエンディアンです。ただし、プロファイルリードでのデータ部であるシステムパラメータは、ビッグエンディアンです（ビッグエンディアンとは、MSBを最初に送出する方式を指します）。

8. 4 エラー表示と対策

(1) ツールのエラー表示

ツール上でのエラー表示に関しては、「ソフトウェアマニュアル オプション FL.NET For Windows® (マニュアル番号 SAJ-3-139)」を参照してください。

(2) CPUモジュールのエラー表示

CPUモジュールインディケータ表示は、表8-3に示すようにメイン、サブモジュールで区別します。

表8-3 CPUモジュールインディケータ表示

モジュール 設定	表示内容	内容および説明	対策
メイン	FL-M @. @	FL.NETモジュール (メイン) は正常にネットワークに参加しています。	エラーではありません。
	FLNM□□□□	FL.NETモジュール (メイン) のボードで異常を検出しました。	この項目内のハードウェアエラーまたはFAプロトコルエラーを参照してください。
サブ	FL-S @. @	FL.NETモジュール (サブ) は正常にネットワークに参加しています。	エラーではありません。
	FLNS□□□□	FL.NETモジュール (サブ) のボードで異常を検出しました。	この項目内のハードウェアエラーまたはFAプロトコルエラーを参照してください。

- @. @は、FL.NETモジュールのバージョン、レビジョンを表します。
- □□□□は、ハードウェアエラーまたはFAプロトコルエラーのエラー表示データを表します。

8 トラブルシューティング

[FAプロトコルエラー]

FL.NETモジュールの動作時、FAプロトコルに規定されているエラーを検知した場合、CPUモジュールインディケータに表8-4のメッセージを表示します。

メッセージの内容によっては、FAプロトコルでの規定に従い、FL.NETモジュールは動作を停止します。

表8-4 FAプロトコルエラー時のCPUモジュールインディケータ表示詳細

表示メッセージ	エラー内容	対策	備考
LER	ネットワーク 未参加状態	FL.NETモジュールはネットワークに参加していません。 トラブルシューティングに従ってください。	ネットワークへの参加処理中です。
PER	FL.NETモジュール 設定エラー	FL.NETモジュール内の設定に異常があります。FL.NET設定ツールにてモジュール内の設定を確認し、異常のある場合には設定を修正してください。設定を修正しても異常が発生する場合には、FL.NETモジュールが故障している可能性があります。モジュールを交換してください。	
ADBL	コモンメモリ 設定重複	自ノードのコモンメモリの設定が、他ノードの設定と重複しています。他ノードのコモンメモリ設定と比較して、重複のあるノードの設定を修正してください。	自ノードのコモンメモリの設定がすべて0になっています。FL.NET For Windows®にて、自ノードを再設定してください。
NDBL	ノード番号重複	ネットワーク上に、FL.NETモジュールに設定されたノード番号と同じ番号を使用しているノードが存在します。他ノードのノード番号設定を確認し、ノード番号に重複のないように設定し直してください。	FAリンクプロトコルの規定に従い、ネットワークへの参加を中止しています。 電源再投入、またはFL.NET For Windows®からの参加要求でネットワークへ再加入します。
TABT	トークン保持時間 タイムアウト	設定されているトークン保持時間を連続3回以上オーバーしました。FL.NETモジュールが故障している可能性があります。モジュールを交換してください。	FAリンクプロトコルの規定に従い、ネットワークへの参加を中止しています。

[ハードウェアエラー]

FL.NETモジュールがハードウェアエラーを検出した場合は、CPUモジュールインディケータに表8-5のエラーメッセージを表示します。また、エラーLEDを点灯し、エラーフリーズ情報を収集します。

FL.NETモジュールの動作は停止します。

表8-5 ハードウェアエラー時のCPUモジュールインディケータ表示詳細

表示メッセージ	エラー内容	対策
BUS	バスエラー	FL.NETモジュールが故障している可能性があります。モジュールを交換してください。
ADDR	アドレスエラー	
ILLG	不当命令	
ZERO	0除算	
PRIV	特権違反	
FMAT	フォーマットエラー	
SINT	スプリアス割り込み	
EXCP	未使用例外	
PTY	パリティエラー	
MDSW	モジュールNo.設定スイッチ 設定誤り	モジュールNo.設定スイッチの設定を確認してください。
ROM1	ROM1サムエラー	FL.NETモジュールが故障している可能性があります。モジュールを交換してください。
RAM1	RAM1コンペアエラー	
RAM2	RAM2コンペアエラー	
ROM3	ROM3サムエラー	
IPNG	IPアドレス未登録	IPアドレスを登録してください。
MAC	MACアドレス未登録	FL.NETモジュールが故障している可能性があります。モジュールを交換してください。
PRG	マイクロプログラムエラー	

(注) FAプロトコルエラーとハードウェアエラーが同時に発生した場合には、ハードウェアエラーのみ表示します。

8 トラブルシューティング

FL.NETモジュールがハードウェアエラーを検出した場合は、LERランプを点灯し、エラーフリーズ情報を登録します。FL.NETモジュールの動作は停止します。

表 8-6 エラーフリーズ情報テーブル詳細

メインモジュール	サブモジュール	2 ³¹ ——— 2 ¹⁶ 2 ¹⁵ ——— 2 ⁰	コード	エラー内容
/D40400	/DC0400	エラーコード		
/D40404	/DC0404		0010H	バスエラー
			0011H	アドレスエラー
/D40410	/DC0410	D0レジスタ	0012H	不当命令
/D40414	/DC0414	D1レジスタ	0013H	0除算
/D40418	/DC0418	D2レジスタ	0014H	特権違反
/D4041C	/DC041C	D3レジスタ	0016H	フォーマットエラー
/D40420	/DC0420	D4レジスタ	0017H	スプリアス割り込み
/D40424	/DC0424	D5レジスタ	0018H	未使用例外 (CHK, TRAPV, L1010など)
/D40428	/DC0428	D6レジスタ		
/D4042C	/DC042C	D7レジスタ	0019H	パリティエラー
/D40430	/DC0430	A0レジスタ	001AH	停電予告
/D40434	/DC0434	A1レジスタ	0100H	モジュールNo.設定スイッチの設定誤り
/D40438	/DC0438	A2レジスタ	0102H	ROM1のサムエラー
/D4043C	/DC043C	A3レジスタ	0103H	RAM1のコンペアエラー
/D40440	/DC0440	A4レジスタ	0105H	RAM2のコンペアエラー
/D40444	/DC0444	A5レジスタ	010BH	ROM3のサムエラー
/D40448	/DC0448	A6レジスタ	0113H	IPアドレス未登録
/D4044C	/DC044C	A7レジスタ	0114H	MACアドレスエラー
/D40450	/DC0450	スタックフレーム (4ワード、6ワード、 バスエラー)		
/D404FC	/DC04FC			

(注) スタックフレームについては、次ページに詳細を示します。

エラーフリーズ情報テーブル内スタックフレームの詳細を図8-1に示します。

メインレジスタ	サブレジスタ	フォーマット\$0 (4ビットスタックフレーム)	フォーマット\$2 (6ビットスタックフレーム)	フォーマット\$C (アプリケーションおよびホストの バスエラースタック)	フォーマット\$C (MOVEM オペランドの バスエラースタック)	フォーマット\$C (4ワードおよび6ワード バスエラースタック)
		2 ¹⁵ — 2 ⁰	2 ¹⁵ — 2 ⁰	2 ¹⁵ — 2 ⁰	2 ¹⁵ — 2 ⁰	2 ¹⁵ — 2 ⁰
/D40450	/DC0450	ステータスレジスタ	ステータスレジスタ	ステータスレジスタ	ステータスレジスタ	ステータスレジスタ
/D40452	/DC0452	プログラム カウンタ	次命令プログラム カウンタ	次命令プログラム カウンタ	リターンプログラム カウンタ	リターンプログラム カウンタ
/D40454	/DC0454	0 ベクタオフセット	ベクタオフセット	ベクタオフセット	ベクタオフセット	ベクタオフセット
/D40456	/DC0456	0 ベクタオフセット	ベクタオフセット	ベクタオフセット	ベクタオフセット	ベクタオフセット
/D40458	/DC0458	フォールトを起こした 命令のプログラムカウンタ	フォールトを起こした 命令のプログラムカウンタ	フォールトを起こした 命令のプログラムカウンタ	フォールトを起こした 命令のプログラムカウンタ	フォールトを起こした 命令のプログラムカウンタ
/D4045A	/DC045A				アドレス	アドレス
/D4045C	/DC045C		DBUF	DBUF	DBUF	例外発生前の ステータスレジスタ
/D4045E	/DC045E					ステータスレジスタ
/D40460	/DC0460		現在命令	現在命令	現在命令	現在命令
/D40462	/DC0462		プログラムカウンタ	プログラムカウンタ	プログラムカウンタ	プログラムカウンタ
/D40464	/DC0464		内部転送コントロールスタ	内部転送コントロールスタ	内部転送コントロールスタ	内部転送コントロールスタ
/D40466	/DC0466	0 0	特殊スタワード	特殊スタワード	特殊スタワード	特殊スタワード

図8-1 エラーフリーズ情報テーブル内スタックフレームの詳細

8 トラブルシューティング

[Cモードハンドラ、演算ファンクション検出のエラーコード表]

Cモードハンドラ、演算ファンクションにて、FL.NETモジュールに対して要求をしたときに発生するエラーコードと対策について、表8-7に示します。

表8-7 検出コード一覧

(1/2)

エラーコード	内 容	原 因	対 策
0x0000	メッセージ正常終了	—————	—————
0x0001	メッセージ応答エラー	指定ノード番号から異常応答メッセージを受信しました。	異常応答メッセージの内容は、エラーメッセージ格納テーブルに格納されています。指定ノードのマニュアルを参照して、指定ノードの状態を確認してください。
0x0002	メッセージ未対応	指定されたノードでは、ユーザの要求したメッセージ機能に対応していません。	そのノードに対して、未対応となったメッセージは発行しないでください。
0xFE00	パラメータエラー	ユーザより指定されたパラメータに異常があります。透過型メッセージ受信を発行した場合、透過型メッセージを受信していません。	要求発行時のパラメータをチェックしてください。また、透過型メッセージ受信は、メッセージの受信があることを確認してから行ってください。
0xFE01	自ノード未接続	FL.NETモジュールは、ネットワークに参加していません。	FL.NETモジュールがネットワークに参加してから要求を行ってください。
0xFE02	指定ノード未接続	ユーザの指定したノード番号は、ネットワークに参加していません。	ネットワークに参加しているノード番号を指定してください。
0xFE03	他メッセージ処理中	前回指定された要求を処理中のため、今回の要求を受け付けられません。	前回指定した要求が終了してから、再度要求を発行し直してください。
0xFE04	メッセージACK応答なし	指定ノード番号からのACK応答を受信しませんでした。	モジュールの故障が考えられます。モジュールを交換してください。
0xFE06	データ受信なし	指定ノード番号へのメッセージ要求発行後、30秒経過してもメッセージ要求に対する応答を受信しませんでした。	モジュールの故障が考えられます。モジュールを交換してください。

(2/2)

エラーコード	内 容	原 因	対 策
0xFE08	ACK受信通番エラー	指定ノード番号からのACK応答で通番エラーを受信しました。	モジュールの故障が考えられます。モジュールを交換してください。
0xFE09	ACK受信通番バージョンエラー	指定ノード番号からのACK応答で通番バージョンエラーを受信しました。	
0xFE12	メッセージキュー満杯	指定ノード番号のメッセージキューが満杯になっています。指定ノード番号は要求を受信できません。	しばらく間をおいてから、要求を再発行するか、指定ノード番号への要求を減らしてください。
0xFE13	初期化エラー	指定ノード番号はメッセージ処理初期化が終了していません。	しばらく間をおいてから、要求を再発行してください。
0xFE16	メッセージサイズエラー	指定ノード番号から、こちらの要求したメッセージのサイズが異常と報告されました。	モジュールの故障が考えられます。モジュールを交換してください。
0xF0XX または 0xFFXX	ドライバ異常	ユーザ要求のメッセージ送信時、ドライバにて異常を検知しました。	

8 トラブルシューティング

[エラーメッセージデータテーブル]

自ノードからのメッセージ要求に対する応答メッセージが異常応答メッセージであった場合、そのメッセージデータは、FL.NETモジュール内のエラーメッセージデータテーブルに格納されます。

以下、エラーメッセージテーブルの詳細仕様を示します。

メインモジュール	サブモジュール	2 ¹⁵	2 ⁰
0xD41380	0xDC1380	エラーメッセージ総数	
0xD41382	0xDC1382	送信元ノード番号	
0xD41384	0xDC1384	受信TCD	
0xD41386	0xDC1386	エラーメッセージ語数 (バイト単位)	
0xD41388	0xDC1388	エラーコードデータ部	
0xD41788	0xDC1788		

項 目	詳 細
エラーメッセージ総数	電源投入後から受信した異常応答メッセージ総数
送信元ノード番号	異常応答メッセージの送信元ノード番号
受信TCD	異常応答メッセージのTCD番号
エラーメッセージ語数	異常応答メッセージのデータ部のサイズ (エラーコードサイズ) です。バイト単位で表示されます。
エラーコードデータ部	異常応答メッセージのデータ部 (エラーコード) を格納するエリアです。最大1024バイトのデータを格納できます。

(注) エラーメッセージテーブルに異常応答メッセージが格納されているときに新たな異常応答メッセージを受信した場合、エラーメッセージ総数を更新 (+1) し、エラーメッセージの内容を上書きします。

(3) モジュールのエラー表示

FL.NETモジュールに異常が発生した場合、モジュール上のLER LEDが点灯します。

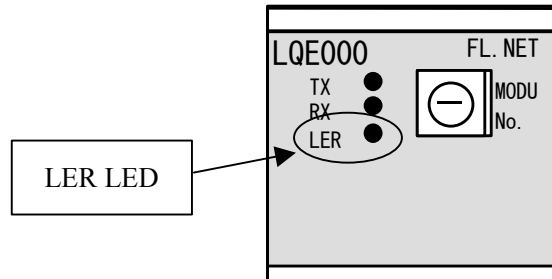


図8-2 モジュールのエラーランプ

ただし、LER LEDは、モジュールがネットワークに参加していない場合でも点灯しますので、LER LEDだけでは本当に異常であるかを判断できません。

LER LEDが点灯した場合、S10miniではCPUモジュールインディケータに詳細内容が表示されますので、内容を確認してください。

インディケータ表示内容の詳細については、「(2) CPUモジュールのエラー表示」を参照してください。

(4) 通信ドライバのエラーアラーム

FL.NETモジュールは、ドライバからのエラーアラームを表示しません。

(5) 通信ログデータの見方

FL.NETモジュール内の通信ログは、設定ツール [FL.NET For Windows®] にて参照できます。詳しい操作手順に関しては、「ソフトウェアマニュアル オプション FL.NET For Windows® (マニュアル番号 SAJ-3-139)」を参照してください。

<このページは余白です>

9 付 録

9. 1 システム構築ガイド

9. 1. 1 イーサネットの概要

イーサネットは、パーソナルコンピュータやプリンタなどの間で通信するためのLAN（Local Area Network）の規格で、通信データフォーマットやケーブル、コネクタなどを規定しています。イーサネットの規格は、IEEEのイーサネットワーキンググループ（IEEE802.3）で制定されていて、現在までに10BASE-5, 10BASE-2, 10BASE-Tなどの方式の規格が制定され、現在も1000BASE-Tなどの新しい方式の規格を検討しています。

図9-1にIEEE802.3の標準化動向を示します。

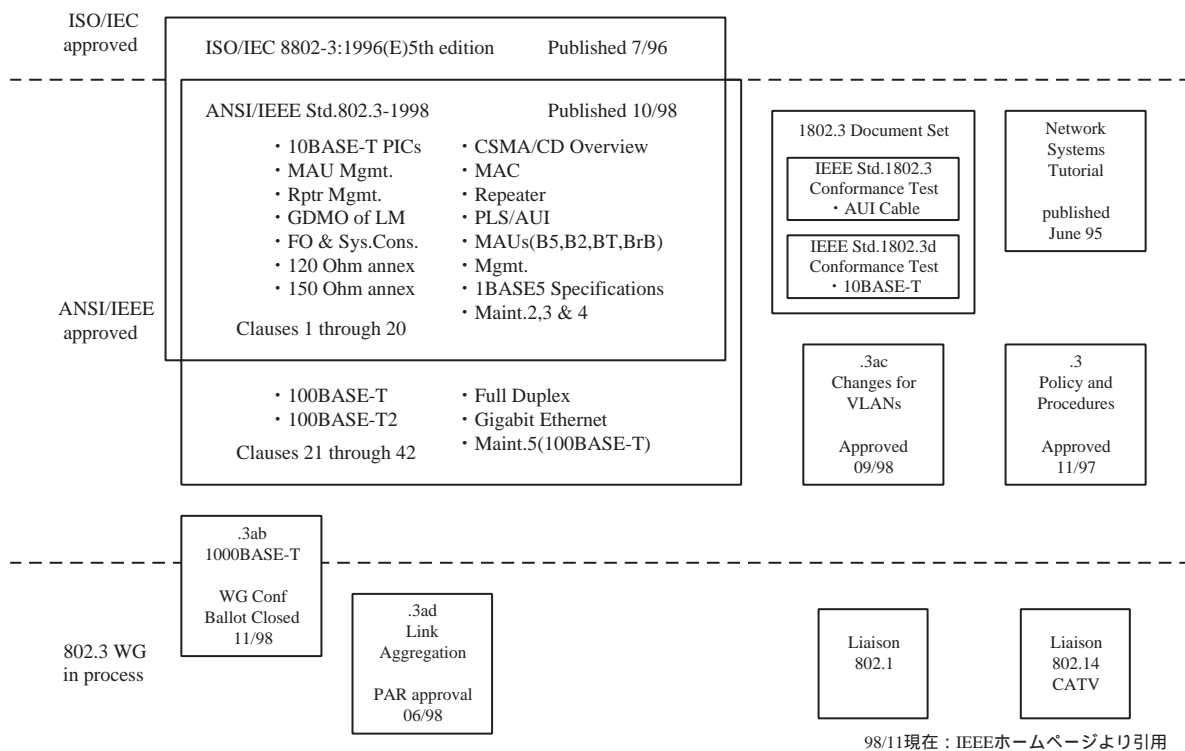


図9-1 IEEE802.3標準化動向

9. 1. 2 10BASE-5の仕様

10BASE-5は、太さ約10mmの同軸ケーブル（Thickケーブル、イエローケーブルともいう）を用いたイーサネットの接続方式のことです。10BASE-5の「10」はイーサネットの伝送速度が10Mbpsであることを、「BASE」は伝送方式がベースバンド方式であることを、「5」は幹線の伝送距離が500mであることを表しています。パーソナルコンピュータなどの機器に接続するためには、同軸ケーブルにトランシーバを取り付け、そこからトランシーバケーブル（AUIケーブルともいう）を介して接続します。

10BASE-5は、ケーブルが太くネットワークの敷設が容易ではないため、オフィスのネットワークではあまり利用されていませんが、伝送距離が長いので、幹線のネットワークでよく利用されています。

図9-2に10BASE-5イーサネットの構成例を示します。

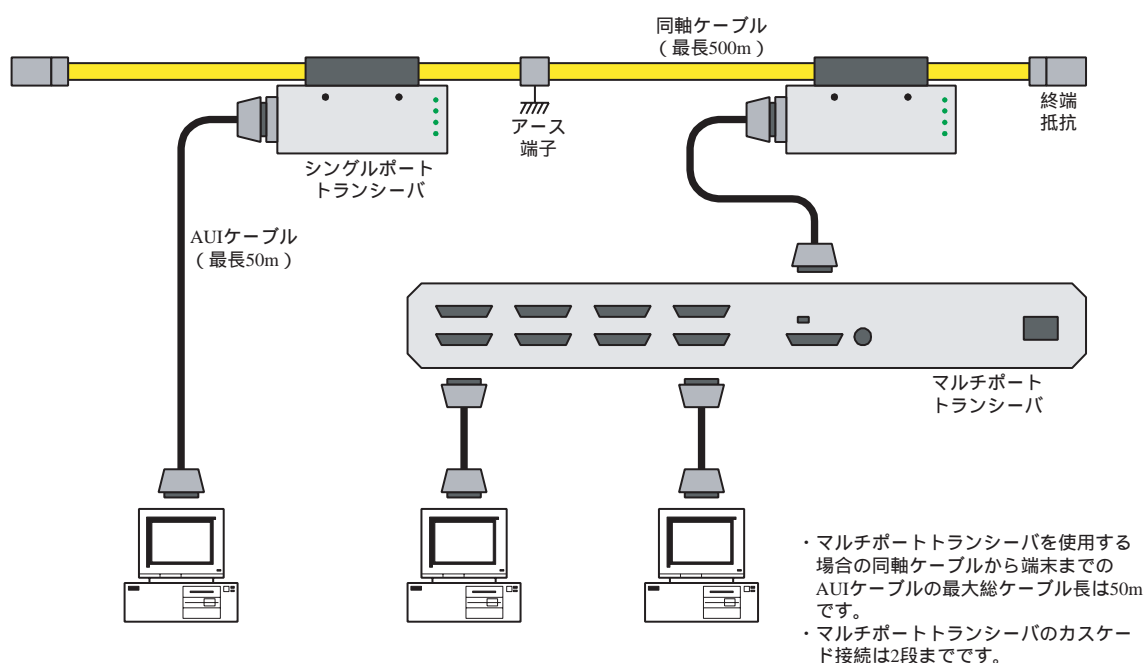


図9-2 10BASE-5イーサネット構成例

9. 1. 3 10BASE-Tの仕様

10BASE-Tは、ツイストペアケーブルを用いたイーサネットの接続方式のことです。10BASE-Tの「10」はイーサネットの伝送速度が10Mbpsであることを、「BASE」は伝送方式がベースバンド方式であることを、「T」は伝送媒体がツイストペアケーブルであることを表しています。10BASE-Tのネットワークでは、パーソナルコンピュータなどの機器は、ハブによってスター状に接続する必要があり、機器同士を直接接続できません（クロスケーブルと呼ばれる特殊なケーブルを用いれば、1対1での直接接続ができますが、一般的ではありません）。ハブから各機器までのケーブルの最長は100mです。

10BASE-Tは、ケーブルが細く引き回しが容易であり、また、各機器を個別にネットワークに接続したり、切り離したりすることができるため、オフィスのネットワークでよく利用されています。

図9-3に10BASE-Tイーサネットの構成例を示します。

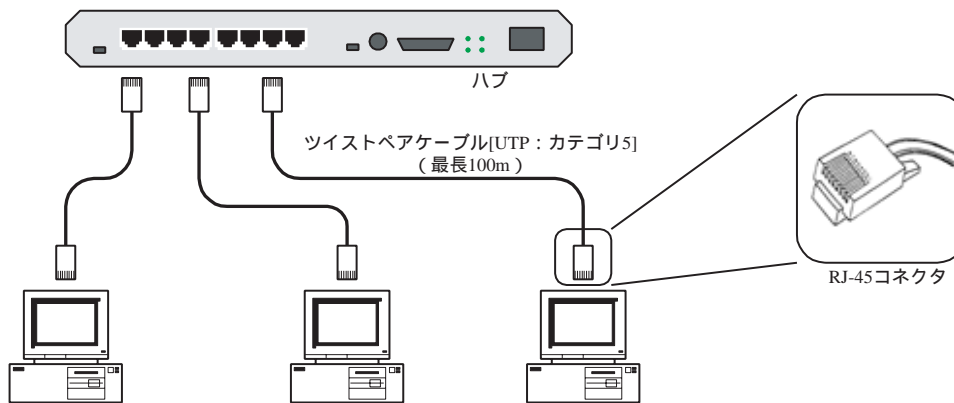


図9-3 10BASE-Tイーサネット構成例

9. 1. 4 その他イーサネットの仕様

(1) 10BASE-2

10BASE-2は、太さ約5mmの同軸ケーブル（Thinケーブルともいう）を用いたイーサネットの接続方式のことです。10BASE-2の「10」はイーサネットの伝送速度が10Mbpsであることを、「BASE」は伝送方式がベースバンド方式であることを、「2」は幹線の伝送距離が185m（≒200m）であることを表しています。パーソナルコンピュータなどの機器に接続するためには、各機器のBNCコネクタにT字型の分岐コネクタを取り付け、その両端に同軸ケーブルを接続します。図9-4に10BASE-2イーサネットの構成例を示します。

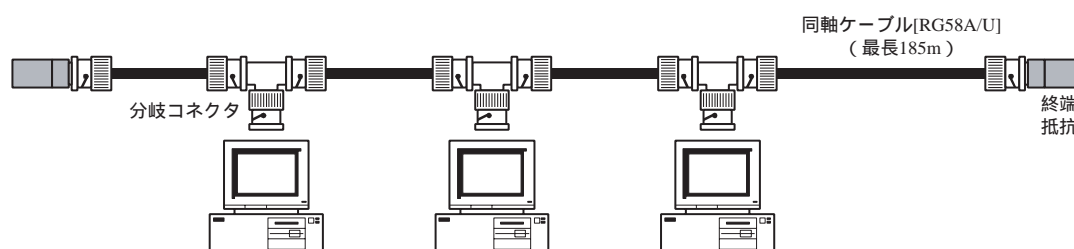


図9-4 10BASE-2イーサネット構成例

(2) 光イーサネット

光イーサネットは、伝送媒体に光ファイバを利用するイーサネットであり、500m以上の長距離伝送や耐ノイズ性が要求されるシステムなどで用いられています。IEEE802.3で規格化された光イーサネットの接続方式には、10BASE-FP, 10BASE-FB, 10BASE-FL, 100BASE-FX, 1000BASE-LX, 1000BASE-SXがあります。図9-5に光イーサネットの構成例を示します。

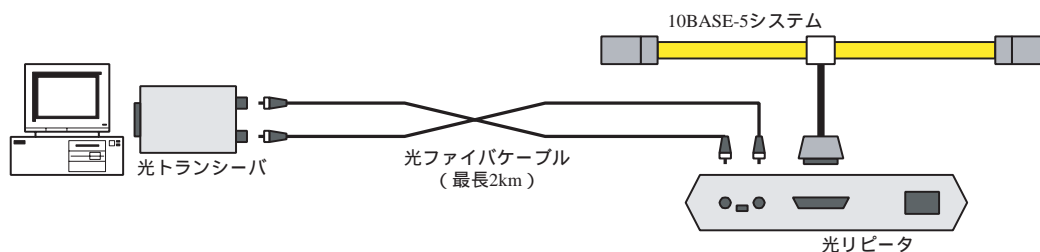


図9-5 光イーサネット構成例

(3) 無線イーサネット

無線LANは、伝送媒体に電波や赤外線を利用するLANであり、携帯機器をLANに接続する場合などに用いられています。無線LANは、IEEEの無線LANワーキンググループ（IEEE802.11）で規格化作業が進められています。無線LANとイーサネットでは、MAC層のプロトコルが異なるため、相互接続するためにはブリッジが必要です。

図9-6に無線イーサネットの構成例を示します。

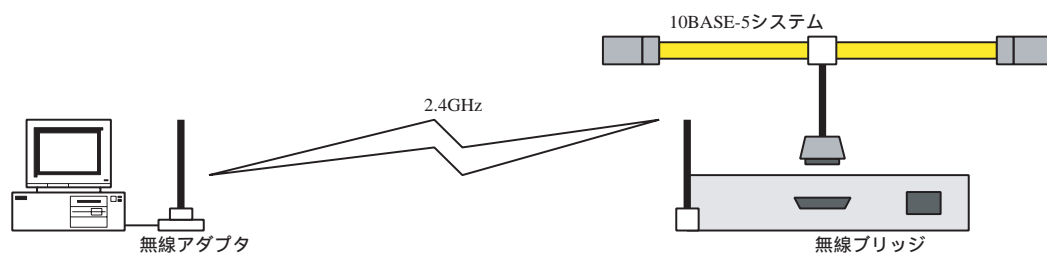
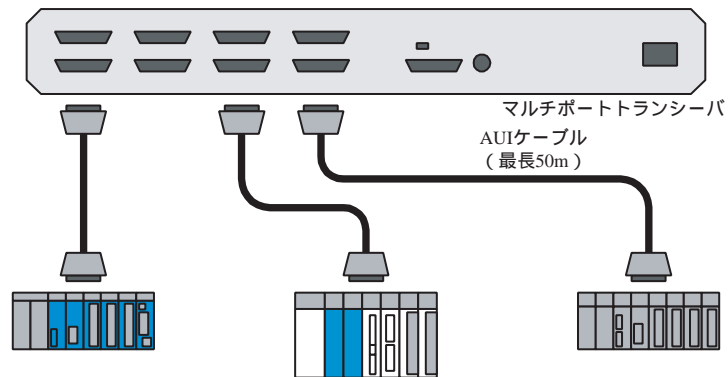


図9-6 無線イーサネット構成例

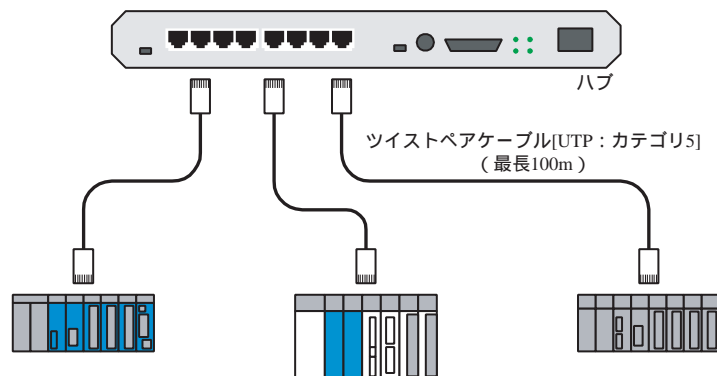
9.2 システム構成例

9.2.1 小規模構成

1台のマルチポートトランシーバまたはハブを用いて、数台の機器によるネットワークシステムを構築できます。



(a) マルチポートトランシーバを使う場合

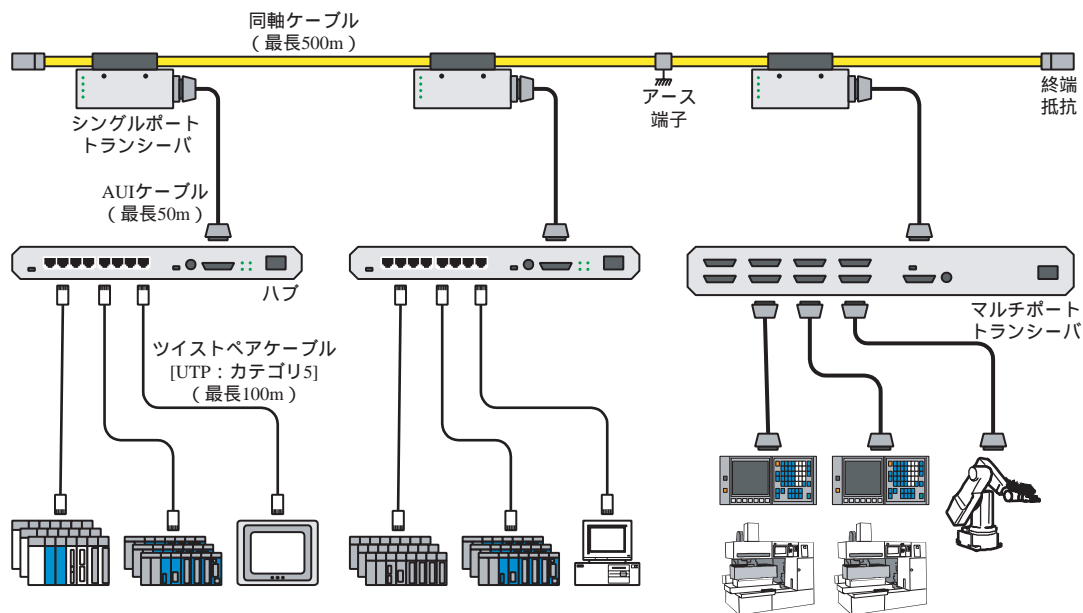


(b) ハブを使う場合

図9-7 小規模構成例

9. 2. 2 基本構成

1本の同軸ケーブルに、数台のマルチポートトランシーバやハブを接続し、数十台の機器によるネットワークシステムを構築できます。



・任意の2端末間に設置できるリピータおよびハブは合計4台までです。

・マルチポートトランシーバを使用する場合の同軸ケーブルから端末までのAUIケーブルは最長50mです。
 ・マルチポートトランシーバのカスケード接続は2段までです。

図 9 - 8 基本構成例

9.2.3 大規模構成

複数の10BASE-5のネットワークセグメントをリピータで接続し、数百台の機器によるネットワークシステムを構築できます。

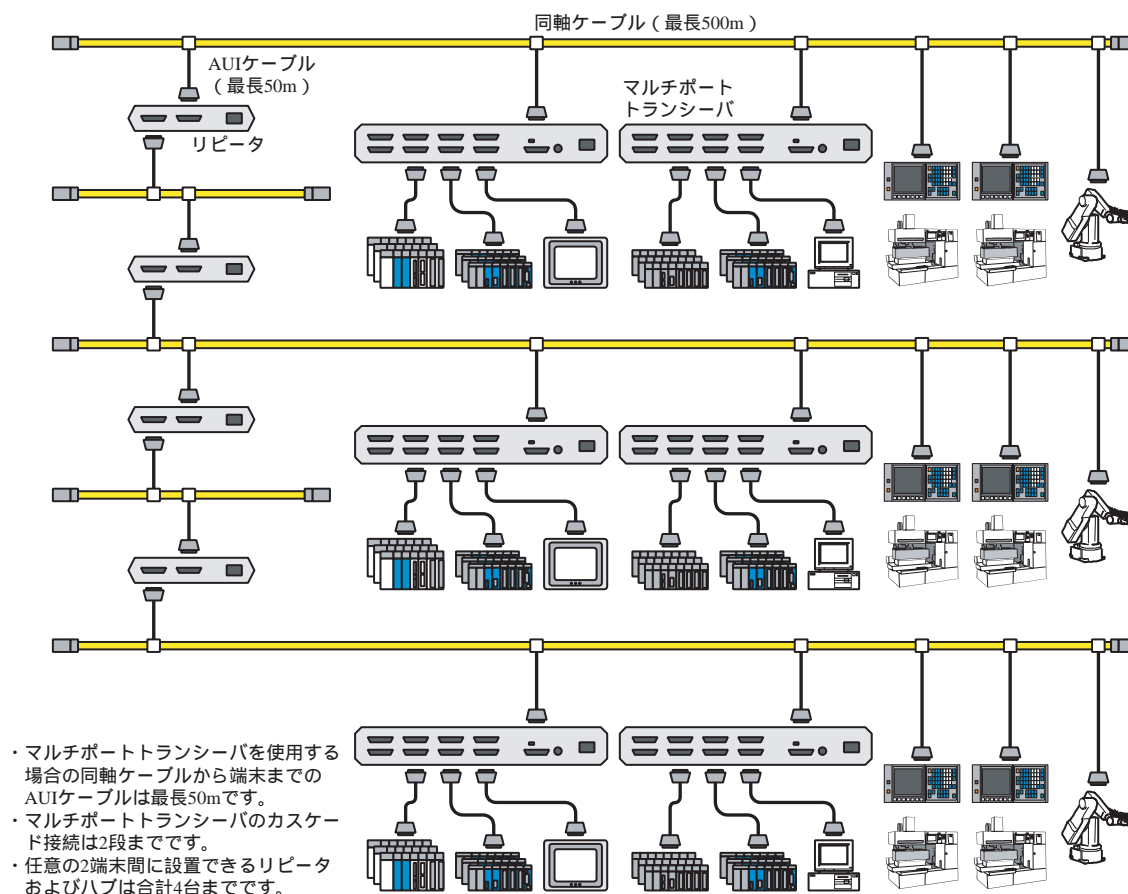


図9-9 大規模構成例

9. 2. 4 長距離分散構成

大規模構成のネットワークシステムにおいて、各ネットワークセグメント間の距離が10BASE-5の伝送距離の制限（500m）を超える場合は、各ネットワークセグメントを光リピータで接続することで、リピータ間が2kmのネットワークシステムを構築できます。

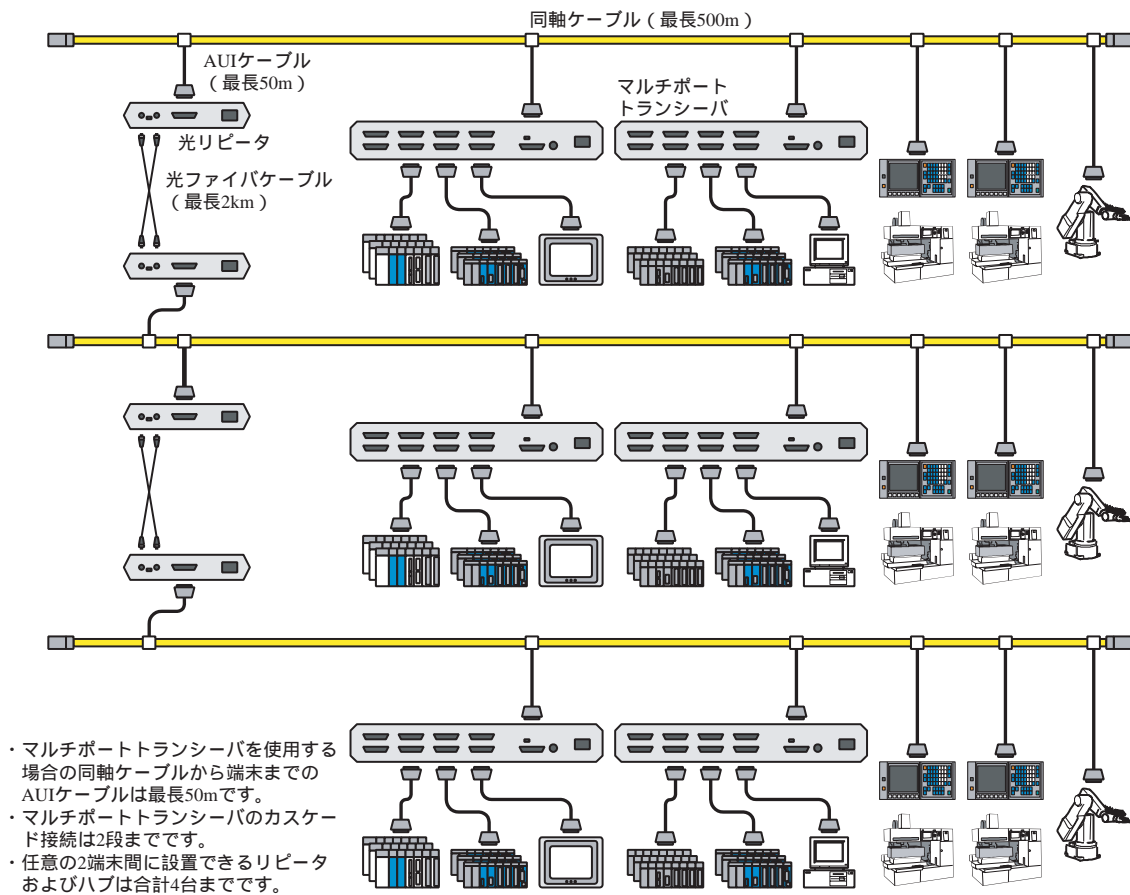


図9-10 長距離分散構成例

9. 2. 5 局所集中構成

数十台の機器が局所的に集中している場合は、スタックابلハブを用いたネットワークシステムを構築できます。

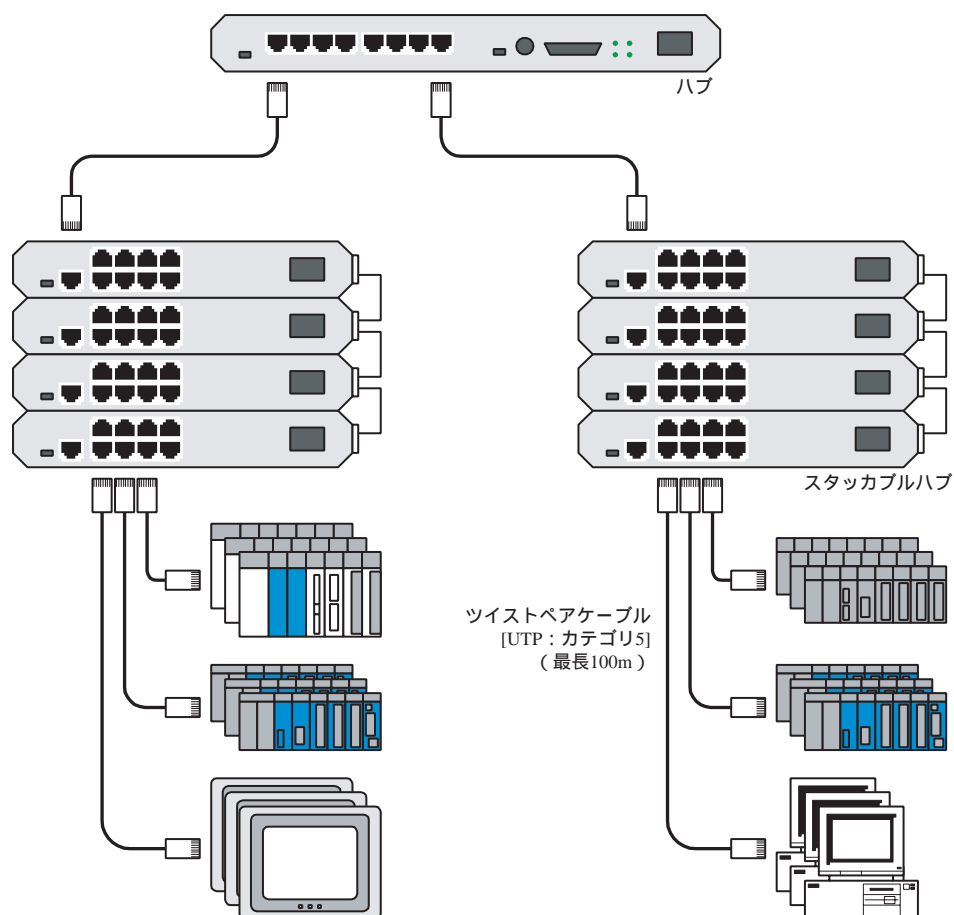


図 9-11 局所集中構成例

9. 2. 6 局所長距離分散構成

基本構成のネットワークシステムにおいて、特定のコントローラが遠距離にある場合や、ネットワークの近辺に高圧電源やノイズ源がある場合は、ネットワークを2つのセグメントに分割し、各セグメント間を光リピータで接続することで、長距離でかつ耐ノイズ性に優れたネットワークシステムを構築できます。

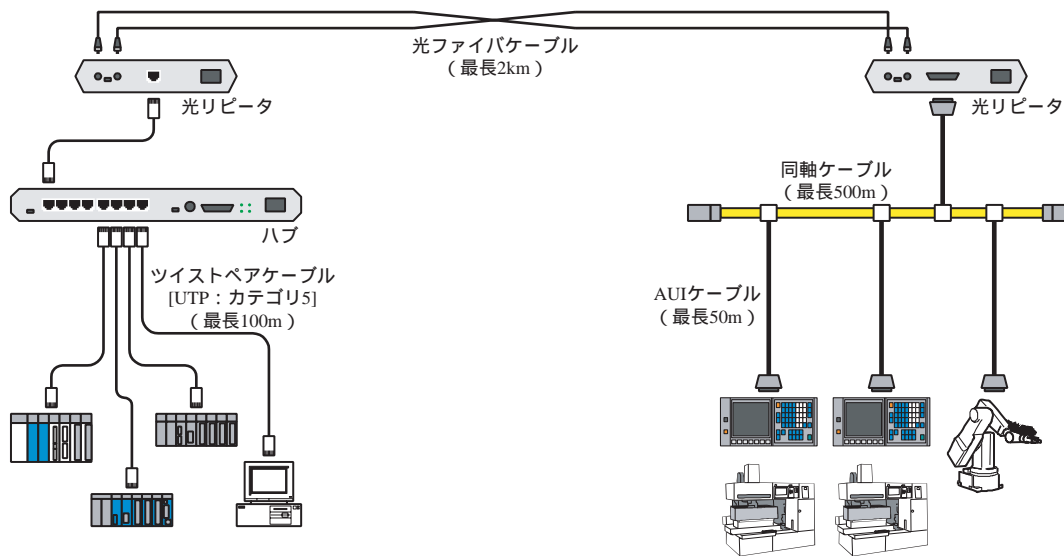


図 9 - 12 局所長距離分散構成例

9. 2. 7 FL-netのシステムの考え方

FL-netは、生産システムにおけるプログラマブルコントローラ、ロボットコントローラ、数値制御装置などの、コントローラ間のリアルタイム通信を目的としています。

FL-netはイーサネットのUDP/IPプロトコル上に、一斉同報を用いたトークンパッシング機構を構築し、その上で、サイクリック通信とメッセージ通信を実現しています。

9. 2. 8 汎用のイーサネットとFL-netの相違点

- FL-netはFA分野用のネットワークであるため、汎用のイーサネット機器がすべて利用できるわけではありません。耐ノイズ性や耐環境性で使用に適さない機器があります。
- FL-netは制御用途のリアルタイム通信として応答性能が要求されているため、FL-net対応のコントローラや制御機器のみを接続できます。
- FL-netは10BASE-5/-TベースのUDP/IP通信の一斉同報機能を使用したサイクリック通信方式のため、現在の規約では、下記の制限事項があります。
 - ・現在の対応機器は、10MbpsのイーサネットLANのみです。
 - ・他の汎用イーサネットとの接続ができません。
 - ・TCP/IP通信機能をサポートしていません。
 - ・スイッチングハブを使用しても効果がありません。
 - ・ルータなどを使用した場合には、機能できない場合があります。

9.3 ネットワークシステムの定義

9.3.1 通信プロトコルの規格

通信プロトコルとは、あるシステムが別のシステムと通信回線などを介して情報をやり取りするためのルール（通信規約）のことを指します。FL-netで使用している通信プロトコルは次のような規格に準拠しています。

表9-1 FL-netの通信プロトコル

FL-netの通信プロトコル	準拠仕様
FL-net	FAリンクプロトコル仕様書 (MSTC FAオープン推進協議会 コントロールネットワーク専門委員会発行)
UDP	RFC768
IP, ICMPなど	RFC791,792,919,922,950
ARPなど	RFC826,894
—	IEEE802.3

9.3.2 通信プロトコルの階層構造

通信プロトコルは階層構造でモデル化され、通信処理をいくつかのレベルに分割・整理して表現、規格化します。FL-netは、以下のように6つのプロトコル層から構成されています。

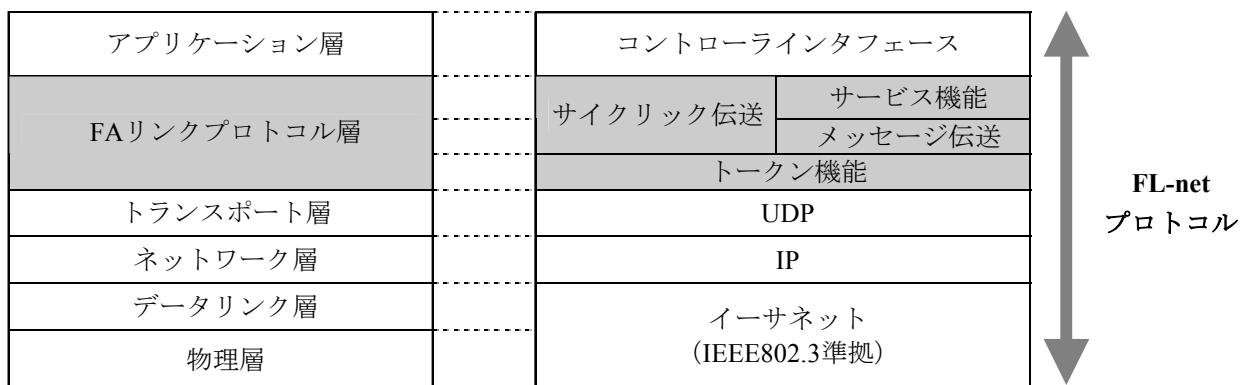


図9-13 FAリンクプロトコルの位置付け

9. 3. 3 FL-netの物理層について

伝送速度が10Mbpsの場合、イーサネットの物理層には5種類の伝送方式があります。

10BASE-5, 10BASE-2, 10BASE-T, 10BASE-F, 10BROAD36（ただし、ほとんど普及していません。）また、これ以外に100Mbpsイーサネットが存在します。

FL-netは、これらの中から10BASE-5（推奨）、10BASE-2および10BASE-Tを採用しています。

9. 3. 4 IPアドレス

イーサネットにて接続された数多くの通信機器の中から指定された通信機器を識別するために、IPアドレス（INETアドレス）と呼ばれるアドレスを使用しています。そのためイーサネットに接続された各通信機器は、それぞれ唯一固有のIPアドレスを設定してください。

IPアドレスは、その通信機器が接続されているネットワークアドレスを表す部分と、その通信機器のホストアドレス部分で構成されていて、ネットワークの大きさによって、クラスA, B, Cの3種類のネットワーククラスに分類できます（このほかに特殊な目的のためにクラスD, Eがあります）。

表 9-2 IPアドレスのクラス

	先頭の1オクテット値	ネットワークアドレス部	ホストアドレス部
クラスA	0~127	xxx.xxx.xxx.xxx	xxx.xxx.xxx.xxx
クラスB	128~191	xxx.xxx.xxx.xxx	xxx.xxx.xxx.xxx
クラスC	192~223	xxx.xxx.xxx.xxx	xxx.xxx.xxx.xxx

xxx.網かけで示された箇所がそれぞれのアドレス部に対応する部分

1つのネットワークの中で、そのネットワークに接続されている通信機器のIPアドレスは、すべて同じネットワークアドレス部となり、ホストアドレス部は重複しない唯一固有の値となります。

FL-netのIPアドレスは、クラスCを使用しネットワークアドレスに「192.168.250.N」（Nはノード番号：1~254）を使用することを推奨しています。

また、下位のホストアドレスとFL-netプロトコルのノード番号を一致させることも推奨します。

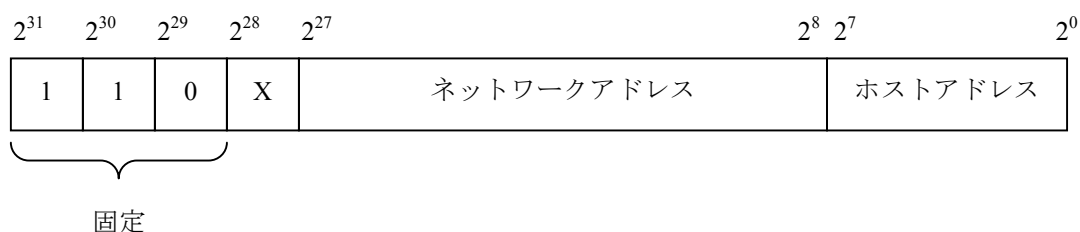


図 9-14 FL-netのIPアドレス

9. 3. 5 サブネットマスク

FL-netプロトコル規約では、サブネットマスクを255.255.255.0固定としています。FL-netのユーザは、サブネットマスクを必ず255.255.255.0に設定してください。

この値はクラスCの本来のネットワークアドレス部とホストアドレス部の区分と同じになります。

9. 3. 6 TCP/IP, UDP/IP通信プロトコル

TCP, UDP, IPはいずれも、いわゆるイーサネットで使用される主要なプロトコルです。

IPは通信プロトコルのネットワーク層に位置して、通信データの流れを制御しています。

TCPとUDPはトランスポート層に位置して、いずれもIPをネットワーク層として利用していますが、サービス内容に大きな違いがあります。

TCPは、上位層に対してデータの区切りを意識させない信頼性のあるサービスを提供します。一方、UDPはIPからのデータのかたまり（データダイアグラム）をそのまま上位層へ伝送するために機能し、データが送信先に到達したかどうか保証しません。データの受信確認・再送などはさらに上位の層に任せています。

UDP自体はTCPに比べて信頼性がないかわりに、オーバーヘッドの小さい通信サービスを提供できます。

FL-netでは、UDPを使用しています。これはTCPの凝ったデータ確認再送の手続きがFL-netに対して冗長であることによります。この手続きを省き、代わりに上位のFL-netプロトコル層で、トークンによる送信権の管理、複数フレームの分割・合成などを行うことで、高速なデータ交換を提供します。

9. 3. 7 ポート番号

FL-netには、トランスポート層の上位に位置するFL-netプロトコル層でサービスを実現するために、次のポート番号があらかじめ定められています。ただし、FL-netのユーザは、パラメータなどにこれらのポート番号を設定する必要はありません。

表 9-3 FL-netのポート番号

名 称	ポート番号
サイクリック伝送用ポート番号	55000 (固定)
メッセージ通信用ポート番号	55001 (固定)
参加要求フレーム用ポート番号	55002 (固定)
送信用ポート番号	55003 (固定)

9.3.8 FL-netのデータフォーマット

(1) FL-netのデータフォーマット概要

FL-netで送受信されるデータは、通信プロトコルの各層で以下のようにカプセル化されています。

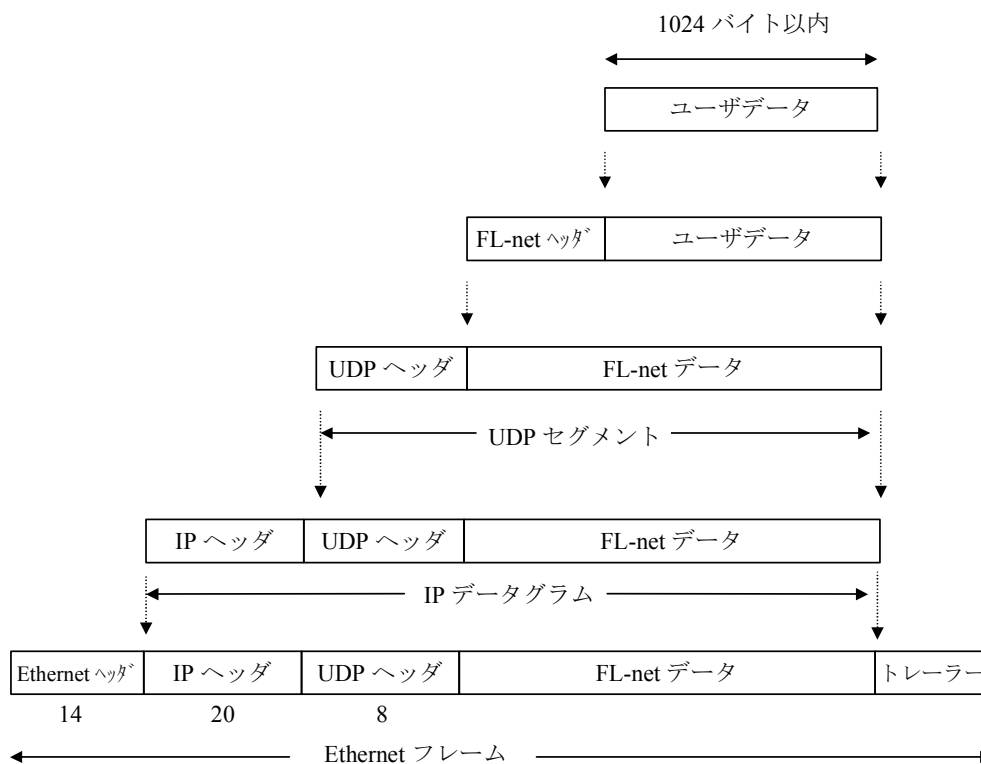


図 9-15 FL-netのデータフォーマット概要

以下に通信回線上で観測できるFL-netデータ（1フレーム分）を示します。例では、128バイトのサイクリックデータが転送されています。

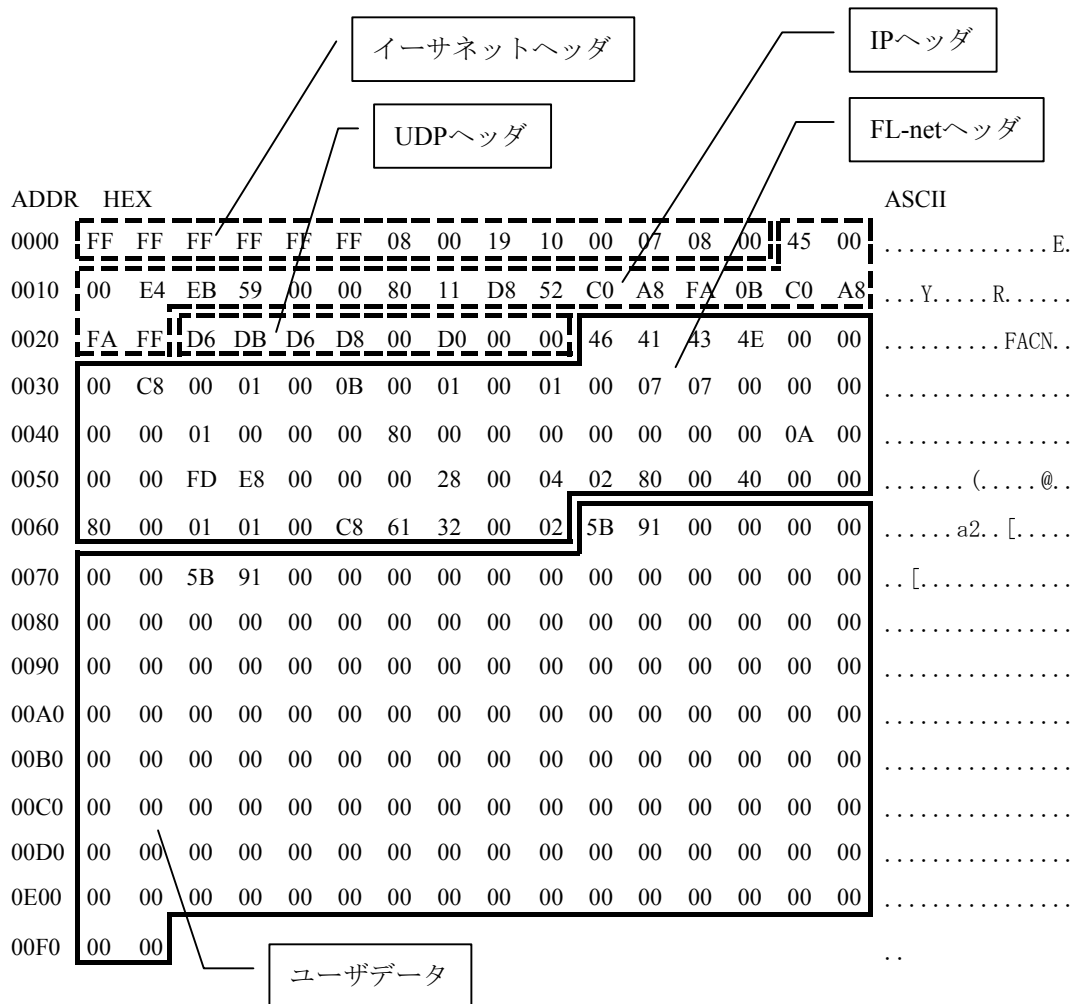


図9-16 FL-netのデータ（1フレーム）例

(2) FL-netのヘッダフォーマット

FL-netヘッダは、64から96バイトの大きさを持っています。

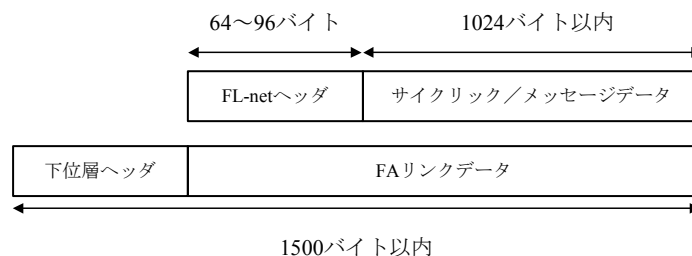


図9-17 FL-netヘッダ

FL-netヘッダは、FL-netプロトコルにおけるすべてのフレームの先頭に付けられます。

9.3.9 FL-netのトランザクションコード

FL-netではメッセージ伝送で次のサービスを実現しています。

- ・ワードブロックデータの読み出し
- ・ワードブロックデータの書き込み
- ・ネットワークパラメータの読み出し
- ・ネットワークパラメータの書き込み
- ・停止指令 (要求のみサポート)
- ・運転指令 (要求のみサポート)
- ・プロファイルの読み出し (応答のみサポート)
- ・ログデータの読み出し
- ・ログデータのクリア
- ・メッセージ折り返し
- ・透過型

それぞれのメッセージには、そのヘッダに要求用のトランザクションコードまたは応答用のトランザクションコードがあり、メッセージフレームを識別します。

表9-4 トランザクションコード一覧

トランザクションコード	適 用
0～59999	透過型のメッセージ
60000～64999	リザーブ
65000	サイクリックヘッダ（トークン付き）
65001	サイクリックヘッダ（トークンなし）
65002	参加要求フレームヘッダ
65003	バイトブロックデータの読み出し（要求）
65004	バイトブロックデータの書き込み（要求）
65005	ワードブロックデータの読み出し（要求）
65006	ワードブロックデータの書き込み（要求）
65007	ネットワークパラメータの読み出し（要求）
65008	ネットワークパラメータの書き込み（要求）
65009	停止指令（要求）
65010	運転指令（要求）
65011	プロファイルの読み出し（要求）
65012	トリガ・ヘッダ
65013	ログの読み出し（要求）
65014	ログのクリア（要求）
65015	メッセージ折り返し試験用（要求）
65016～65202	リザーブ（将来拡張用）
65203	バイトブロックデータの読み出し（応答）
65204	バイトブロックデータの書き込み（応答）
65205	ワードブロックデータの読み出し（応答）
65206	ワードブロッ・データの書き込み（応答）
65207	ネットワークパラメータの読み出し（応答）
65208	ネットワークパラメータの書き込み（応答）
65209	停止指令（応答）
65210	運転指令（応答）
65211	プロファイルの読み出し（応答）
65212	リザーブ
65213	ログデータの読み出し（応答）
65214	ログデータのクリア（応答）
65215	メッセージ折り返し試験用（応答）
65216～65399	リザーブ（将来拡張用）
65400～65535	リザーブ

9. 4 FL-netのネットワーク管理

9. 4. 1 FL-netのトークン管理

(1) トークン

ノードが送信できるのは、基本として、そのノードがトークンを保持しているときです。

トークンを保持していないときに送信できるのは、トークン監視時間のアップによるトークン再発行とネットワーク未加入時の参加要求フレームの2つのみです。

- FAネットは、1つのトークンをノード間で回します。
- 各ノードは、このトークンを受け取ってから、次のノードにトークンを引き渡すまで、ネットワークに対する送信権を保持します。
- トークンは、FL-netに参加するすべてのノードを巡回します。
- トークンは、サイクリックデータを伴って送信することができます。
- トークンは、データを付けずにトークンのみを回すこともできます。
- トークンは、タイマによって監視され一定時間ネットワークに流れないと自動的に再発行されます。
- トークンがネットワーク上に2つ以上あるとき、1つに統一する機能を持ちます。

(2) トークンの流れ

トークンは、基本的にネットワークに1つだけが存在します。

ネットワークに2つ以上のトークンが存在した場合、ノードは宛先ノード番号が小さい方を優先し、他方を破棄します。

トークンを含むフレーム（トークンフレーム）には、トークンの宛先ノード番号とトークン送出ノード番号を持ちます。

各ノードは、受信したトークンフレームのトークンの宛先ノード番号と一致した場合にトークン保持ノードとなります。

トークンのローテーションの順番は、ノード番号によって決定されます。

各ノードは参加ノード管理テーブルに登録されているノードの中の昇順でトークンのローテーションを行います。

最大ノード番号のノードは、最小ノード番号のノードにトークンを渡します。

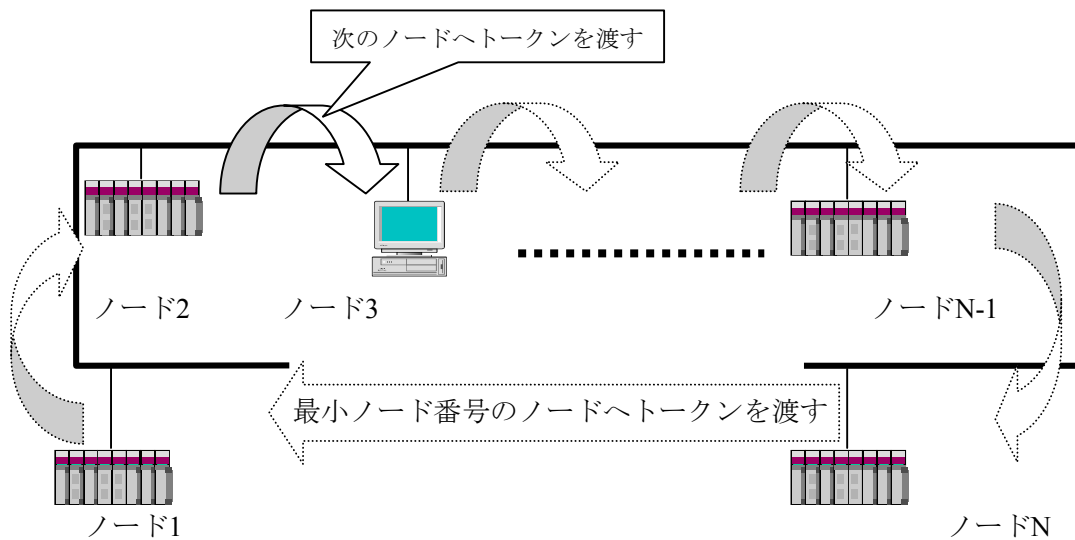
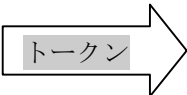
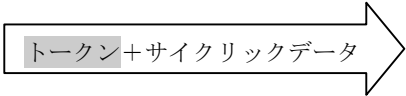
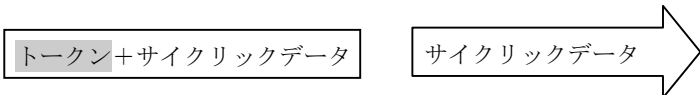

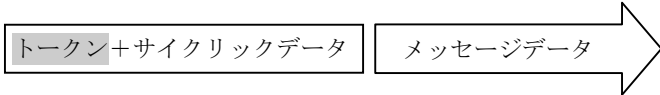
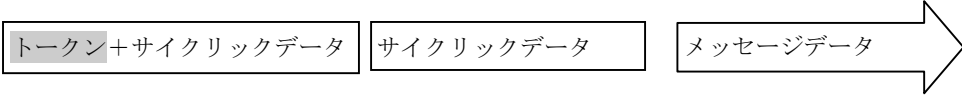


図9-18 トークンの流れ

(3) トークンとデータ

トークンを送信するときに伴うデータのパターンは、次の6種類です。

表 9-5 トークンとデータ

項 目	内 容
伴うデータがないとき	トークンのみを送信します。
	
サイクリックデータだけのとき	サイクリックデータにトークンを付けて送信します。
	
サイクリックデータのみで、サイクリックデータを分割して送るとき	サイクリックデータのみを送信し、最後のフレームにトークンを付けて送信します。
	
メッセージデータだけのとき	メッセージデータを送信後、トークンを送信します。
	
サイクリックデータとメッセージデータのとき	メッセージデータを送信後、サイクリックデータにトークンを付けて送信します。
	
サイクリックデータとメッセージデータで、サイクリックデータを分割して送るとき	メッセージデータを送信後、サイクリックデータのみを送信し、最後のフレームにトークンを付けて送信します。
	

(4) フレームの間隔（最小許容フレーム間隔）

他ノードからトークンを受けて自ノードがフレームを出すまでの時間をフレーム間隔と呼びます。各ノードが最低限フレームを出すまで待たなければならない時間を最小許容フレーム間隔と呼びます。

FL-netでは、この最小許容フレーム間隔をネットワークで共有します。

各ノードは、ネットワークに参加しているノードが設定している最小許容フレーム間隔の最大値をノードの加入／離脱があるたびに計算され、更新されます。

9. 4. 2 FL-netへの加入／FL-netからの離脱

(1) FL-netへの加入

各ノードは立ち上がり時、それぞれ加入トークン検出時間がアップするまで回線を監視します。このとき、トークンを受信しなかった場合はネットワーク立ち上がり時と判定し、ネットワークへ新規参加します。また、トークンを受信した場合は途中参加状態と判定し、ネットワークへ途中参加します。

(a) 新規参加

加入トークン検出時間を経過してもトークンを受信しない場合は、トリガの送信準備を行い、ノード番号/8の余り×4ms後に実際に送信します。トリガの送信前にトリガを受信した場合はトリガを送信しません。トリガを受信した時点から参加要求フレーム受信待ち時間（1200ms）の間、ノード番号、アドレスなどの重複チェック、参加ノード管理テーブルの更新をしながら、全ノードが参加要求フレームを送信するのを待ちます。トリガを受信した時点から参加要求フレーム送信待ち時間（ノード番号×4ms）経過後に、参加要求フレームを送信します。このとき、他ノードの参加要求フレームによってアドレスの重複を認識したノードは、領域1と2のコモンメモリ先頭アドレス、コモンメモリサイズを0にし、サイクリックデータは送信しません。アドレスの重複を認識したノードは、アドレス重複フラグを設定し、コモンメモリ・データ有効通知フラグをリセットします。参加要求フレーム受信待ち時間が終了した時点でノード番号が一番小さいノードが参加ノード管理テーブルに従い、最初にトークンを送信します。ノード番号の重複を認識したノードは、すべての送受信をしません。

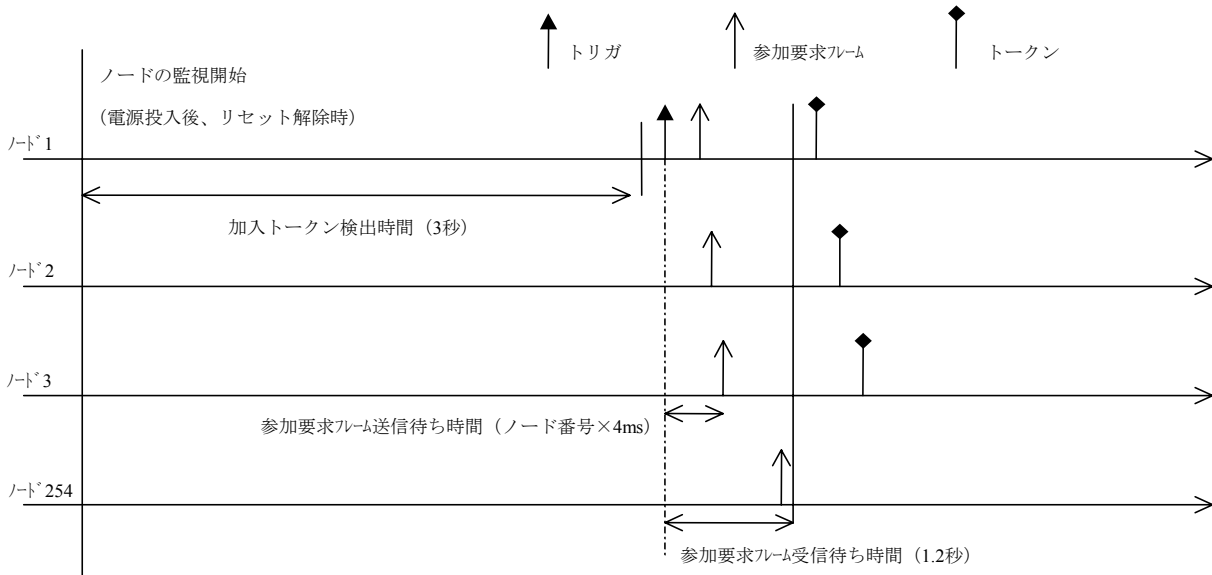


図9-19 立ち上げ時のタイムチャート1

(b) 途中参加

加入トークン検出時間内にトークンを受信するとすでにリンクが確立していると認識し、トークンが3周するまで参加要求フレームの送信を待ちます。その間受信したフレームによって、ノード番号、アドレスなどの重複チェックを行い、参加ノード管理テーブルを更新します。このとき、アドレスの重複を検出した場合、領域1と2のコモンメモリ先頭アドレス、コモンメモリサイズを0にし、サイクルリックデータは送信しません。アドレスの重複を認識したノードは、アドレス多重化のフラグを設定し、コモンメモリ・データ有効通知フラグをリセットします。ノード番号に異常がなかった場合、ノードは参加要求フレーム送信待ち時間経過後、参加要求フレームを送信します。参加要求フレームは、トークンの保持とは無関係に送信されます。ノード番号の重複を認識したノードは、参加要求フレームを送信せずにネットワークに参加しません。

(備考)

加入トークン検出時間 : ネットワークが稼働状態かチェックをするための時間です。

周回 : 周回の基準は、一番小さいノード番号宛トークンを受信したときを基準とします。

参加要求フレーム送信待ち時間 : 参加要求フレームの送出は、新規に参加する他ノードと重ならないように (自ノード番号×4ms) 経過後に送信します。

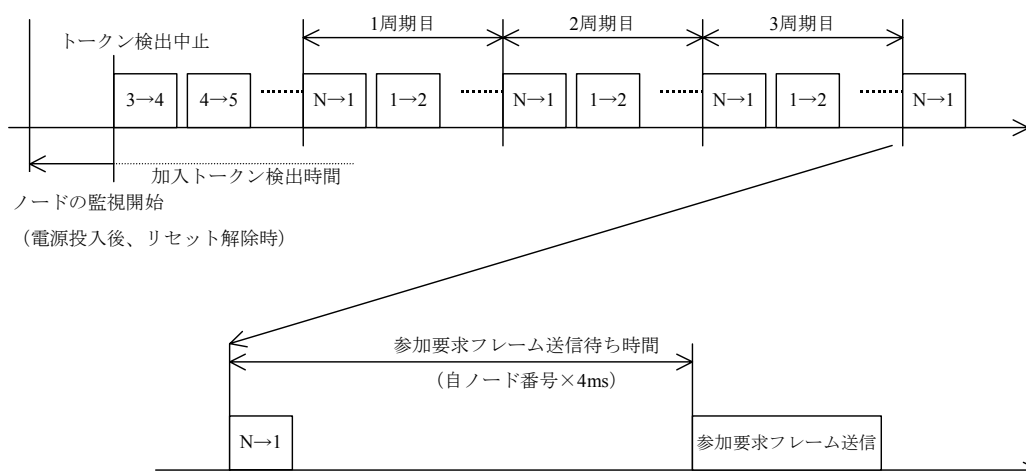


図9-20 立ち上げ時のタイムチャート2

(2) FL-netからの離脱

各ノードは、トークンフレーム受信ごとにノード番号をチェックし、3回連続してあるノードからのトークンフレームを受信しなければ、離脱したものとします (トークン保持ノードがトークン監視時間経過後もトークンを送出しない場合も含まれます)。

上記のようにノードがネットワークから離脱したと判断したとき、管理テーブルからそのノードの情報を削除します。

9 付 録

9. 4. 3 ノードの状態管理

ノードの状態管理は、自ノード管理テーブル、参加ノード管理テーブル、ネットワーク管理テーブルからなります。概要を表9-6に示します。

表9-6 ノードの状態管理のテーブル概要

名 称	内 容
自ノード管理テーブル	自ノードの設定について管理します。
参加ノード管理テーブル	ネットワークに加入しているノードに関する情報を管理します。
ネットワーク管理テーブル	ネットワークに共通する情報を管理します。

9. 4. 4 FL-netの自ノード管理テーブル

(1) 基本機能

自ノードの設定に関するデータを管理します。概要を以下に示します。

- 参加要求フレームや、ネットワークパラメータの読み出しに使用されます。
- 管理データは、ノードの立ち上げ時にFL-net上位層から設定されます。
- ノード名およびコモンメモリにおける送信領域の先頭アドレスとサイズを、ネットワークから設定できます。

(2) 管理データ

表9-7 自ノード管理テーブル

項 目	バイト長	内 容
ノード番号	1バイト	1~254
コモンメモリにおける 領域1・データ先頭アドレス	2バイト	ワード・アドレス (0~0x1FF)
コモンメモリにおける 領域1・データサイズ	2バイト	サイズ (0~0x1FF)
コモンメモリにおける 領域2・データ先頭アドレス	2バイト	ワード・アドレス (0~0x1FFF)
コモンメモリにおける 領域2・データサイズ	2バイト	サイズ (0~0x1FFF)
上位層の状態	2バイト	RUN/STOP/ALARM/WARNING/NORMAL
トークン監視時間	1バイト	1ms単位
最小許容フレーム間隔	1バイト	100 μ s単位
ベンダ名	10バイト	ベンダの名称
メーカー型式	10バイト	メーカーの型式、デバイスの名称
ノード名 (設備名)	10バイト	ユーザ設定によるノードの名称
プロトコルのバージョン	1バイト	0x80固定
FAリンクの状態	1バイト	加入/離脱など
自ノードの状態	1バイト	ノード番号重複検知など

9. 4. 5 FL-netの参加ノード管理テーブル

(1) 基本機能

ネットワークに参加しているノード状態は、各ノードが保持している管理テーブルによって監視されます。ネットワークに加入するノードに関してノード単位で管理するデータを扱います。概要を以下に示します。

- 立ち上がり時トークンフレームを受信し参加ノード管理テーブル、ネットワーク管理テーブルを更新します。
- トークンフレームの受信ごとに各ノードは参加ノード管理テーブルを更新します。
- 新規参加の参加要求フレームを受信すると参加ノード管理テーブルを更新します。
- 各ノードのトークンフレームの非受信またはタイムアウトを連続3回検出により、該当ノードをテーブルから削除します。

(2) 管理データ

各ノードのトークンを常時監視し参加ノード管理テーブルを作成して管理します。

表 9-8 参加ノード管理テーブル

項 目	バイト長	内 容
ノード番号	1バイト	1~254
上位層の状態	2バイト	RUN/STOP/ALARM/WARNING/NORMAL
コモンメモリにおける 領域1・データ先頭アドレス	2バイト	ワード・アドレス (0~0x1FFF)
コモンメモリにおける 領域1・データサイズ	2バイト	サイズ (0~0x1FFF)
コモンメモリにおける 領域2・データ先頭アドレス	2バイト	ワード・アドレス (0~0x1FFF)
コモンメモリにおける 領域2・データサイズ	2バイト	サイズ (0~0x1FFF)
リフレッシュサイクル許容時間	2バイト	1ms単位
トークン監視時間	1バイト	1ms単位
最小許容フレーム間隔	1バイト	100 μ s単位
リンクの状態	1バイト	加入/離脱情報など

「0x1FFF」は、16進数 (1FFF hex) を示します。

(備考) 受信するトークンフレームに含まれています。

9 付 録

9.4.6 FL-netの状態管理

- (1) 基本機能
ネットワークの状態に関するパラメータを管理します。
- (2) 管理データ

表9-9 ネットワーク管理テーブル

項 目	バイト長	内 容
トークン保持ノード番号	1バイト	現在トークンを保持しているノード
最小許容フレーム間隔	1バイト	100 μ s単位
リフレッシュサイクル許容時間	2バイト	1ms単位
リフレッシュサイクル測定時間 (現在値)	2バイト	1ms単位
リフレッシュサイクル測定時間 (最大値)	2バイト	1ms単位
リフレッシュサイクル測定時間 (最小値)	2バイト	1ms単位

9.4.7 FL-netのメッセージ通番管理

- (1) 基本機能
メッセージ伝送における通番と通番バージョン番号を管理します。
- (2) 送信用管理データ

表9-10 メッセージ通番管理の送信用管理データ

項 目	バイト長	内 容
通番バージョン番号	4バイト	送信メッセージ伝送の通番バージョン
通番 (1対N送信)	4バイト	0x1~0xFFFFFFFF
通番 (1対1送信)	4バイト×256	0x1~0xFFFFFFFF

「0xFFFFFFFF」は、16進数 (FFFFFFFF hex) を示します。

- (3) 受信用管理データ

表9-11 メッセージ通番管理の受信用管理データ

項 目	バイト長	内 容
通番バージョン番号	4バイト	0x1~0xFFFFFFFF
通番 (1対1受信)	4バイト	0x1~0xFFFFFFFF
通番 (1対N受信)	4バイト	0x1~0xFFFFFFFF

「0xFFFFFFFF」は、16進数 (FFFFFFFF hex) を示します。

9. 5 ネットワーク構成部品

9. 5. 1 イーサネットの構成部品一覧

図9-21にイーサネットを構成する部品を示します。

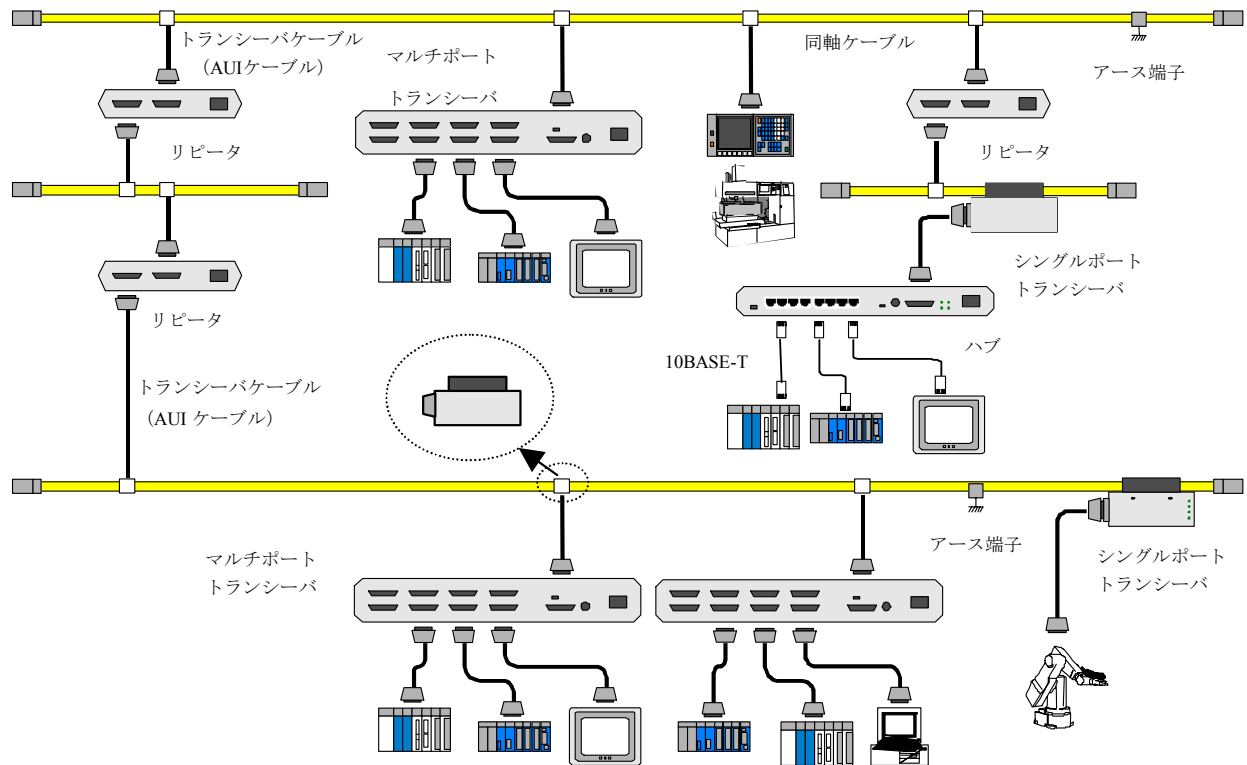


図9-21 イーサネットの構成部品一覧

表9-12 イーサネットの構成部品一覧

品 名	メーカ	形 式	備 考
FL.NETモジュール	(株) 日立製作所	LQE000	S10mini用
トランシーバ	日立電線 (株)	HLT-200	コネクタ形
		HLT-200TB	タップ形
		HBN200TZ	
		HBN200TD	
リピータ	日立電線 (株)	HLR-200H	同軸ケーブルの伝送距離延長用
マルチポート トランシーバ	(株) 日立製作所	H-7612-64	4ポート/8ポート
		H-7612-68	(AC電源内蔵タイプ)
同軸ケーブル	日立電線 (株)	HBN-CX-100	屋内用ケーブル長指定 (最長500m)
同軸コネクタ	日立電線 (株)	HBN-N-PC	同軸ケーブル用
中継コネクタ	日立電線 (株)	HBN-N-AJJ	同軸ケーブル用
ターミネータ	日立電線 (株)	HBN-T-NJ	J形
		HBN-T-NP	P形
アース端子	日立電線 (株)	HBN-G-TM	同軸ケーブル用
トランシーバケーブル	日立電線 (株)	HBN-TC-100	D-sub15ピンコネクタ付き ケーブル長指定 (最長50m)
ツイストペアケーブル	日立電線 (株)	HUTP-CAT5 4P	ケーブル長およびストレート/クロス指定 (最長100m)
マルチポートリピータ (ハブ)	アライド テレシス (株)	Centre COM 3624TRS	24ポート
12V電源	デンセイ ラムダ (株)	HK-25A-12	10BASE-5用12V

9.5.2 10BASE-5関連

(1) トランシーバ

トランシーバとは、同軸ケーブル（イエローケーブル）上に流れている信号をノードが必要とする信号に変換、またはその逆の変換をする装置です。

トランシーバを同軸ケーブルに取り付ける際には、2.5m間隔の整数倍で設置してください。同軸ケーブル上のマーキングに沿って設置してください。

トランシーバと同軸ケーブルは、ノードやトランシーバの電源供給装置の電源を停止してから接続してください。通電中に接続すると、ショートする場合があります。

使用するトランシーバは、IEEE802.3の規格にあったものを使用してください。

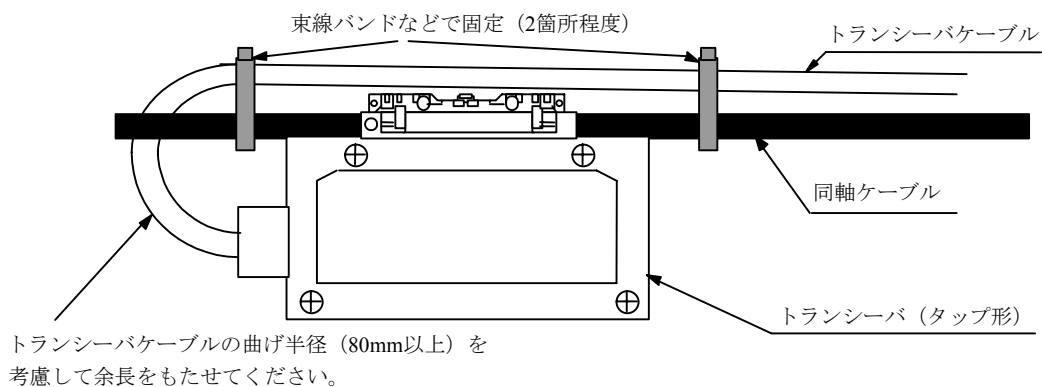


図9-22 トランシーバ概観

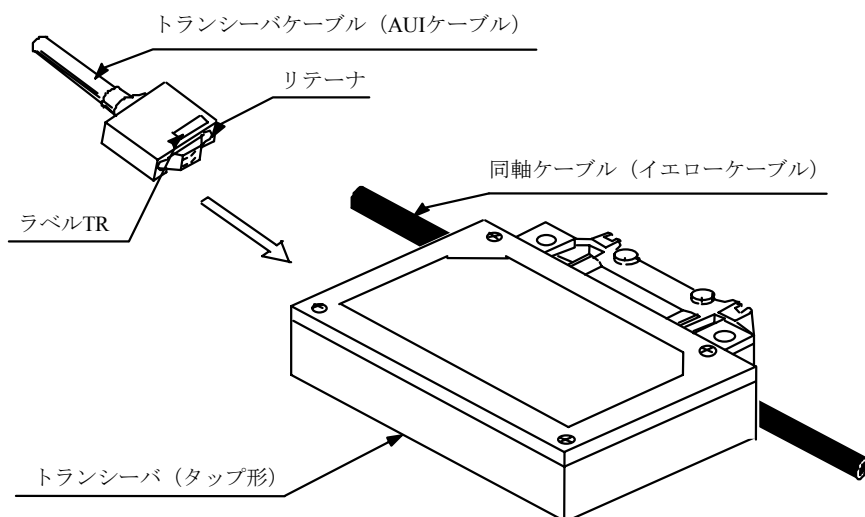


図9-23 トランシーバとトランシーバケーブル (AUI)

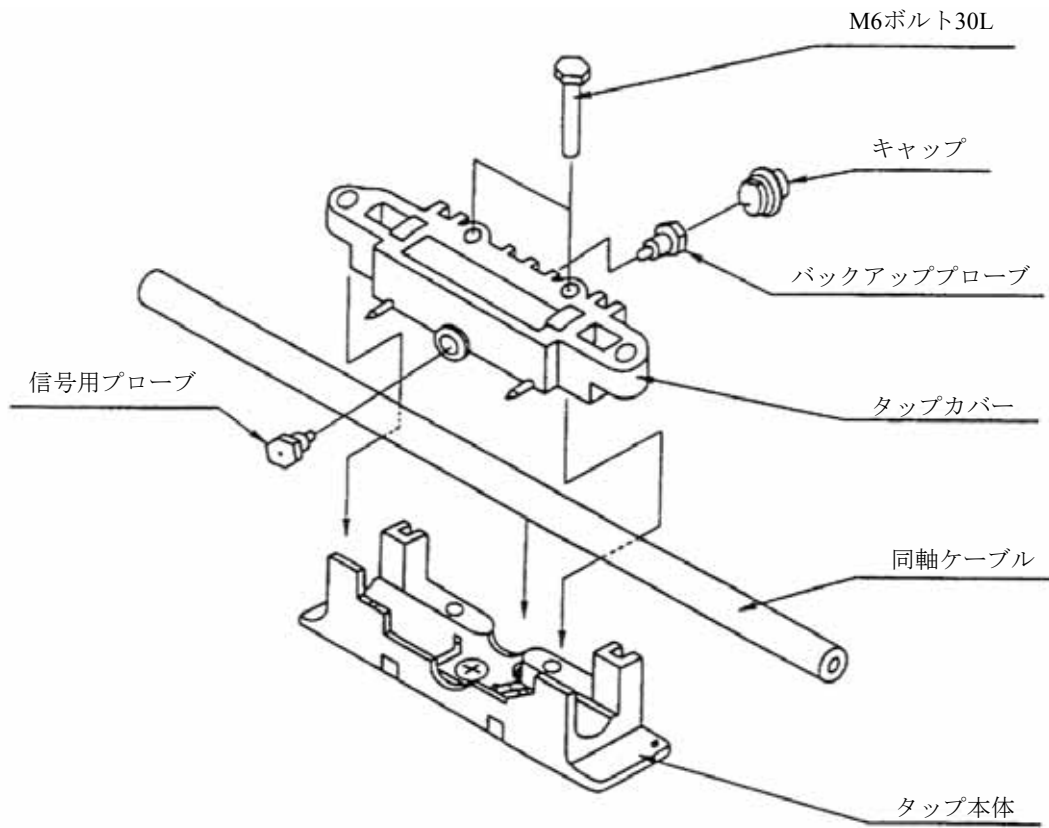


図9-24 タップと同軸ケーブルの取り付け

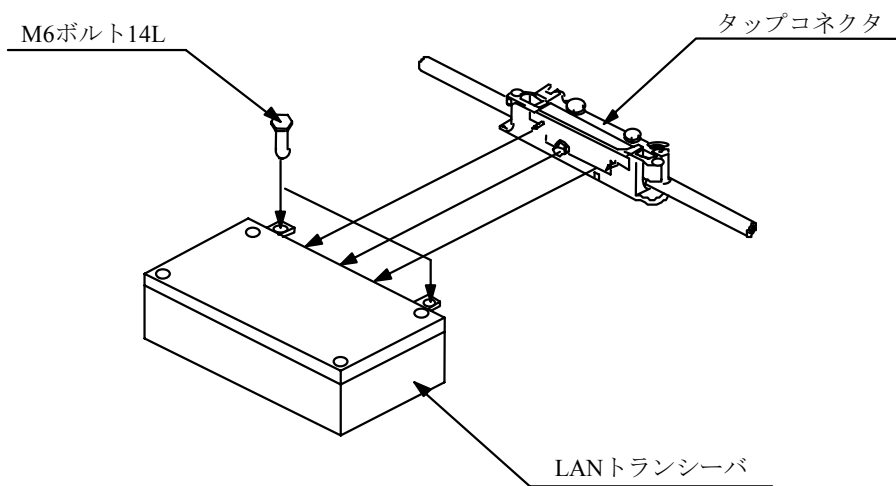


図9-25 タップとトランシーバ本体の取り付け

● トランシーバ (タップ形)

タップ型トランシーバは、同軸ケーブルに穴をあけ中心導体に接触する針を差し込むとともに、シールド導体に鱗の歯のような爪で絶縁ジャケットを破り接続します。接続には、専用工具が必要です。

トランシーバの電源 (DC12V) は、トランシーバケーブルを経由してノードから供給します。なお、ノードによっては、トランシーバケーブルを使用する際に、DC12Vの電源を必要とするタイプもあります。詳細は、ノードのハードウェアマニュアルを参照してください。

トランシーバの「SQE」スイッチは、以下のように設定します。

ノードに接続時 : ON

リピータに接続時 : OFF

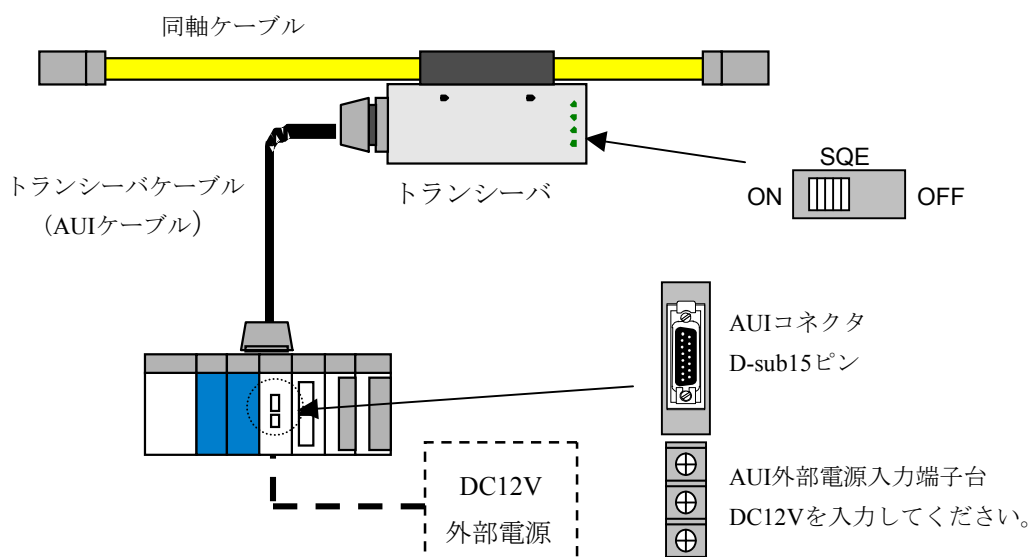


図 9-26 イーサネットのトランシーバ (タップ形)

● トランシーバ (コネクタ形)

コネクタ型トランシーバは、同軸ケーブルにコネクタを取り付け、そのコネクタとトランシーバのコネクタを接続します。

接続には、専用工具が不要で簡単に取り付けや取り外しができます。

トランシーバの電源は、トランシーバケーブルを経由してノードから供給します。

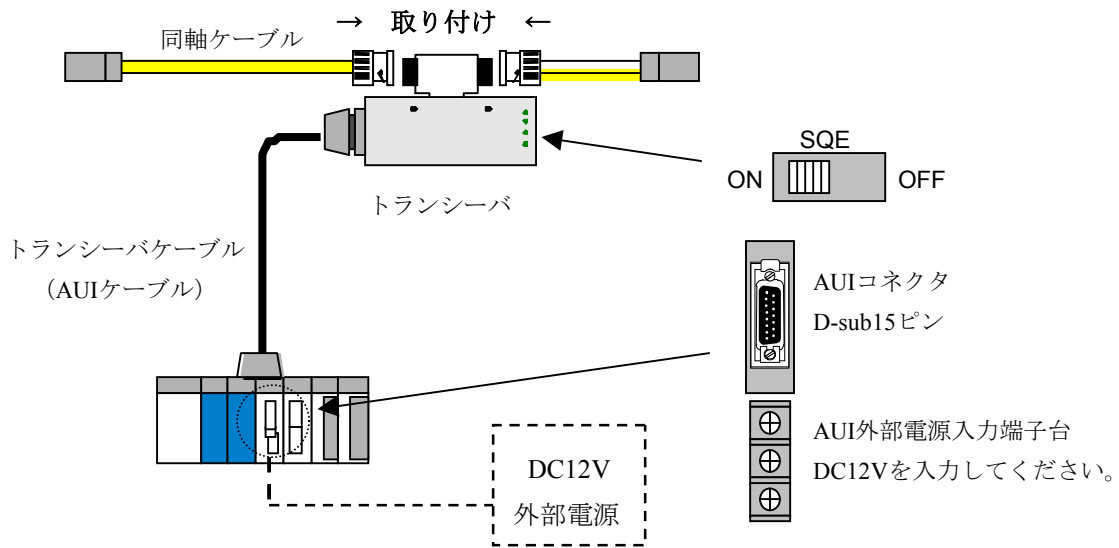
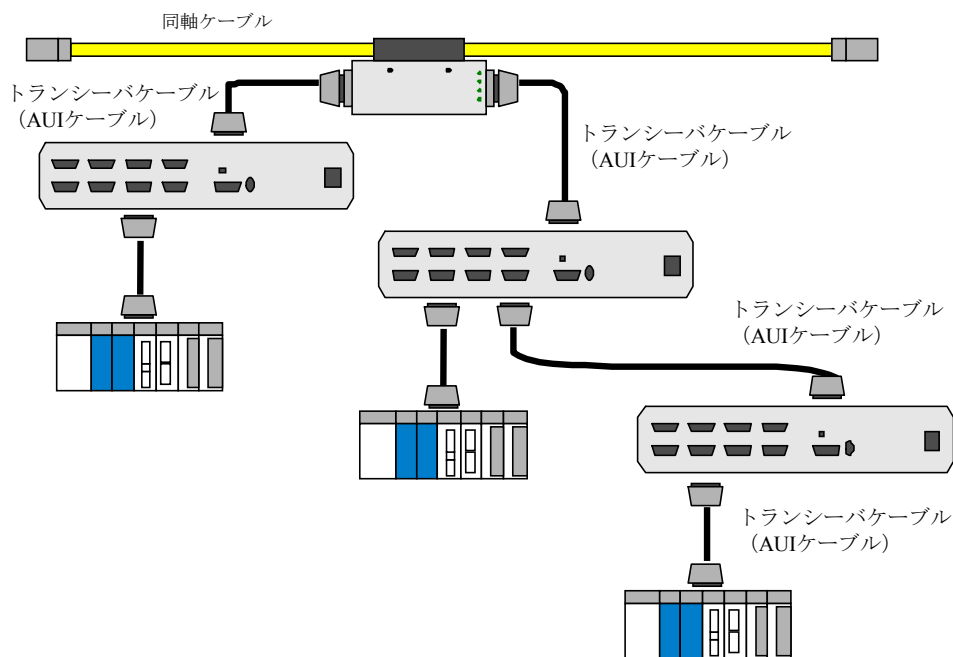


図 9-27 イーサネットのトランシーバ (コネクタ形)

● マルチポートトランシーバ

タップ型トランシーバ、コネクタ型トランシーバでは、1つのトランシーバに対して接続できる端末数が1台であるものを、複数台接続できるようにしたトランシーバです。4ポート、8ポートトランシーバなどがあります。



トランシーバの電源は、電源ケーブルを接続して供給します。

図 9-28 イーサネットのマルチポートトランシーバ

● リピータ

リピータとは、伝送信号の再中継を行う装置で、異なるメディアセグメント間の相互接続や、メディアセグメントの距離延長、接続端末台数の増加、ケーブルメディアの変換に使用する装置です。リピータは、相互接続された片方のセグメントから受け取った信号を波形整形し、決められたレベルに増幅して、リピータに接続されたすべてのセグメントに送出（リピート）します。リピータに接続できるトランシーバケーブル長は、最大50mですが、ノイズ防止などを考慮し2m以下にすることを推奨します。また、SQEスイッチはOFFに設定に注意してください。

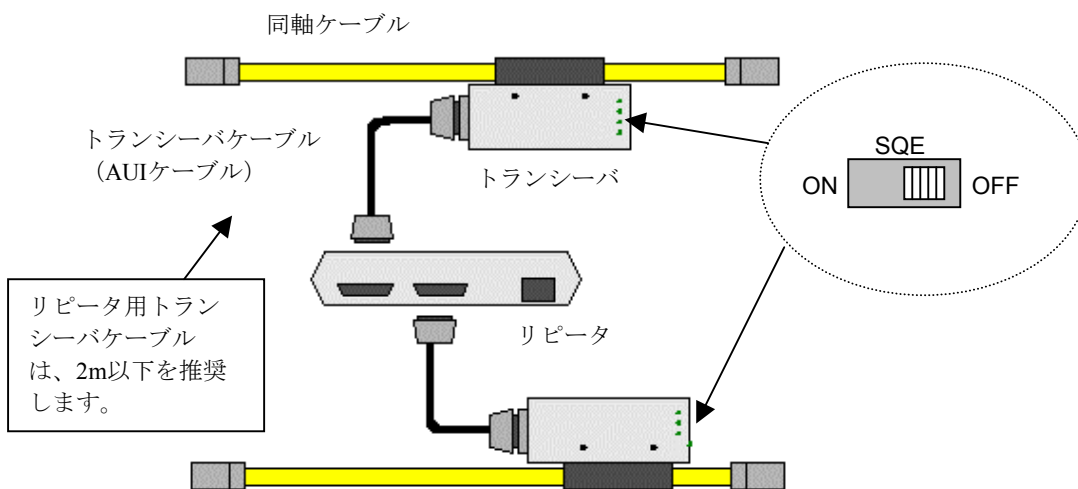


図9-29 イーサネットのリピータ

(2) 同軸ケーブル

同軸ケーブルは、中心導体と、シールドとして作用する外部導体とで構成されたケーブルです。イーサネットの接続に用いられる同軸ケーブルは、50Ωのインピーダンスで、10BASE-2用のRG58A/U、10BASE-5（通称イエローケーブル）があります。

10BASE-2ケーブルは最長185m、10BASE-5ケーブルの最長500mです。

なお、同軸ケーブルを使用する場合には、ノイズ防止のため必ずアース接続（1点アース接地かつD種接地）してください。

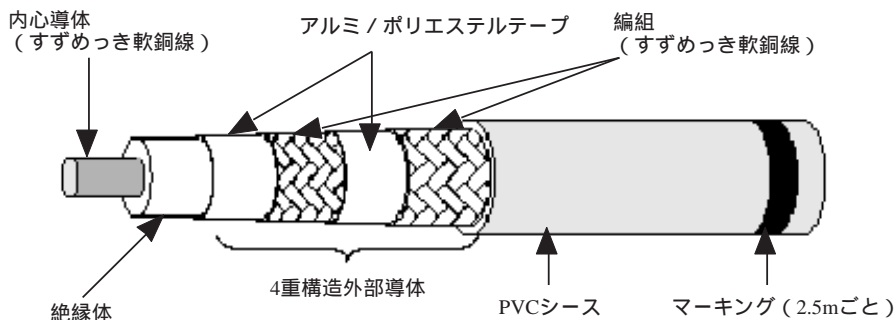


図9-30 イーサネットの同軸ケーブル

(3) 同軸コネクタ

同軸コネクタは、通称N型コネクタとも呼ばれ同軸ケーブルと終端装置や、コネクタ型トランシーバを接続するときに使用するコネクタです。



図9-31 イーサネットの同軸コネクタ

(4) 中継コネクタ

同軸ケーブル間を延長するためのコネクタです。リピータはセグメントを延長する場合に使用するのに対し、中継コネクタは同じセグメント上のケーブルを延長するために使用します。

中継コネクタを複数接続すると、同軸ケーブルの電気抵抗が変化する場合があるので注意してください（使用しないことを推奨します）。



図9-32 イーサネットの中継コネクタ

(5) ターミネータ（終端抵抗）

バス型配線時において、信号の反射を防ぐためにケーブルの両端に接続する装置で、必ず接続してください。終端装置の接続をしない場合、信号の反射（衝突）が発生しネットワークダウンとなります。終端装置には、トランシーバがタップ型時に使用するJ形、コネクタ型時に使用するP形があります。終端装置は、同軸ケーブル上のマーキングのところに設置してください。



図9-33 イーサネットのターミネータ（終端抵抗）

(6) 同軸ケーブルアース端子

同軸ケーブル上のノイズにより、通信データエラーを予防するための装置です。同軸ケーブル上に必ず1点接地してください。なお、アースはD種接地をしてください。



図9-34 イーサネットの同軸ケーブルアース端子

(7) トランシーバケーブル

トランシーバとノードを接続するためのケーブルです。トランシーバケーブルは、両端にD-sub15ピンのAUIコネクタが装着されています。トランシーバケーブルとして使用できる最長は50mですが、FA現場ではノイズ防止などを考慮して15m以下のケーブルを使用することを推奨します。

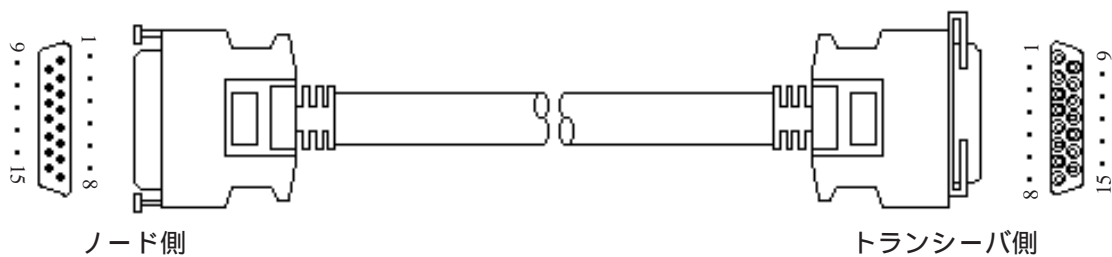


図9-35 イーサネットのトランシーバケーブル

(8) 10BASE-5/T変換器

10BASE-5のインタフェースを持つケーブルを10BASE-Tに接続するための変換器です。

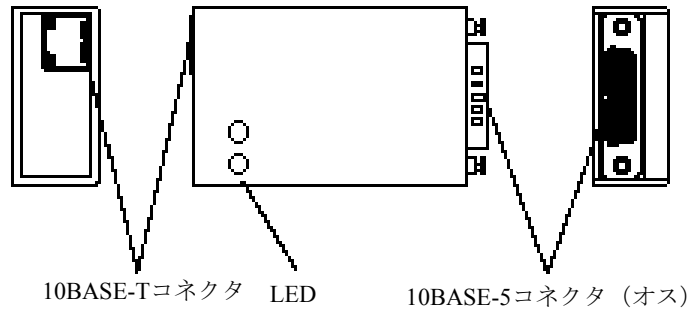


図9-36 イーサネットの10BASE-5/T変換器

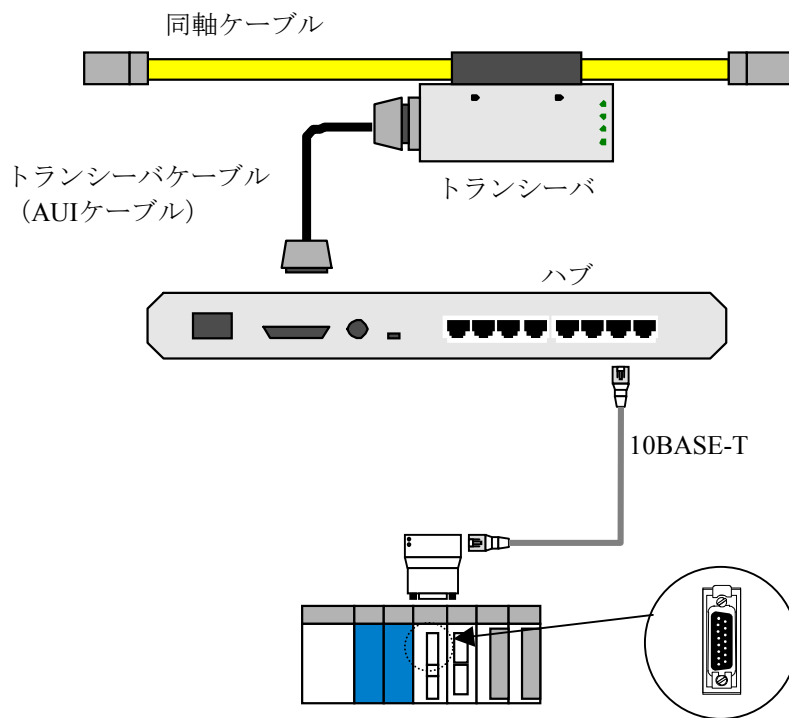


図9-37 イーサネットの10BASE-5/T変換器取り付け

(9) 同軸／光変換メディアコンバータ・リピータ

同軸／光変換メディアコンバータ・リピータとは、同軸ケーブル上（10BASE-5/10BASE-2）の電気信号を光信号に変換するための機器です。リピータ間を接続するためのFOIRL（Fiber Optic Inter Repeater Link）や端末と接続するための10BASE-FLなどがあります。同軸／光変換メディアコンバータ・リピータは、ノイズ防止やケーブル延長に使用します。

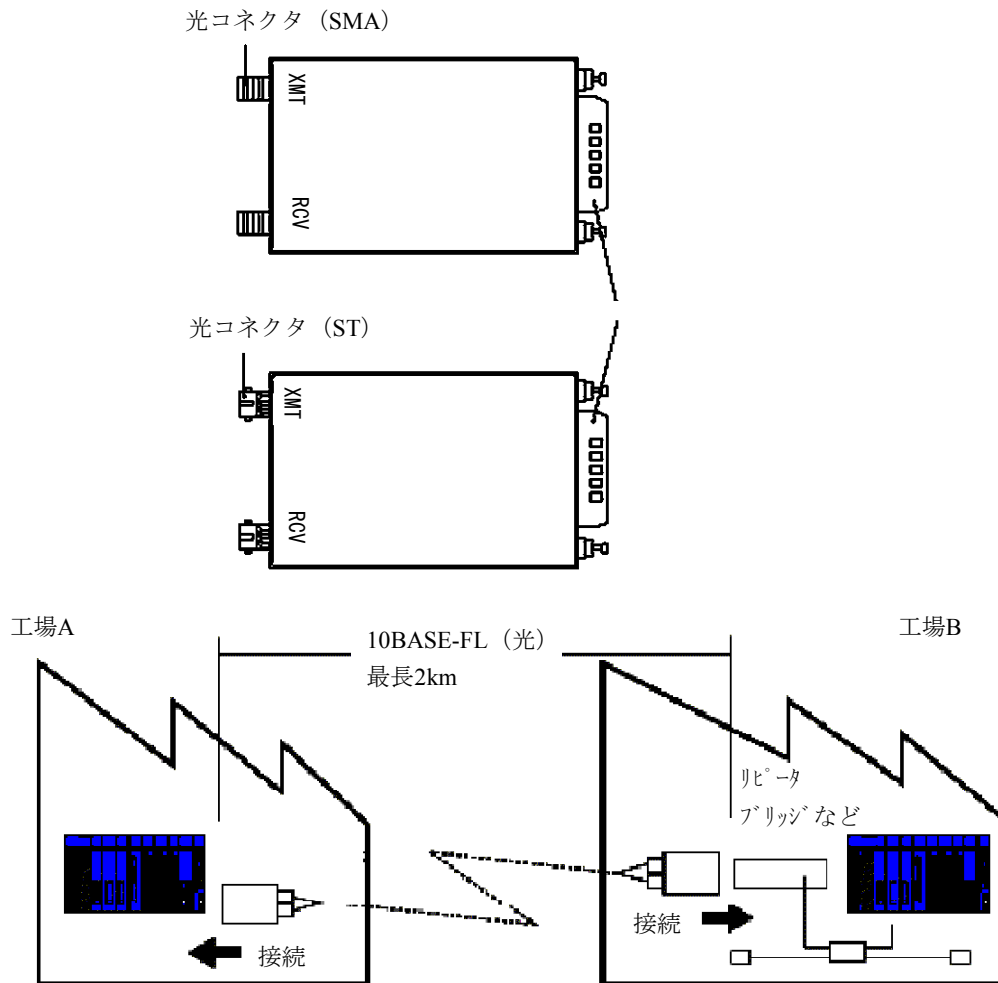


図9-38 イーサネットの同軸／光変換メディアコンバータ・リピータ

9.5.3 10BASE-T関連

(1) ハブ

10BASE-Tで使用するツイストペアケーブルを収容するためのリピータ機能を持った集線装置を指します。

ハブには、10BASE-2のインタフェースを持ったものやカスケード（多段階接続）するためのインタフェースを持ったものなど、複数の種類があります。なお、ハブは、最大4つまでカスケードできますが、複数のハブを1つのハブとして使用できる、スタックابلハブもあります。

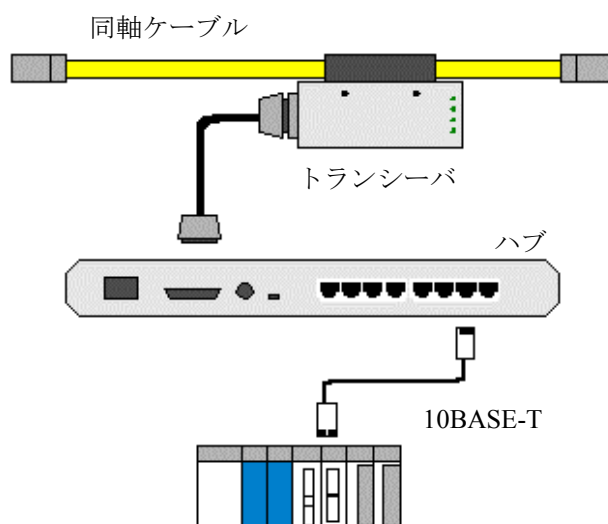


図9-39 イーサネットのハブ

(2) 10BASE-Tケーブル

ツイストペアケーブルまたはより対線とも呼ばれ銅線を2本1組でより線とし、それを何組かまとめて外部保護カバーで覆ったものです。ケーブルには以下の種類があります。

- ・STPケーブル（シールド付き）
- ・UTPケーブル（シールドなし）
- ・クロスケーブル（ノード間を直接接続）
- ・ストレートケーブル（ハブを経由して接続）

10BASE-Tケーブルにおける伝送速度の最大値は、10Mbpsです。ケーブルの最長は、100mです。ケーブル両端の接続用コネクタは、ISO8877で規定されている8極モジュラコネクタを使用します。なお、10BASE-Tケーブルは、カテゴリ5準拠の製品を使用してください。

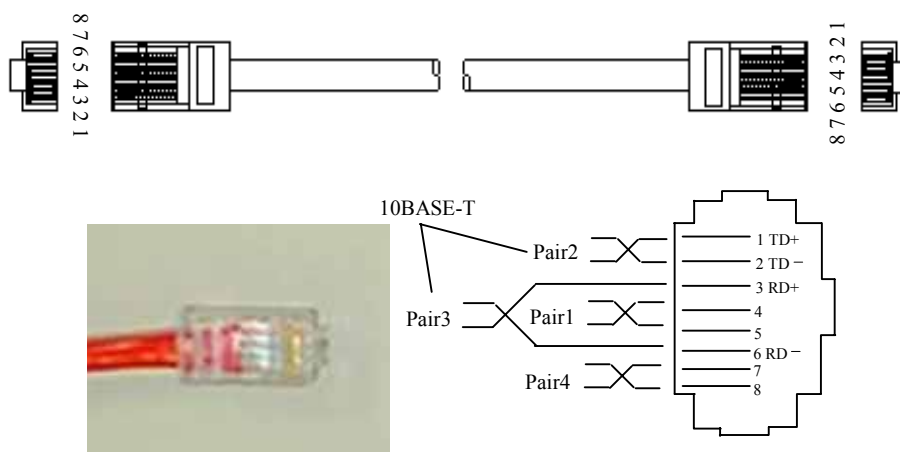


図 9-40 イーサネットの10BASE-Tケーブル

(3) 10BASE-T／光変換メディアコンバータ・リピータ

10BASE-T／光変換メディアコンバータ・リピータとは、10BASE-Tケーブル上の電気信号を光信号に変換するための機器です。

リピータ間を接続するためのFOIRL（Fiber Optic Inter Repeater Link）や端末と接続するための10BASE-FLなどがあります。10BASE-T／光変換メディアコンバータ・リピータは、ノイズ防止やケーブル延長に使用します。

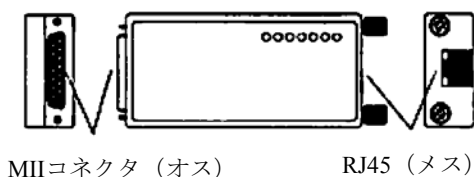


図 9-41 イーサネットの10BASE-T／光変換メディアコンバータ・リピータ

9.6 FL-netのネットワーク施工方法

9.6.1 10BASE-5同軸ケーブルの配線

(1) ケーブルの布設配線

ケーブルの布設配線方法は、場所によりいくつかの取り付け方法が考えられますが、その主なものは以下のとおりです。

- ・壁面露出配線
- ・フリーアクセス、床ビット内配線
- ・ケーブルラック内配線
- ・天井内コロガシ配線

(2) 布設配線工事上の留意事項

布設配線工事上の留意事項は以下のとおりです。

- ・10BASE-5同軸ケーブルは、原則として屋内に布設、配線されるものです。
- ・壁面などへの固定はケーブル自重によるストレスがかかるため、特殊な場合を除き約1mの間隔にて固定します。その際、ケーブルが変形しないようにしてください。
- ・ケーブルラック、天井にケーブルを固定する場合の固定間隔はケーブルが変形しないようにしてください。
- ・床下または床際にケーブルを配線する場合は、歩行または器物によりケーブルに変形、損傷を受けやすいので保護してください。
- ・ケーブルの外部導体は保安上、接地することが望ましい。
- ・接地する場合は、1セグメントの1点で接地を行い、D種接地以上とします。
- ・接地点以外のケーブルの金属露出部分が大地や外の金属部分に接触しないようにN型コネクタ、L型コネクタ、直線スリーブ、ターミネータは付属のブーツを被せるか、絶縁テープを巻き絶縁してください。
- ・パワーケーブル（AC100V以上）との隔離距離は60cm以上取ってください。

(3) 同軸ケーブルの布設に関わる主な諸元

布設に関わる同軸ケーブルの主な諸元を表9-13に示します。

表9-13 同軸ケーブルの諸元

項目	仕様
布設時	半径100mm以上
固定時	半径100mm以上
張力	最大245N
ケーブル質量	188kg (1km単位)

(4) 10BASE-5同軸ケーブル

S10mini FL.NETモジュールの推奨ケーブルは、「9.5.1 イーサネットの構成部品一覧」を参照してください。

(5) 同軸コネクタの取り付け

(a) 適用コネクタ

S10mini FL.NETモジュールの推奨ケーブルは、「9.5.1 イーサネットの構成部品一覧」を参照してください。

(b) 加工手順

同軸コネクタ (N-PC) の取り付け手順を以下に示します。

① PVCシース剥ぎ



図9-42 同軸ケーブル被覆 (PVCシース) 剥ぎ

② アルミテープ除去

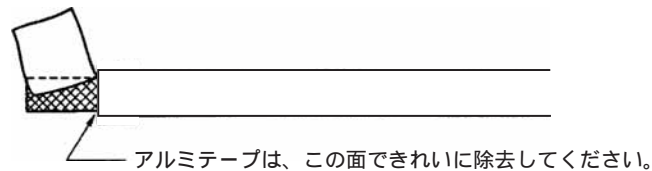


図9-43 同軸ケーブルアルミテープ除去1

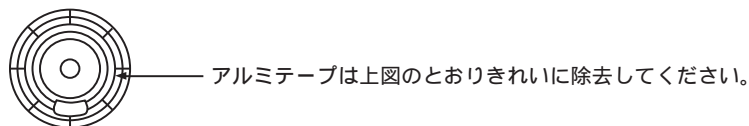


図9-44 同軸ケーブルアルミテープ除去2

③ 絶縁体剥ぎ

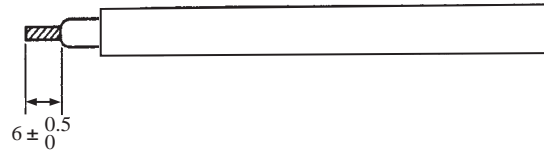


図 9-45 同軸ケーブル絶縁体剥ぎ

④ 部品組み込みおよびシールド処理

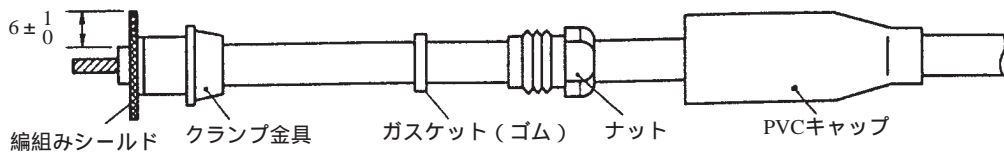


図 9-46 同軸ケーブルの部品組み込みおよびシールド処理

⑤ ピンコンタクトはんだ付け

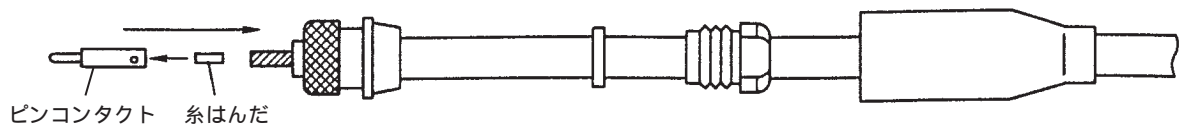


図 9-47 同軸ケーブルのシールド処理およびピンコンタクトはんだ付け

⑥ コネクタ組み立て

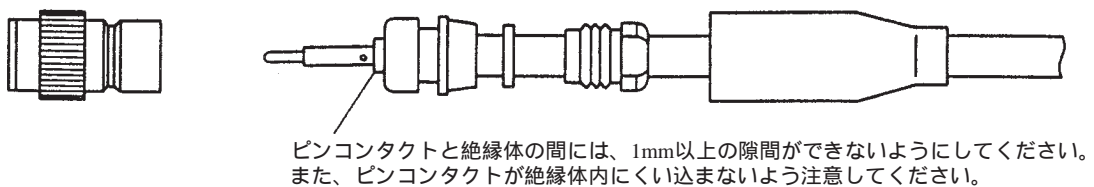


図 9-48 同軸ケーブルのコネクタ組み立て

(6) トランシーバ

(a) トランシーバ（タップ形）の設置・取り付け

トランシーバの設置場所および取り付け方法は現場の状況によりさまざまですが、主な設置場所は次のようなところが考えられます。

- ・ 壁面
- ・ 床下（フリーアクセス、ビット内）
- ・ 天井内
- ・ ケーブルラック上
- ・ ステーションの側

トランシーバを取り付ける上での留意事項は以下のとおりです。

- ・ トランシーバは、取り付け脚が付いた状態で据置形として、また木ねじなどにて固定してください。
- ・ 天井内、床下にトランシーバを設置する場合は、保守点検が容易な場所に設置してください。
- ・ トランシーバの取り付け間隔は、2.5mとしてください（同軸ケーブルに付けてある2.5mごとのマーキング部に取り付けます）。

(b) 適用トランシーバ

S10mini FL.NETモジュールの推奨ケーブルは、「9.5.1 イーサネットの構成部品一覧」を参照してください。

(c) 加工・取り付け手順

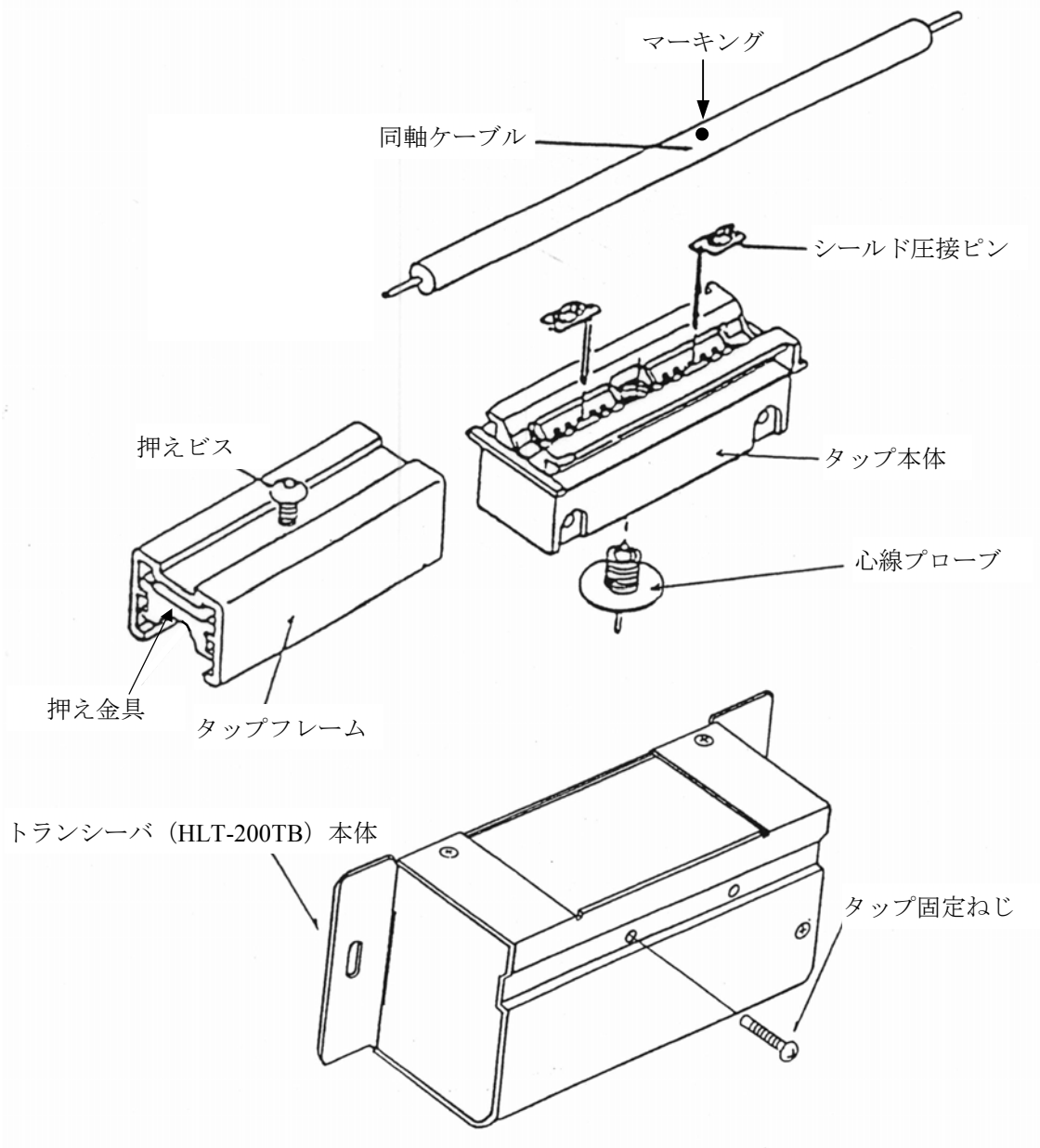


図9-49 トランシーバ各部の名称

- ① シールド圧接ピンをタップ本体へ挿入します。

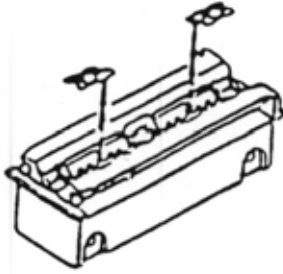


図 9-50 トランシーバのシールド圧接ピンをタップ本体挿入

- ② 押えビスを外れない程度に緩めます。

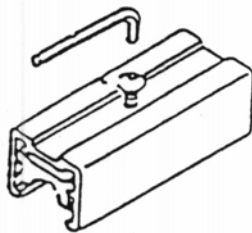


図 9-51 トランシーバのタップフレームの押えビス緩め

- ③ タップ本体を同軸ケーブルの2.5mごとのマーキングに合わせます。フレームをスライドさせて挿入し、押えビスを締めて固定します。このとき、タップ本体の上面と押え金具の隙間が、約1mm程度になるまで締めます。

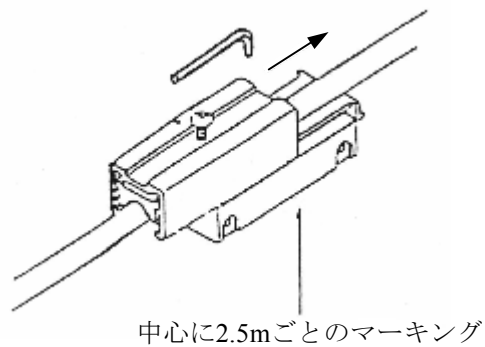


図 9-52 トランシーバのタップフレームとタップ本体装着

タップフレームは、同軸ケーブルがシールド圧接ピンの中央位置になるよう注意して挿入してください。ある程度まで締めて、押え金具が極端に傾いているようなときは、押えビスを緩めて、再度中央位置にしてから締め直してください。

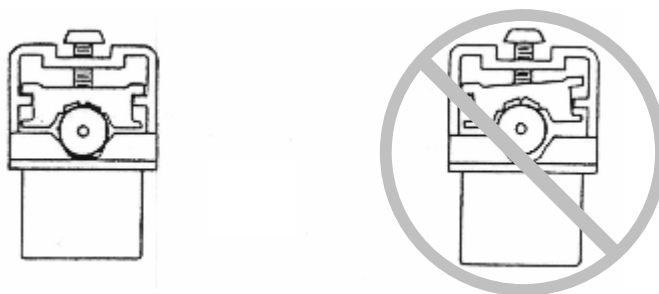


図 9-53 トランシーバのタップフレームと同軸ケーブルの挿入

- ④ 同軸ケーブルの心線プローブ用の穴を、ドリルで白い絶縁物が見えるまで開けます。押えビスが熱いと、アルミテープが残ることがあるので注意してください。また、穴に入ったシールド屑は取り除いてください。

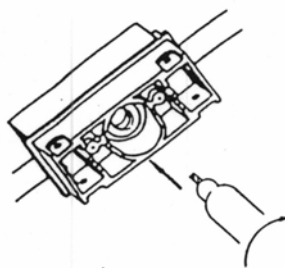


図 9-54 同軸ケーブルの心線プローブ用の穴あけ

- ⑤ 心線プローブを専用取り付けスパナで締めます。

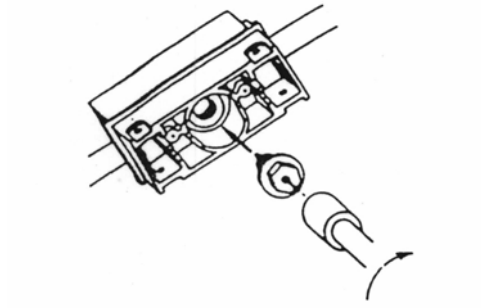


図9-55 同軸ケーブルへの心線プローブ取り付け

以上で、タップコネクタの取り付けが終了です。

[参考] 正しく接続されているかの試験方法を以下に示します。

- ・シールド圧接ピン間は、ショートしている。
- ・同軸ケーブルの両端末にターミネータが付いているときの、心線プローブとシールド圧接ピン間は 25Ω である。

ただし、すでにシステムが動作している場合は、上記の試験は決して行わないでください。システムの誤動作の原因になります。

- ⑥ タップコネクタにトランシーバ本体を取り付けます。このとき、シールド圧接ピンおよび心線プローブは垂直になるように取り付けてください。

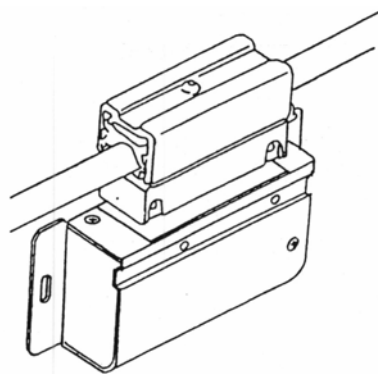


図9-56 タップコネクタへのトランシーバ本体挿入

- ⑦ 挿入した後、シールド圧接ピンおよび心線プローブが曲がっていると思われる場合は、一度引き抜きます。正確に入っていない場合は、目視できるくらいに著しく曲がるので、もう一度心出しをします。タップ固定ねじをトランシーバ本体上部の穴から挿入して、締めます。

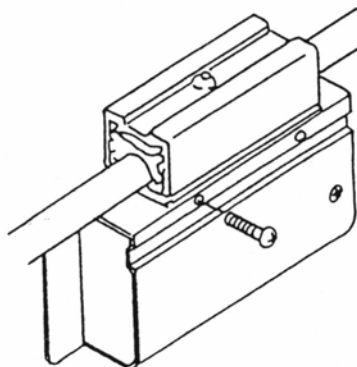


図9-57 トランシーバ本体とタップの固定

(d) SQEスイッチの設定

SQEスイッチの設定を、表9-14に示します。

表9-14 SQEスイッチの設定

項 目	設 定
ノードに接続時	ON
リピータに接続時	OFF
光リピータに接続時	OFF
ルータに接続時	ON
マルチポートトランシーバに接続時	OFF
ブリッジに接続時	OFF
ハブ (マルチポートリピータ) に接続時	OFF

(7) トランシーバ (コネクタ形) の設置・取り付け

「(6) (a) トランシーバ (タップ形) の設置・取り付け」と同じです。

(a) 適用トランシーバ

S10mini FL.NETモジュールの推奨ケーブルは、「9.5.1 イーサネットの構成部品一覧」を参照してください。

(b) 加工・取り付け手順

① 同軸ケーブルの加工

同軸ケーブルへの同軸コネクタ取り付け手順は、「(5) 同軸コネクタの取り付け」を参照してください。

- ② トランシーバ本体に同軸コネクタを取り付けます。このとき、トランシーバ本体が外れないようにねじ止めしてください。また、ターミネータにはゴムブーツを被せ絶縁してください。

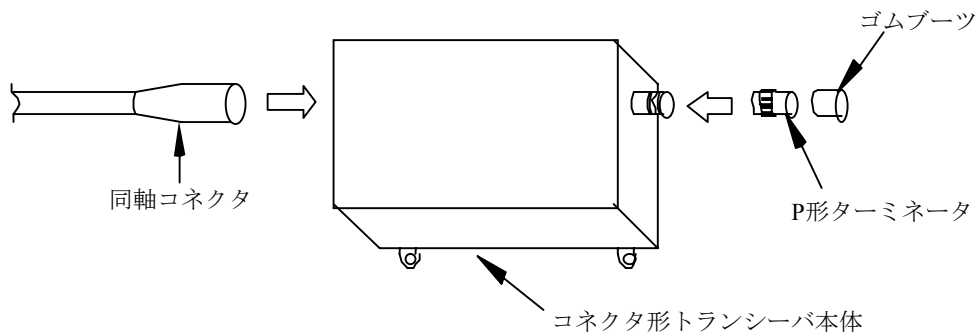


図 9-58 コネクタ形トランシーバ取り付け

(c) SQEスイッチの設定

「表 9-14 SQEスイッチの設定」を参照してください。

(8) リピータの設置・取り付け

① リピータの接続

トランシーバケーブルは、必ずリピータの電源を切ってから接続してください。

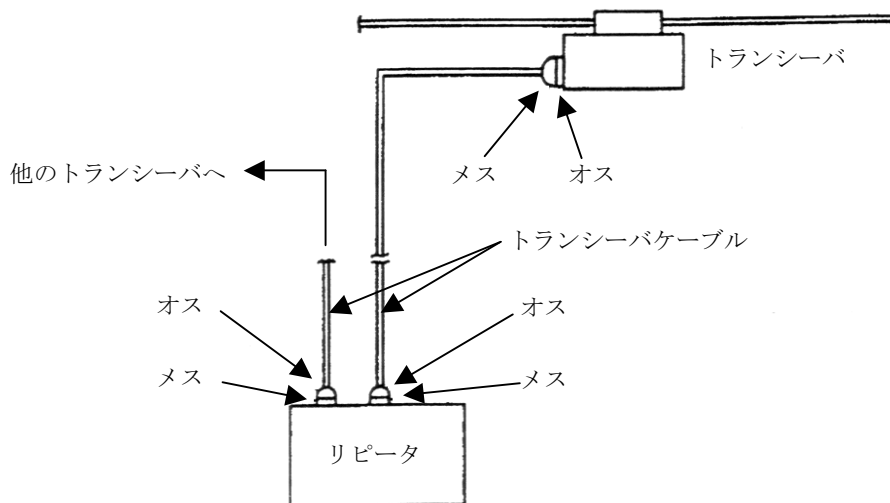


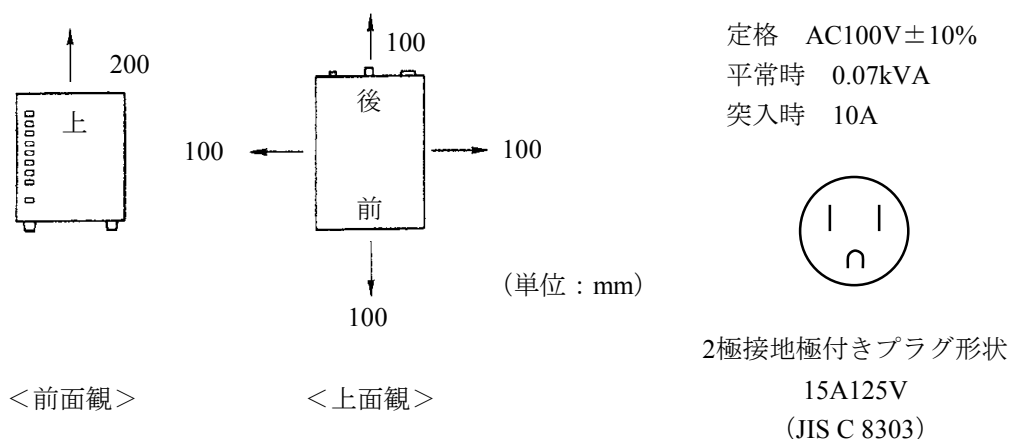
図 9-59 リピータ接続

② 適用リピータ

S10mini FL.NETモジュールの推奨ケーブルは、「9.5.1 イーサネットの構成部品一覧」を参照してください。

③ 設置場所とスペース

- ・リピータを設置する場所は、ワークステーション（サーバ）付近で容易に保守できる場所（一般事務室で天井裏、地下などは不可）を選び、前後、左右、上方に少なくとも下記スペースを確保してください。なお、リピータはAC電源が必要です。接地付きコンセントを用意してください。



(注) 前面にはできるだけスペースを確保してください。

図9-60 リピータの設置スペース

- ・ちりやほこりの多いところでは使用しないでください。
- ・底面に空気の取り込み口、上面に吹き出し口があります。ふさがないでください。
- ・リピータの設置場所付近には保守を考慮し電話を取り付けることを推奨します。
- ・誤って電源を切ることのないよう独立した電源を使用してください。リピータの電源が切れると通信ができません。

④ SQEスイッチの設定

「表9-14 SQEスイッチの設定」を参照してください。

(9) ターミネータ、コネクタの絶縁

中継コネクタ、L型コネクタの絶縁方法を以下に示します。

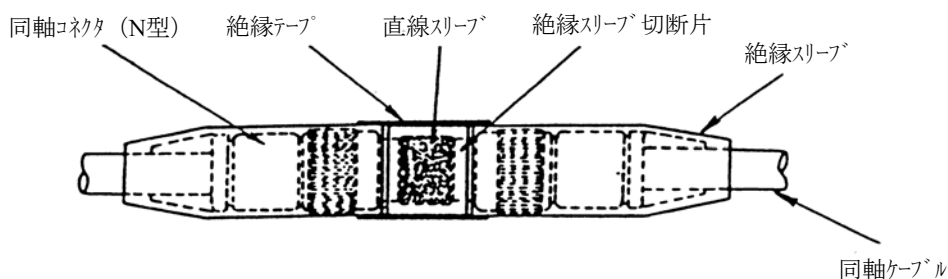


図9-61 中継コネクタの絶縁

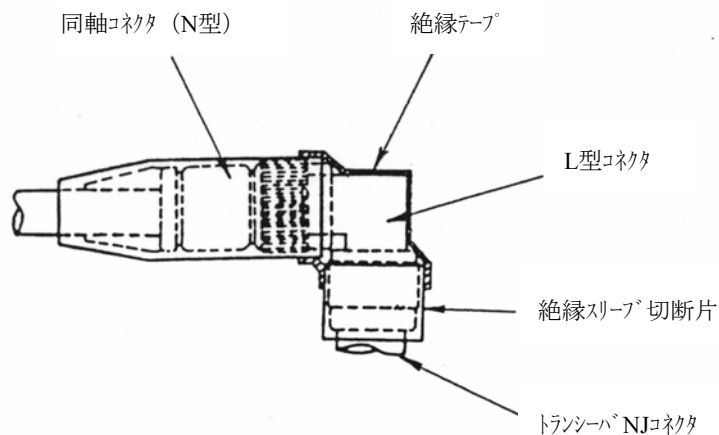


図9-62 L型コネクタの絶縁

ターミネータ (T-NPオス)、(T-NJメス) の絶縁方法を以下に示します。

- ・オス T-NP は絶縁スリーブ (黒色) (I-NPC) を被せます。
- ・メス T-NJ は絶縁スリーブ (黒色) (I-NJP) を被せ、絶縁テープで固定します。

(10) 適用ターミネータ (終端抵抗)

S10mini FL.NETモジュールの推奨ケーブルは、「9.5.1 イーサネットの構成部品一覧」を参照してください。

(11) トランシーバケーブルの取り付け

トランシーバおよびトランシーバケーブルの設置例を以下に示します。

< 壁面設置例 (1) >

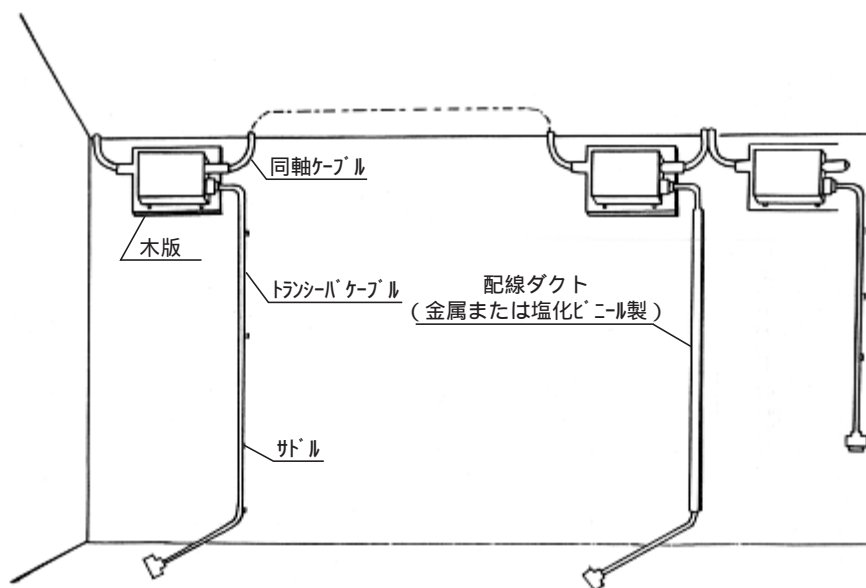
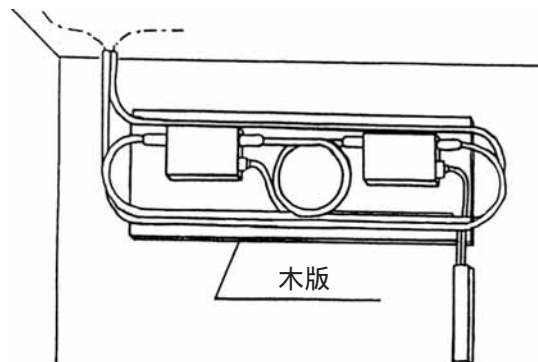
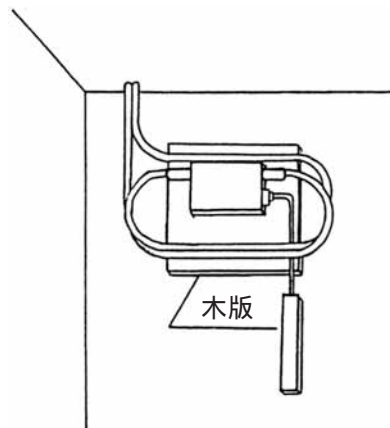


図9-63 トランシーバおよびトランシーバケーブルの壁面設置例1

< 壁面設置例 (2) >



< 壁面設置例 (3) >



< 壁面設置例 (4) >

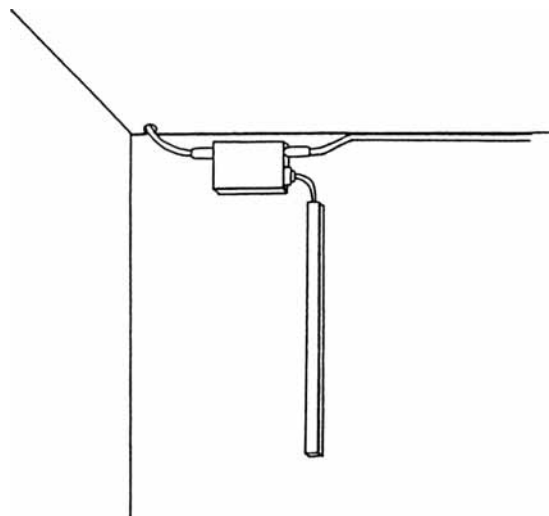
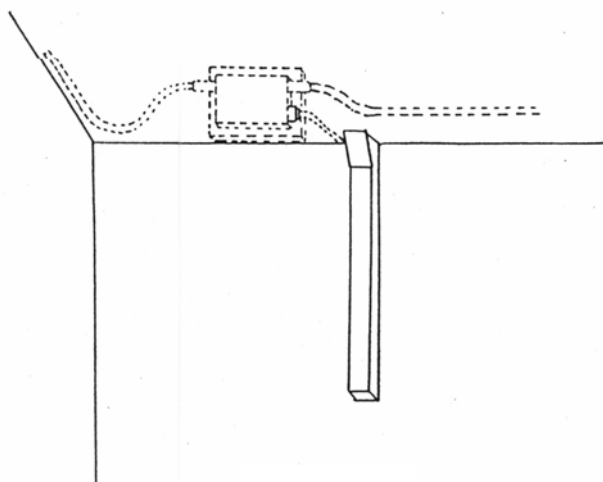


図 9-64 トランシーバおよびトランシーバケーブルの壁面設置例2

<天井内設置例>



<床下内設置例>

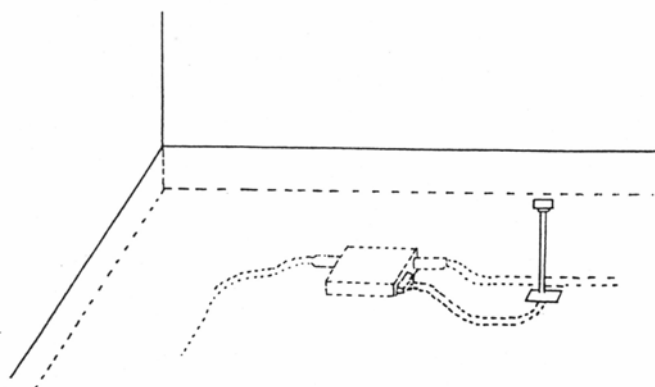


図9-65 トランシーバおよびトランシーバケーブルの天井内、床下内設置例

(a) 適用トランシーバケーブル

S10mini FL.NETモジュールの推奨ケーブルは、「9.5.1 イーサネットの構成部品一覧」を参照してください。

(12) 同軸ケーブルのアース端子の取り付け

(a) 適用アース端子

S10mini FL.NETモジュールの推奨ケーブルは、「9.5.1 イーサネットの構成部品一覧」を参照してください。

(b) 加工・取り付け手順

同軸ケーブルのアース端子取り付け方法を以下に示します。同軸ケーブルはアース端子（G-TM）を用い、1点アース（D種接地以上）を取ってください。

同軸ケーブルは任意の1点でアースを取ってください。

① 挿入つめを本体に挿入します。

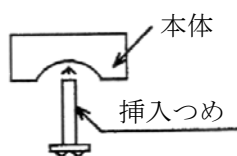


図9-66 アース端子本体への挿入つめ取り付け

② アース端子を同軸ケーブルに取り付けて、M4のビスを交互に締め付けます。このとき圧着端子はどちらかのビスに取り付けます。同軸セグメント上の位置はアースが取り付けやすい任意の1箇所のみとしてください。

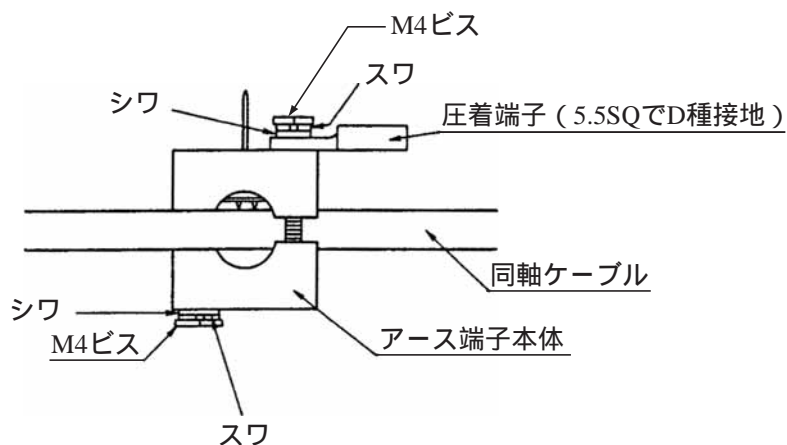


図9-67 アース端子取り付け

- ③ 締め付け後、挿入つめの余分を切断します。

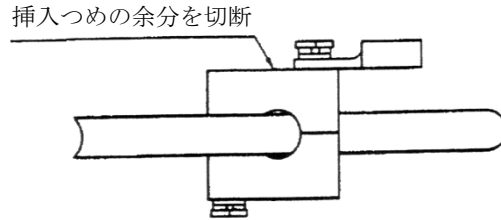


図 9-68 挿入つめの余分を切断

- ④ 1点アース (D種接地) を取ってください。

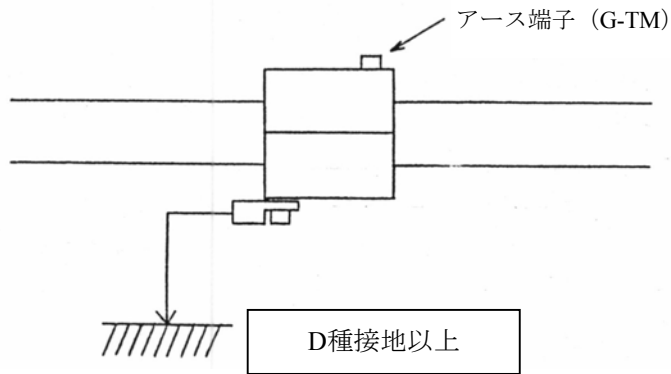


図 9-69 同軸ケーブルのアース端子取り付け

9. 6. 2 10BASE-T (UTP) ケーブル

(1) 10BASE-T (UTP) ケーブルの作成方法

① 10BASE-T (UTP) ケーブルのシース（被覆）剥ぎ

シースを40mm程切断し、よりを戻しながら配列順に整列します。

通常はノーマル（ストレート）を使用します。

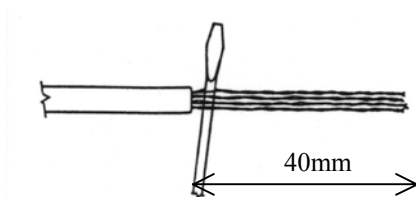


図 9-70 UTPケーブルのシース剥ぎ

配列

	T568B	T568A
	ノーマル	クロス
8	茶	茶
7	白/茶	白/茶
6	緑	橙
5	白/青	白/青
4	青	青
3	白/緑	白/橙
2	橙	緑
1	白/橙	白/緑

② 10BASE-T (UTP) ケーブルの信号線の切断

シース部から信号線を14mm程を残しニッパなどで切断します。

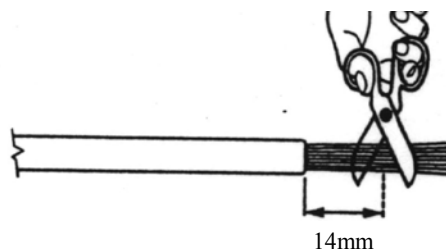


図 9-71 UTPケーブルの信号線の切断

③ UTPケーブルのコネクタへの挿入

配列順をくずさないようにコネクタに装着し、ケーブルが先端まで届いているかを正面および上下より確認します。

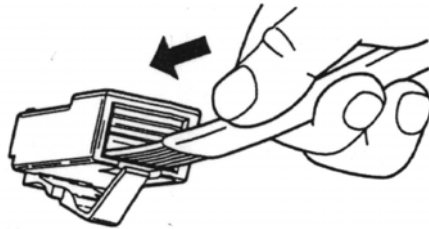


図9-72 UTPケーブルのコネクタへの挿入

④ UTPケーブルコネクタの組み立て

挿入状態を確認後、専用工具にて圧接します。圧接終了後、必ず専用テスターにて接続状態をテストしてください。

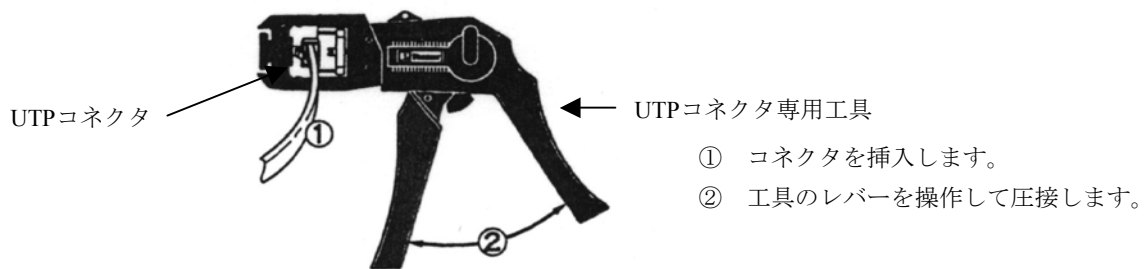


図9-73 UTPケーブルコネクタの組み立て

⑤ 適用ケーブル

S10mini FL.NETモジュールの推奨ケーブルは、「9.5.1 イーサネットの構成部品一覧」を参照してください。

(2) ハブ

(a) 適用ハブ

S10mini FL.NETモジュールの推奨ケーブルは、「9.5.1 イーサネットの構成部品一覧」を参照してください。

9.7 FL-netシステムの接地

9.7.1 FL-netシステムの接地の概要

FL-netシステムのコントローラ制御盤接地方法について、制御盤を建屋の鉄骨に接地する場合の例を図9-74と図9-75に示します。

制御盤を建屋の鉄骨に接地する場合の条件を以下に示します。この条件を満たさない場合には、コントローラ専用の接地（D種接地以上）をしてください。

- ・鉄骨どうしが溶接されていること
- ・大地と鉄骨の間は、D種接地工事基準を満足していること
- ・制御盤の接地点に強電回路の電流が流れ込まないこと
- ・制御盤の接地点と強電盤の接地点は、15m以上離すこと

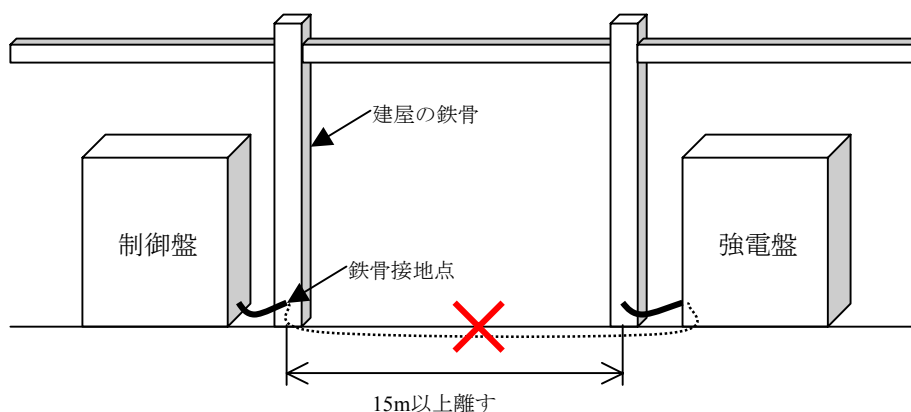


図9-74 コントローラ制御盤の接地方法例1（鉄骨接地の場合）

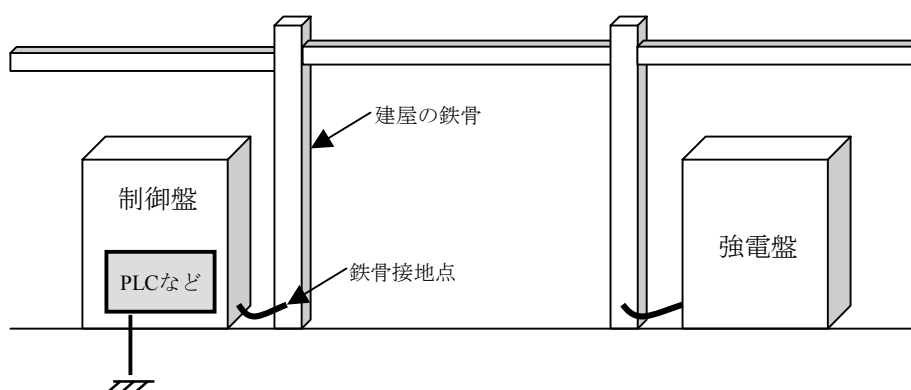


図9-75 コントローラ制御盤の接地方法例2（コントローラ単独D種接地の場合）

9. 7. 2 電源配線と接地

FL-netシステムの電源配線と接地について、分電盤およびコントローラ盤の電源配線およびアース接地例を図9-76に示します。

電源配線およびアース接地する場合は、下記に従ってください。

- ・制御電源とコントローラ電源間は、静電シールド付き絶縁トランスを使用して絶縁してください。
- ・分電盤およびコントローラ制御盤は、そのフレームをD種接地してください。
- ・コントローラのFG（フレームグラウンド）端子は、制御盤のフレームに接続しないで、コントローラ専用の接地（D種接地以上）をしてください。
- ・コントローラの入力電源配線は、できるだけ最短距離とし、ツイスト（より）配線してください。
- ・コントローラのLG（ライングラウンド）端子は、絶縁トランスのシールド端子に接続し、盤のフレームアースに接続してください。

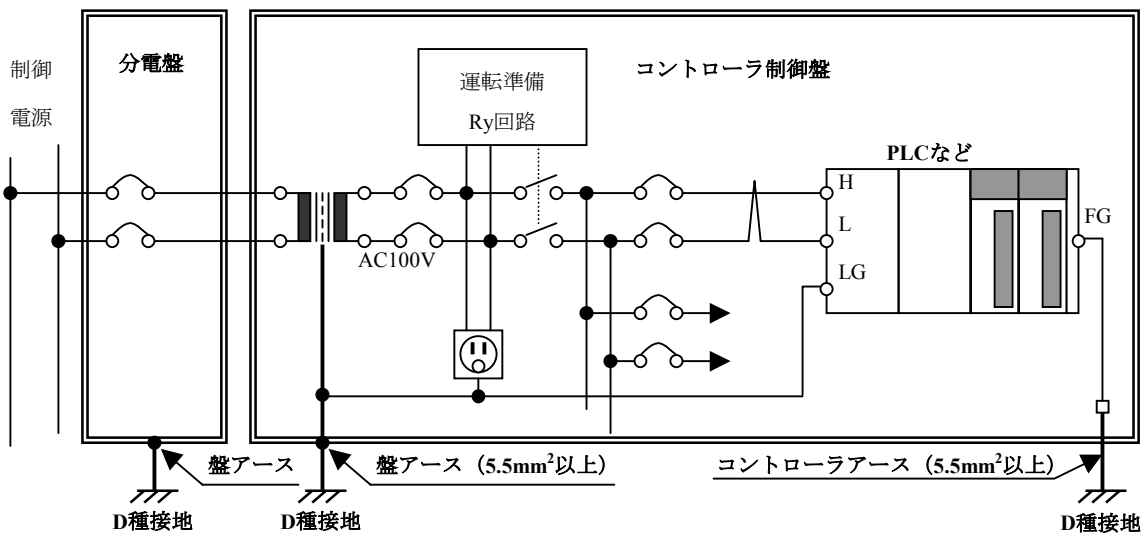


図9-76 FL-netシステムの電源配線と接地の例

9. 7. 3 FL-netシステムのネットワーク機器の電源配線とアース接地

FL-netシステムのネットワーク機器の電源配線とアース接地について、電源配線およびアース接地例を図9-77に示します

電源配線およびアース接地する場合は、下記に従ってください。

- ・同軸ケーブルのアース端子は、コントローラ専用D種接地に接続してください。
- ・10BASE-T用のハブ（HUB）は、そのフレームアースをコントローラ専用D種接地に接続してください。また、その電源は、コントローラの電源と同じ静電シールド付き絶縁トランスから給電してください。
- ・コントローラのFG（フレームグラウンド）端子は、制御盤のフレームに接続しないで、コントローラ専用の接地（D種接地以上）をしてください。
- ・FL.NETモジュールのFG（フレームグラウンド）端子は、コントローラのFG（フレームグラウンド）端子に接続してください。
- ・トランシーバ（AUI）ケーブルのシールドアースは、FL.NETモジュールのFG（フレームグラウンド）端子に接続してください。
- ・トランシーバ（AUI）に直流電源（DC12Vなど）の給電が必要な場合には、ネットワーク専用の安定化電源ユニットを設け、その直流出力をFL.NETモジュールの所定の端子に接続してください。また、そのAC100V入力電源は、コントローラと同様に静電シールド付き絶縁トランスから給電してください。

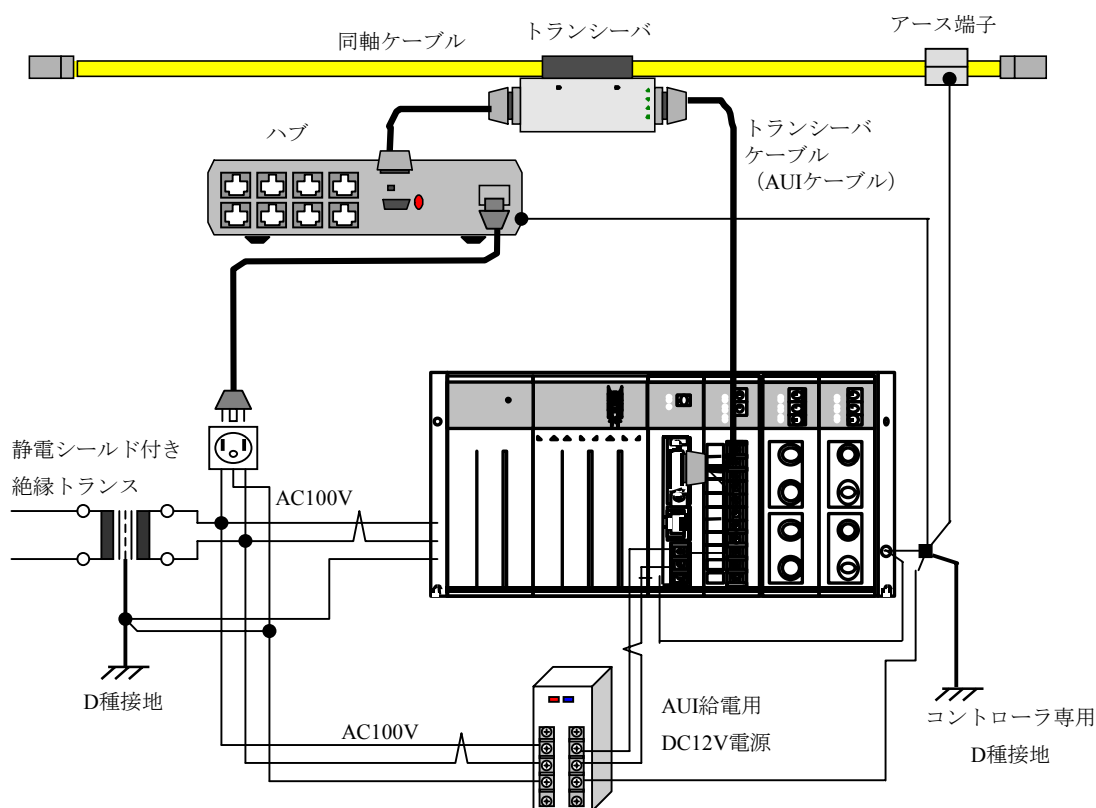


図9-77 FL-netシステムのネットワーク機器の電源配線とアース接地例

9. 7. 4 配線ダクト・電線管の配線と接地

配線ダクト・電線管の配線とアース接地について、図9-78と図9-79に示します。配線施工に関しては、下記に従ってください。

- ・配線ダクトを使用して配線する場合には、セパレータを使用して動力ケーブルと信号ケーブルをそのレベルに合わせて分離してください。また、その配線ダクト（フタおよびセパレータを含む）は、D種接地してください。
- ・電線管を使用して配線する場合には、動力ケーブルと信号ケーブルをそのレベルに合わせて個々に電線管を準備してください。また、その電線管は、JIS C 8305で定めるものを使用し、D種接地してください。

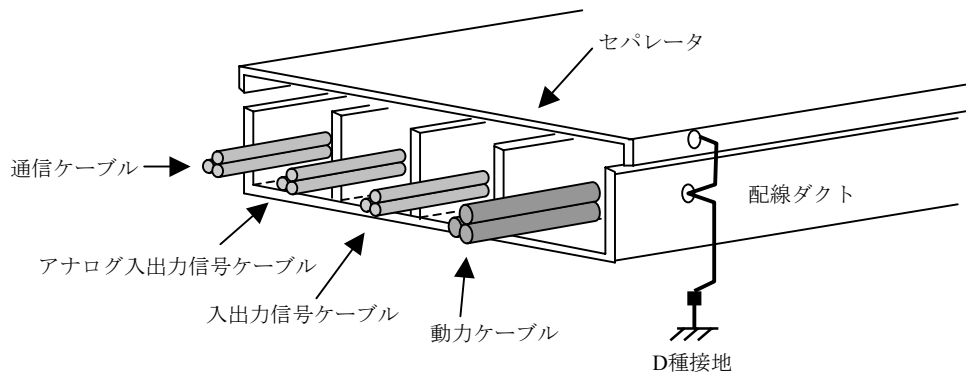


図9-78 配線ダクトを使用した場合の配線例

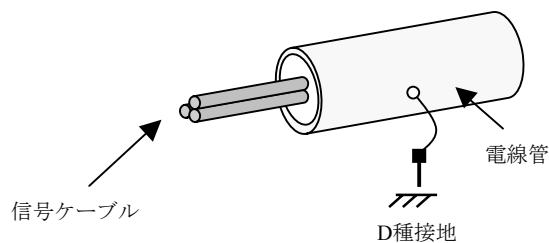


図9-79 電線管を使用した場合の配線例

9. 8 FL-net工事施工チェックシート

表 9-15 FL-net工事施工チェックシート

FL-net工事施工チェックシート			
通信ライン名： _____ 局番： _____ 点検日付： _____			
点検者 会社名： _____			
氏 名： _____			
チェック項目	記入欄	設定スイッチ欄	
ケーブル	コネクタは全部確実にロックされているか。		
	ケーブルの曲げ半径は既定値以上となっているか。		
	コネクタはジャケット等で保護されているか。		
	配線識別番号（線番）は貼り付けられているか、また間違いはないか。		()
	通信ケーブルが重量物の下敷になっていないか。		
	通信ケーブルが動力ケーブル等と束線されていないか。		
	リピータ用AUIケーブルの長さは、2m以内か。トランシーバ用は、50m以内か。		
	同軸ケーブル（10BASE-5）の長さは500m以内か。		
	同軸ケーブルは、アース端子で正しく接地されているか。		
	同軸ケーブルのシールドとトランシーバは、絶縁されているか。		
	同軸ケーブルに正しく終端抵抗が取り付けられているか。		
	ハブやリピータの段数は規定以内か。		
ユニット	ツイストペアケーブルは、ストレートケーブルを使用しているか。		
	ツイストペアケーブルは、カテゴリ5のものを用品、その長さは100m以内か。		
	機器のGND端子は正しく接地されているか。		
	各ユニットは確実にベースに締め付けられているか。		
ハブ等	ベースユニットは確実に制御盤に固定されているか。		
	AUIケーブルは確実にロックされているか。		
	AUIケーブル取り付け部に扉などにより無理な力がかからないか。		
	RJ45コネクタはきちんと装着されているか。		
	AUIケーブルのコネクタはロックされているか。		
	線番は貼り付けられているか。		()
	トランシーバはマーク位置に正しく設置されているか。		
トランシーバのSQEスイッチは、機器の仕様どおりに正しく設定されているか。		()	
ハブはきちんと固定されているか。			
ハブのハブ、MAU切り替えスイッチの設定に間違いはないか。		()	
ハブに供給される電源電圧は、規定値どおりか。			

・改造、変更、点検の際は必ずチェックし、記入すること
 ・記入欄には、OKは“O”、NGは“×”と記入し、設定スイッチ欄の（ ）内にはロータリスイッチの番号と、DIPスイッチのON/OFFを記入すること

9.9 FL-netのプロファイル

9.9.1 機器通信情報の分類

FL-netでは、ネットワークに接続される機器の通信に関わる情報を図9-80の3種類に分類しています。

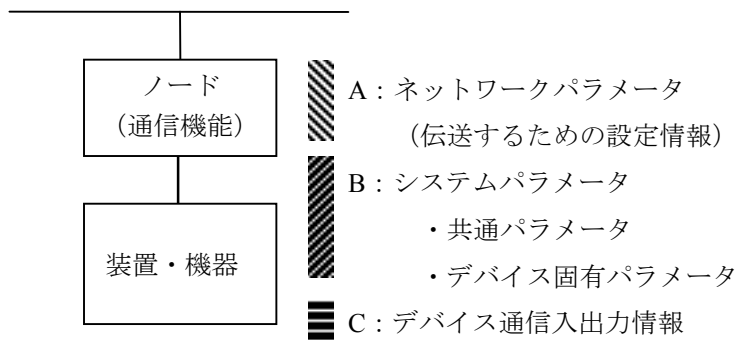


図9-80 機器通信情報の分類

- ネットワークパラメータ (A) は、伝送に必要な設定情報です。
- システムパラメータ (B) は、ネットワーク上にどのような機器 (デバイス) が接続されているかを識別するための管理情報で静的なパラメータであり、共通パラメータとデバイス固有パラメータに分けられています。
- デバイス通信入出力情報 (C) は、ネットワーク上の他の機器からアプリケーションの必要に応じてアクセスでき、アプリケーション動作や機器状態によってその内容が変化する動的な情報を含みます。

9. 9. 2 共通パラメータの詳細

共通パラメータの詳細を表9-16に示します。

表9-16 共通パラメータの詳細

パラメータ名称	名称文字 [PrintableString型] (長さ), (文字)	データタイプ [型]	パラメータ内容 (長さ), (内容)
デバイスプロファイル 共通仕様バージョン	6, "COMVER"	INTEGER	例: 1, 1
システムパラメータ識 別文字	2, "ID"	PrintableString	7, "SYSPARA"
システムパラメータ改 変番号	3, "REV"	INTEGER	例: 1, 0
システムパラメータ変 更日付	7, "REVDATE"	[INTEGER], 2, (0001- 9999), [INTEGER], 1, (01-12), [INTEGER], 1, (01-31)	例: 2, 1998 1, 9 1, 30
デバイス種別	10, "DVCATEGORY"	PrintableString	例: 3, "PLC" (注)
ベンダ名	6, "VENDOR"	PrintableString	例: 4, "MSTC"
製品型名	7, "DVMODEL"	PrintableString	例: 3, "JOP"

(注) デバイス種別のパラメータ内容を以下に示します。

“PC” または “PLC” …プログラマブルコントローラ

“NC” または “CNC” …数値制御装置

“RC” または “ROBOT” …ロボットコントローラ

“COMPUTER” …パーソナルコンピュータ、パネルコンピュータ、ワークステーション、
表示器などのコンピュータ

“SP-*…*” …仕様をベンダが指定 (*…*は半角英数字)

“OTHER” …その他

転送構文では、システムパラメータ全体、共通パラメータ全体、システムパラメータ変更日付、デバイス固有パラメータ全体 (オプション) でSEQUENCE構造型としてください (デバイス固有パラメータ内の構造化は任意)。

9. 9. 3 デバイス固有パラメータの詳細 (使用する場合)

表9-17 デバイス固有パラメータ詳細

パラメータ名称	名称文字	データタイプ	パラメータ内容
デバイス固有パラ メータ識別文字	2, "ID"	PrintableString	7, "DEV PARA"
内容はベンダがデバイスごとに自由に定義します。			

9 付 録

9. 9. 4 システムパラメータの例（PLCの例）

(1) システムパラメータの表形式文書表記でPLCの例

表 9-18 システムパラメータの表形式文書表記（PLCの例）

パラメータ名称	名称文字 [PrintableString型] (長さ), (文字)	データタイプ [型]	パラメータ内容 (長さ), (内容)
SysPara			
デバイスプロファイル共通仕様バージョン	6, "COMVER"	INTEGER	1, 1
システムパラメータ識別文字	2, "ID"	PrintableString	7, "SYSPARA"
システムパラメータ改変番号	3, "REV"	INTEGER	1, 0
システムパラメータ変更日付	7, "REVDATE"	[INTEGER] , 2, (0001-9999) , [INTEGER] , 1, (01-12) , [INTEGER] , 1, (01-31)	2, 1998 1, 9 1, 30
デバイス種別	10, "DVCATEGORY"	PrintableString	3, "PLC"
ベンダ名	6, "VENDOR"	PrintableString	29, "MSTC-JOP Electric Corporation"
製品型名	7, "DVMODEL"	PrintableString	5, "PLC-M"

パラメータ名称	名称文字	データタイプ	パラメータ内容
PlcmPara			
デバイス固有パラメータ識別文字	2, "ID"	PrintableString	7, "DEVPARA"
CPU1名称	8, "CPU1NAME"	PrintableString	9, "PMSP35-5N"
CPU2名称	8, "CPU2NAME"	PrintableString	9, "PMSP25-2N"
CPU3名称	8, "CPU3NAME"	PrintableString	9, "PMSP25-2N"
CPU4名称	8, "CPU4NAME"	PrintableString	9, "PMBP20-0N"
モジュール105名称	9, "IO105NAME"	PrintableString	9, "PMWD64-4N"
モジュール106名称	9, "IO106NAME"	PrintableString	9, "PMLD01-0N"
モジュール107名称	9, "IO107NAME"	PrintableString	9, "PMLE01-5N"

(2) 抽象構文

<型定義>

```

PlcmRecord ::=SEQUENCE
    {
        syspara          SysparaType,
        plcmpara         PlcmType
    }

SysparaType ::=SEQUENCE
    {
        nameCOMVER       NameType,
        paraCOMVER       INTEGER,
        nameID           NameType,
        paraID           NameType,
        nameREV          NameType,
        paraREV          INTEGER,
        nameREVDATE      NameType,
        paraREVDATE      DateType,
        nameDVCATEGORY   NameType,
        paraDVCATEGORY   NameType,
        nameVENDOR       NameType,
        paraVENDOR       NameType,
        nameDVMODEL      NameType,
        paraDVMODEL      NameType
    }

PlcmType ::=
SEQUENCE
    {
        nameID           NameType,
        paraID          NameType,
        module          SEQUENCE OF ModInfo
        DEFAULT {}
    }

NameType ::= PrintableString
DateType ::= SEQUENCE
    {
        year            INTEGER,
        month           INTEGER,
        day             INTEGER
    }

ModInfo ::= SEQUENCE
    {
        nameMODULE      NameType,
        paraMODULE      NameType
    }

```

<値定義>

```

{
    syspara {
        nameCOMVER          "COMVER",
        paraCOMVER          1,
        nameID              "ID",
        paraID              "SYSPARA",
        nameREV             "REV",
        paraREV             0,
        nameREVDATE         "REVDATE",
        paraREVDATE         { year 1998,
                           month 9,
                           day 30 },
        nameDVCATEGORY     "DVCATEGORY",
        paraDVCATEGORY     "PLC",
        nameVENDOR         "VENDOR",
        paraVENDOR         "MSTC-JOP Electric Corporation",
        nameDVMODEL        "DVMODEL",
        paraDVMODEL        "PLC-M"
    }
    plcpara {
        nameID              "ID",
        paraID              "DEVPARA",
        module
        {
            { nameMODULE "CPU1NAME",
              paraMODULE "PMSP35-5N" },
            { nameMODULE "CPU2NAME",
              paraMODULE "PMSP25-2N" },
            { nameMODULE "CPU3NAME",
              paraMODULE "PMSP25-2N" },
            { nameMODULE "CPU4NAME",
              paraMODULE "PMBP20-0N" },
            { nameMODULE "IO105NAME",
              paraMODULE "PMWD64-4N" },
            { nameMODULE "IO106NAME",
              paraMODULE "PMLD01-0N" },
            { nameMODULE "IO107NAME",
              paraMODULE "PMLE01-5N"}
        }
    }
}

```


(3) 転送構文データ配列 (符号化)

識別子	長さ	内容
\$30	\$820133	
\$30	\$7D	
\$13	\$06	“COMVER”
\$02	\$01	1
\$13	\$02	“ID”
\$13	\$07	“SYSPARA”
\$13	\$03	“REV”
\$02	\$01	0
\$13	\$07	“REVDATE”
\$30	\$0A	
\$02	\$02	\$07CE
\$02	\$01	\$09
\$02	\$01	\$1E
\$13	\$0A	“DVCATEGORY”
\$13	\$03	“PLC”
\$13	\$06	“VENDOR”
\$13	\$1D	“MSTC-JOP Electric Corporation”
\$13	\$07	“DVMODEL”
\$13	\$05	“PLC-M”
\$30	\$81B1	
\$13	\$02	“ID”
\$13	\$07	“DEVPARA”
\$30	\$15	
\$13	\$08	“CPU1NAME”
\$13	\$09	“PMSP35-5N”
\$30	\$15	
\$13	\$08	“CPU2NAME”
\$13	\$09	“PMSP25-2N”
\$30	\$15	
\$13	\$08	“CPU3NAME”
\$13	\$09	“PMSP25-2N”
\$30	\$15	
\$13	\$08	“CPU4NAME”
\$13	\$09	“PMBP20-0N”
\$30	\$16	
\$13	\$09	“IO105NAME”
\$13	\$09	“PMWD64-4N”
\$30	\$16	
\$13	\$09	“IO106NAME”
\$13	\$09	“PMLD01-0N”
\$30	\$16	
\$13	\$09	“IO107NAME”
\$13	\$09	“PMLE01-5N”

(4) 回線上のデータ並び

以下に回線上に送信されるデータの順序を示します。

下表の相対アドレス00欄で、最初のアドレス (0) からデータがバイトごとに横矢印方向の順序で送信されます。相対アドレス00欄の次は、相対アドレス10欄になり、以下同様に相対アドレスの順序で送信されます。

相対アドレス	データ (16進表示)															
	(0)	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(A)	(B)	(C)	(D)	(E)	(F)
00	30	82	01	33	30	7D	13	06	"C"	"O"	"M"	"V"	"E"	"R"	02	01
10	01	13	02	"I"	"D"	13	07	"S"	"Y"	"S"	"P"	"A"	"R"	"A"	13	03
20	"R"	"E"	"V"	02	01	00	13	07	"R"	"E"	"V"	"D"	"A"	"T"	"E"	30
30	0A	02	02	07	CE	02	01	09	02	01	1E	13	0A	"D"	"V"	"C"
40	"A"	"T"	"E"	"G"	"O"	"R"	"Y"	13	03	"P"	"L"	"C"	13	06	"V"	"E"
50	"N"	"D"	"O"	"R"	13	1D	"M"	"S"	"T"	"C"	"_"	"J"	"O"	"P"	"_"	"E"
60	"I"	"e"	"c"	"t"	"i"	"c"	"_"	"C"	"o"	"i"	"p"	"o"	"i"	"a"	"t"	
70	"I"	"o"	"n"	13	07	"D"	"V"	"M"	"O"	"D"	"E"	"L"	13	05	"P"	"L"
80	"C"	"_"	"M"	30	81	B1	13	02	"I"	"D"	13	07	"D"	"E"	"V"	"P"
90	"A"	"R"	"A"	30	15	13	08	"C"	"P"	"U"	"1"	"N"	"A"	"M"	"E"	13
A0	09	"P"	"M"	"S"	"P"	"3"	"5"	"_"	"5"	"N"	30	15	13	08	"C"	"P"
B0	"U"	"2"	"N"	"A"	"M"	"E"	13	09	"P"	"M"	"S"	"P"	"2"	"5"	"_"	"2"
C0	"C"	"N"	30	15	13	08	"P"	"U"	"3"	"N"	"A"	"M"	"E"	13	09	"P"
D0	"M"	"S"	"P"	"2"	"5"	"_"	"2"	"N"	30	15	13	08	"C"	"P"	"U"	"4"
E0	"N"	"A"	"M"	"E"	13	09	"P"	"M"	"B"	"P"	"2"	"0"	"_"	"0"	"N"	30
F0	16	13	09	"I"	"O"	"1"	"0"	"5"	"N"	"A"	"M"	"E"	13	09	"P"	"M"
100	"W"	"D"	"6"	"4"	"_"	"4"	"N"	30	16	13	09	"I"	"O"	"1"	"0"	"6"
110	"_"	"0"	"N"	30	16	13	09	"I"	"O"	"1"	"0"	"7"	"N"	"A"	"M"	"E"
120	13	09	"N"	"A"	"M"	"E"	13	09	"P"	"M"	"L"	"D"	"0"	"1"	"P"	"M"
130	"L"	"E"	"0"	"1"	"_"	"5"	"N"									

9. 9. 5 システムパラメータの例 (CNCの例)

表9-19にCNCのシステムパラメータ例を示します。

表9-19 システムパラメータの表形式文書表記 (CNCの例)

パラメータ名称	名称文字 [PrintableString型] (長さ), (文字)	データタイプ [型]	パラメータ内容 (長さ), (内容)
SysPara			
デバイスプロファイル共通仕様バージョン	6, "COMVER"	INTEGER	1, 1
システムパラメータ識別文字	2, "ID"	PrintableString	7, "SYSPARA"
システムパラメータ変更番号	3, "REV"	INTEGER	1, 0
システムパラメータ変更日付	7, "REVDATE"	[INTEGER], 2, (0001-9999), [INTEGER], 1, (01-12), [INTEGER], 1, (01-31)	2, 1998 1, 9 1, 30
デバイス種別	10, "DVCATEGORY"	PrintableString	3, "CNC"
ベンダ名	6, "VENDOR"	PrintableString	9, "MSTCJ LTD"
製品型名	7, "DVMODEL"	PrintableString	16, "MSTCJ Series 16a"

パラメータ名称	名称文字	データタイプ	パラメータ内容
CncPara			
デバイス固有パラメータ識別文字	2, "ID"	PrintableString	7, "DEVPARA"
モデル名	5, "MODEL"	PrintableString	8, "MS16a-MA"
系列	6, "SERIES"	PrintableString	4, "MSF1"
レビジョン	3, "REV"	INTEGER	1, 0
システム	7, "System"	SEQUENCE	*
システム情報	7, "SysInfo"	SEQUENCE	*

* このパラメータはConstructed typeで以下のデータを持ちます。

System			
オプション構成フラグ	5, "SFLAG"	BIT STRING	8, "00100101"
制御軸数	4, "AXES"	INTEGER	2, 4

SysInfo			
入力仮想アドレス	2, "IN"	OCTET STRING	6, "000000"
出力仮想アドレス	3, "OUT"	OCTET STRING	6, "040000"

(1) 抽象構文

<型定義>

```

CncRecord ::= SEQUENCE
{
  SysPara                               SysParaType,
  CncPara                               CncParaType,
}

SysParaType ::= SEQUENCE
{
  nameCOMVER                           NameType,
  paraCOMVER                            INTEGER,
  nameID                                 NameType,
  paraID                                 NameType,
  nameREV                                NameType,
  paraREV                                INTEGER,
  nameREVDATE                           NameType,
  paraREVDATE                           DateType,
  nameDVCATEGORY                         NameType,
  paraDVCATEGORY                         NameType,
  nameVENDOR                             NameType,
  paraVENDOR                             NameType,
  nameDVMODEL                            NameType,
  paraDVMODEL                            NameType
}

CncParaType ::= SEQUENCE
{
  nameID                                 NameType,
  paraID                                 NameType,
  nameMODEL                              NameType,
  paraMODEL                              NameType,
  nameSERIES                             NameType,
  paraSERIES                             NameType,
  nameREV                                NameType,
  paraREV                                INTEGER,
  nameSystem                             NameType,
  paraSystem                             SystemType,
  nameSysInfo                            NameType,
  paraSysInfo                            SysInfoType
}

SystemType ::= SEQUENCE
{
  nameINPUT                              NameType,
  paraINPUT                              BIT STRING,
  nameAXES                               NameType,
  paraAXES                               INTEGER
}

SysInfoType ::= SEQUENCE
{
  nameIN                                 NameType,
  paraIN                                 OCTET STRING,
  nameOUT                                NameType,
  paraOUT                                OCTET STRING
}

NameType := PrintableString
DateType := SEQUENCE
{
  year                                  INTEGER,
  month                                INTEGER,
  day                                  INTEGER
}

```

< 値定義 >

```

{
  SysPara
  {
    nameCOMVER      "COMVER",
    paraCOMVER      1,
    nameID           "ID",
    paraID           "SYSPARA",
    nameREV          "REV",
    paraREV          0,
    nameREVDATE     "REVDATE",
    paraREVDATE     { year      1998,
                    month     9,
                    day       30      },
    nameDVCATEGORY  "DVCATEGORY",
    paraDVCATEGORY  "CNC",
    nameVENDOR       "VENDOR",
    paraVENDOR       "MSTCJ LD",
    nameDVMODEL      "DVMODEL",
    paraDVMODEL      "MSTCJ Series 16a"
  }

  CncPara
  {
    nameID           "ID",
    paraID           "DEVPARA",
    nameMODEL        "MODEL",
    paraMODEL        "MS16a-MA",
    nameSERIES        "SERIES",
    paraSERIES        "MSF1",
    nameREV          "REV",
    paraREV          0,
    nameSystem       "System",
    paraSystem       { nameINPUT    "SFLAG",
                    paraINPUT    '00100101'B,
                    nameAXES     "AXES",
                    paraAXES     4      },
    nameSysInfo      "SysInfo",
    paraSysInfo      { nameIN      "IN",
                    paraIN      '000000000000'H
                    nameOUT     "OUT",
                    paraOUT     '000400000000'H  }
  }
}

```

<このページは余白です>

ご利用者各位

〒101-8010

東京都千代田区神田駿河台4丁目6番地
株式会社日立製作所

お 願 い

各位にはますますご清栄のことと存じます。

さて、この資料をより良くするために、お気付きの点はどんなことでも結構ですので、
下欄にご記入の上、弊社営業担当または弊社所員に、お渡しくださいますようお願い申
しあげます。なお、製品開発、サービス、その他についてもご意見を併記して頂ければ
幸甚に存じます。

ご住所 〒	_____
貴会社名 (団体名)	_____
芳 名	_____
製品名	_____
ご意見欄	_____ _____ _____ _____ _____ _____ _____ _____ _____ _____ _____ _____ _____ _____ _____