

ソフトウェアマニュアル
オペレーション

HIDIC
S10 シリーズ

NX/ACP-S10 For Windows®

2α
シリーズ

対象機種

HIDIC-S10/2

HIDIC-S10/2 E

HIDIC-S10/2 H

HIDIC-S10/2 Hf

本製品を輸出される場合には、外国為替および外国貿易法の規則
をご確認のうえ、必要な手続きをお取りください。
なお、ご不明な場合は、弊社担当営業にお問い合わせください。

1999年 3月 (第1版) SAJ - 3 - 134 (A)

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写することは、
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。

All Rights Reserved, Copyright © 1999, Hitachi, Ltd.



安全上のご注意

システムの構築やプログラムの作成などの操作を行う前には、このマニュアルの記載内容をよく読み、書かれている指示や注意を十分理解してください。

誤った操作により、システムの故障が発生することがあります。

このマニュアルは、必要なときすぐに参照できるよう、手近な所に保管してください。

このマニュアルの記載内容について理解できない内容、疑問点または不明点がございましたら、最寄りの当社営業もしくはSEまでお知らせください。

お客様の誤った操作に起因する事故発生や損害につきましては、当社は責任を負いかねますのでご了承ください。

当社提供ソフトウェアを改変して使用した場合には、発生した事故や損害につきましては、当社は責任を負いかねますのでご了承ください。

当社提供以外のソフトウェアを使用した場合の信頼性については、当社は責任を負いかねますのでご了承ください。

ファイルのバックアップ作業を日常業務に組み入れてください。ファイル装置の障害、ファイルアクセス中の停電、誤操作、その他何らかの原因によりファイルの内容を消失することがあります。このような事態に備え、計画的にファイルをバックアップしてください。

はじめに

NX/ACP-S10システムをお買い上げいただきまことにありがとうございます。

このマニュアルは、HIDIC S10/2 シリーズ（以下S10と略します）で協調自律分散システムを構築するための通信プラットフォームであるNXACPについて、その機能とマクロコールのリンケージ仕様を中心に、プログラムをMicrosoft® Windows® 95搭載のパーソナルコンピュータ（以下パソコンもしくはPCと略します）上で作成する方法について説明したものです。

なお、このマニュアルは、このシステムのリアルタイムOSであるCPMS(E)、Windows®やWSで使用されるソケットの概念、パソコンとWindows® 95およびMS-DOS®についての基礎知識を持っている読者を対象に記述されています。

< マニュアル構成 >

1 概 説

NXACPの特徴、システム構成、基本的な用語説明、および機能概要について解説しています。

2 機能ガイド

NXACPのシステム構築、プログラム開発、テスト、運用で必要となる基本機能について次の6節に分類し解説しています。

- 2.1 マルチキャスト通信機能
- 2.2 データフィールド管理機能
- 2.3 テスト機能
- 2.4 システム管理機能
- 2.5 運用管理機能
- 2.6 システム構築機能

3 マクロ仕様

NXACPが提供するマクロコールの機能とリンケージ仕様について解説しています。

< 関連マニュアル >

ソフトウェアマニュアル オペレーション RPDP/S10 For Windows® (SAJ-3-133)

ソフトウェアマニュアル 概説&マクロ仕様 コンパクトPMS V5 (SAJ-3-201)

ハードウェアマニュアル オプション ET.NET (LWE550) (SAJ-2-124)

NeXUS/Dlink ガイド&リファレンス (RS90-3-X019)

- ・ マニュアル番号のXは改訂番号を示します。

Microsoft®、Windows® 95、MS-DOS® は米国Microsoft Corporationの米国およびその他の国における登録商標または商標です。

目 次

1	概 説	1
1.1	概 要	2
1.1.1	NXACPとは	2
1.1.2	ハードウェア構成	3
1.1.3	ソフトウェア構成	4
1.2	用語の説明	5
1.2.1	基本用語の説明	5
1.3	機能概要	11
1.3.1	マルチキャスト通信機能	11
1.3.2	タスク起動機能	14
1.3.3	常駐バッファ機能	14
1.3.4	データフィールド管理機能	15
1.3.5	テスト機能	16
1.3.6	システム管理機能	17
1.3.7	運用管理機能	18
1.3.8	システム構築機能	19
2	機能ガイド	21
2.1	マルチキャスト通信機能	22
2.1.1	通信の特長	22
(1)	メッセージ通信の単位	22
(2)	メッセージの分割・組立	23
(3)	ユーザプログラム構造	24
(4)	メッセージ処理順序	26
2.1.2	メッセージ送信機能	28
(1)	ネットワーク送信	28
(2)	自ノード内送信	35
2.1.3	メッセージ受信機能	36
(1)	メッセージ受信	36
(2)	マルチキャストグループの定義	39
2.1.4	リモートデータフィールド制御機能	41
(1)	システム接続形態	41
(2)	メッセージ送受信	43
2.1.5	バッファ管理	45

2.2	データフィールド管理機能	46
2.2.1	生存信号送信機能	46
2.3	テスト機能	48
2.3.1	メッセージ入出力制御	48
2.3.2	テスト形態	50
2.3.3	リモートデータフィールドとモード	52
2.4	システム管理機能	54
2.4.1	障害通知機能	54
2.4.2	ユーザタスク管理機能	55
2.4.3	メッセージトレース機能	56
2.5	運用管理機能	57
2.5.1	NXACPとデータフィールドの運用管理	57
2.6	システム構築機能	58
2.6.1	NXACPのインストール	59
2.6.2	システム構築	60
2.6.3	NXACP本体のローディング	61
2.6.4	構築情報とローディング	63
	(1) 構築情報の設定	63
	(2) 構築情報のコンパイル	71
	(3) 構築情報のローディング	72
2.6.5	容量見積り	73
3	マクロ仕様	75
3.1	総説	76
3.1.1	マクロの種類と一覧	76
3.2	マルチキャスト通信マクロ	77
3.2.1	putran	77
3.2.2	getran	79
3.2.3	getmsg	81
3.3	運用マクロ	83
3.3.1	acpinit	83
3.3.2	acpquit	84
3.4	タイマ起動パラメータ変更マクロ	85
3.4.1	sdtimer	85
3.4.2	stimer	86
3.4.3	swake	87

付 録	89
付録A リターンコード詳細	90
付録B 組み込みサブルーチンの登録	97
付録C 異常NXヘッダ通知フォーマット	99
付録D メッセージヘッダフォーマット	101
付録E 生存信号ヘッダフォーマット	102

目 次

図 1 - 1	ネットワーク最小構成	3
図 1 - 2	ネットワーク最大構成	3
図 1 - 3	ソフトウェア構造	4
図 1 - 4	NXACP要素説明	5
図 1 - 5	データフィールド	5
図 1 - 6	ローカルデータフィールドとリモートデータフィールド	6
図 1 - 7	メッセージ	7
図 1 - 8	トランザクションコード	8
図 1 - 9	論理ノード	9
図 1 - 10	マルチキャストグループ	10
図 1 - 11	ポート	10
図 1 - 12	マルチキャスト通信	11
図 1 - 13	マルチキャストグループ	12
図 1 - 14	メッセージの分割・再生	12
図 1 - 15	メッセージのフィルタリング	13
図 1 - 16	常駐バッファ	14
図 1 - 17	生存信号	15
図 1 - 18	状態監視ノード	15
図 1 - 19	テスト機能	16
図 1 - 20	UPタスク管理	17
図 1 - 21	NXACP運用管理	18
図 1 - 22	システム構築	19
図 2 - 1	通信単位	22
図 2 - 2	putranマクロ	24
図 2 - 3	getranマクロ	25
図 2 - 4	処理順序	26
図 2 - 5	優先レベル使用時の制限事項	27
図 2 - 6	データフィールド番号と送信範囲	28
図 2 - 7	データフィールド番号とネットワークセグメント	29
図 2 - 8	マルチキャストと選択受信	29
図 2 - 9	リモートデータフィールド(1)	30
図 2 - 10	リモートデータフィールド(2)	31
図 2 - 11	リモートデータフィールド(3)	31
図 2 - 12	特定ノード宛送信(1)	33

図 2 - 13 特定ノード宛送信 (2)	34
図 2 - 14 自ノード内通信	35
図 2 - 15 getran()の特長	36
図 2 - 16 受信処理の流れ	37
図 2 - 17 共用受信	38
図 2 - 18 宛先送信マルチキャストグループと受信マルチキャストグループ	40
図 2 - 19 データフィールドタイプ	41
図 2 - 20 リモートデータフィールド (1)	42
図 2 - 21 リモートデータフィールド (2)	42
図 2 - 22 リモートデータフィールドのメッセージ送受信	44
図 2 - 23 生存信号	46
図 2 - 24 メッセージ入出力制御	49
図 2 - 25 テスト形態 (1)	50
図 2 - 26 テスト形態 (2)	50
図 2 - 27 テスト形態 (3)	51
図 2 - 28 リモートデータフィールドとモード (1)	52
図 2 - 29 リモートデータフィールドとモード (2)	53
図 2 - 30 リモートデータフィールドとモード (3)	53
図 2 - 31 メッセージ到着時のタスク状態チェック	55
図 2 - 32 ABORT 後処理	56
図 2 - 33 NXACP の運用	57
図 2 - 34 各マクロと運用	57
図 2 - 35 インストールと構築手順	58
図 2 - 36 リソース配置	59
図 2 - 37 送信周期	66

表 目 次

表 1 - 1	カテゴリ A 通信機能一覧	2
表 1 - 2	通信仕様一覧	2
表 2 - 1	マルチキャストグループとポート	29
表 2 - 2	トランザクションコードとユーザタスク対応	32
表 2 - 3	マルチキャストグループとノード対応	33
表 2 - 4	トランザクションコードとノード対応	34
表 2 - 5	トランザクションコードとユーザタスク対応	37
表 2 - 6	マルチキャストグループ番号と受信ポート番号	39
表 2 - 7	マルチキャストグループ番号と送受信ポート番号	39
表 2 - 8	データフィールドタイプとバッファ定義	45
表 2 - 9	生存信号タイプ	47
表 2 - 10	生存信号付加情報	47
表 2 - 11	ノードモードとメッセージモード	48
表 2 - 12	モードとポート番号	49
表 2 - 13	リソース割付け	62
表 2 - 14	構築ファイル一覧	63
表 2 - 15	起動タイプとパラメータ	69
表 2 - 16	容量算出シート	73
表 3 - 1	マクロ一覧	76

1 概 説

1. 1 概 要

1. 1. 1 NXACPとは

NXACPとは、プログラマブルコントローラ対応自律分散オンラインパッケージの総称で、NX/ACP-S10は、HIDIC S10/2 シリーズを対象とした通信プラットフォームです。また、ワークステーション（以降WS）やパソコン（以降PC）上の自律分散オンラインパッケージであるNX Dlinkとともに協調して動作し、自律分散ネットワーク環境を提供することを目的としたソフトウェアです。

NX/ACP-S10は、「オープン自律分散インタフェース技術説明書 V1.1」に示す『カテゴリA』の下記機能をサポートしています。

表 1 - 1 カテゴリA通信機能一覧

（ : サポート、× : 未サポート）

第一次機能			第二次機能		
クラス名	機能名		クラス名	機能名	
A-Base-1	マルチキャスト通信機能		A-Opt-1-a	二重化LAN制御マルチキャスト通信機能	×
A-Base-2	生存信号送出機能		A-Opt-2-a	障害情報送出機能	
A-Opt-3	1対1通信機能（最大16K ¹ ノット）	×			
A-Opt-4	ネットワーク経路構築機能	×			

【補足】協調自律分散プロトコルにサポートされている優先制御は未サポートです。

NX/ACP-S10では、優先レベル（PRI）を0固定にてメッセージ送信します。

また、メッセージ受信時も優先レベルに従った優先処理はせず、到着順に処理します。

NX/ACP-S10の主な通信仕様を以下に示します。

表 1 - 2 通信仕様一覧

項 目	仕 様
ネットワーク	Ethernet
通信プロトコル	UDP / IP
メッセージサイズ	最大16384バイト
ローカルデータフィールド登録数	最大3 DF（自ノード内通信含）
リモートデータフィールド登録数	最大2 DF / 1ローカルデータフィールド
データフィールド登録総数	最大5 DF
送信マルチキャストグループ登録数	最大16グループ
受信マルチキャストグループ登録数	最大6グループ（1 DFにつき）
同一トランザクションコード入力タスク数	最大16タスク / 1 TCD（1 DFにつき）

【補足】各情報の指定有効範囲については「2. 6 システム構築機能」を参照してください。

1. 1. 2 ハードウェア構成

このシステムの機能を利用するために必要なハードウェア構成を図1 - 1に示します。

最小構成

パソコンとS10メインモジュールで接続した構成が最小構成となります。

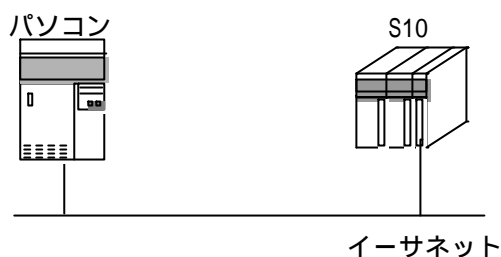


図1 - 1 ネットワーク最小構成

最大構成

2本のネットワークを接続して使用した場合（メインおよびサブモジュール）が最大構成となります。

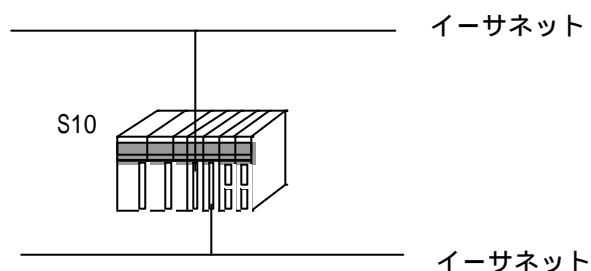


図1 - 2 ネットワーク最大構成

ルーティング機能を使用して、ルータやゲートウェイを経由したネットワークとも送受信できます。ただし、そのネットワークセグメント数は、1つの直接接続されたネットワークセグメントに対し最大2セグメントで、1つのS10から送受信できるネットワークセグメントは、直接接続されたネットワークとルータなどを経由したネットワークをすべて合わせて4セグメントまでです。

なお、ルータやゲートウェイを経由したネットワークシステムを構築する場合の注意、制限事項については、「ハードウェアマニュアル オプション ET.NET(LWE550)」（マニュアル番号 SAJ-2-124）を参照してください。

1. 1. 3 ソフトウェア構成

このシステムの位置付け、ソフトウェア構成を図1 - 3に示します。また、前提とするソフトウェアは以下のとおりです。

- ・HIDIC S10/2 CPMS(E) (リアルタイムモニタシステム)
- ・HIDIC S10/2 ET.NET (イーサネットドライバ)
- ・HIDIC S10/2 RPDP (リアルタイムシステム開発パッケージ)

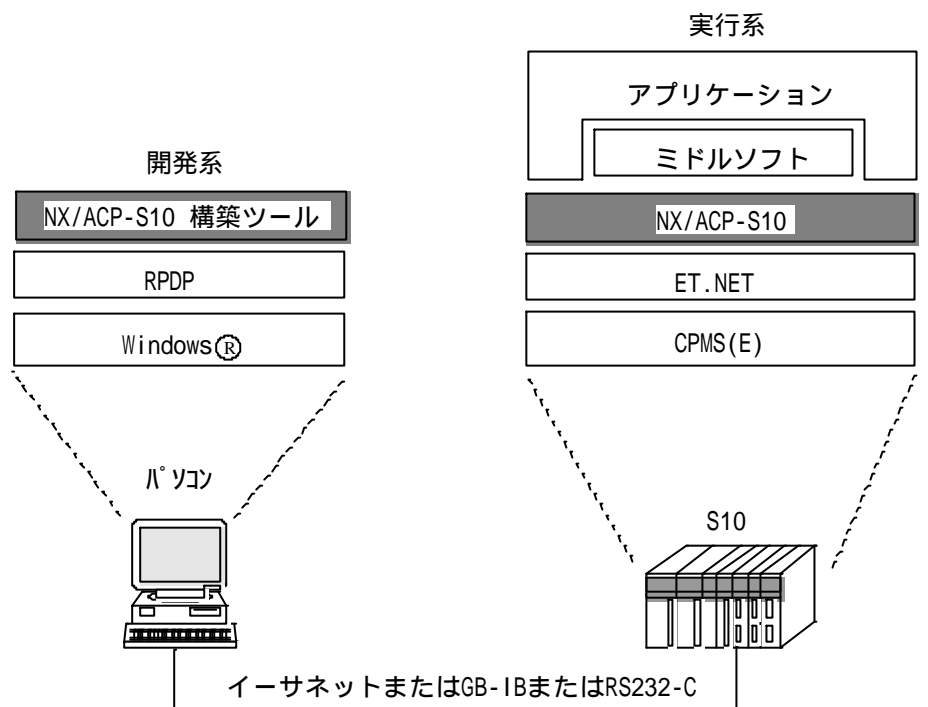


図1 - 3 ソフトウェア構造

バージョンを以下に示します。

対象PCs	システムF/D名称およびバージョン	
S10/2	CPMS ロードシステム	0 3 - 0 0 以降(S-7890-04)
	CPMS デバッガシステム	0 3 - 0 0 以降(S-7890-06)
	ET.NET システム	0 3 - 0 0 以降(S-7890-29)
	NX/ACP-S10 システム	0 1 - 0 0 以降(S-7891-11)
	RPDP/S10 システム	0 3 - 0 0 以降(S-7891-10)
S10/2 E	CPMSEロードシステム	0 3 - 0 0 以降(S-7890-05)
S10/2 H	CPMSEデバッガシステム	0 3 - 0 0 以降(S-7890-07)
S10/2 Hf	ET.NET システム	0 3 - 0 0 以降(S-7890-29)
	NX/ACP-S10 システム	0 1 - 0 0 以降(S-7891-11)
	RPDP/S10 システム	0 3 - 0 0 以降(S-7891-10)

1. 2 用語の説明

1. 2. 1 基本用語の説明

NXACPは、図1 - 4に示す要素群で構成します。

NXACPでは、従来のネットワークパッケージとは異なり、ネットワーク構成に依存しない抽象的な用語を使用します。

この項では、図1 - 4の各構成要素について説明します。図1 - 4に示す()内の数字は、説明番号に対応しています。

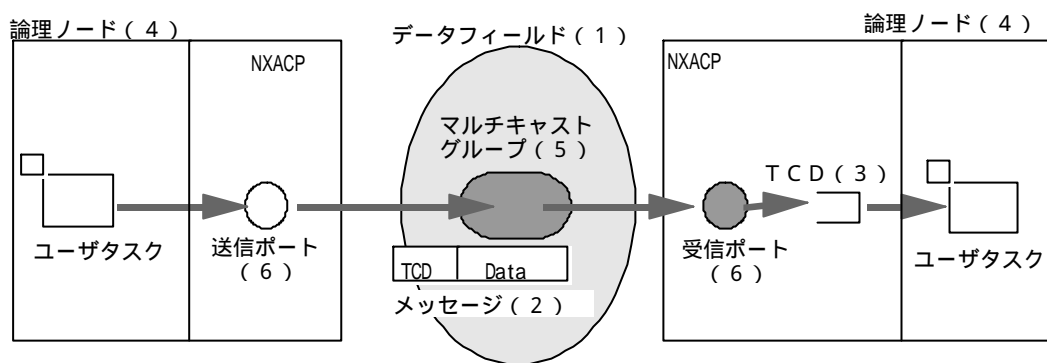


図1 - 4 NXACP要素説明

(1) データフィールド

特定の属性情報によって構成されるメッセージの流れる場をデータフィールド (DF) と呼びます。NXACPでは、データフィールドは1つのLANセグメントに対応し、複数の機器が属します。つまり、データフィールドは、送受信するメッセージの伝送範囲を決めるために使用し、LNSPでのループ番号に対応しています。

分散システム内の各データフィールドを、一意に識別するための番号をデータフィールド番号 (DFN) と呼びます。

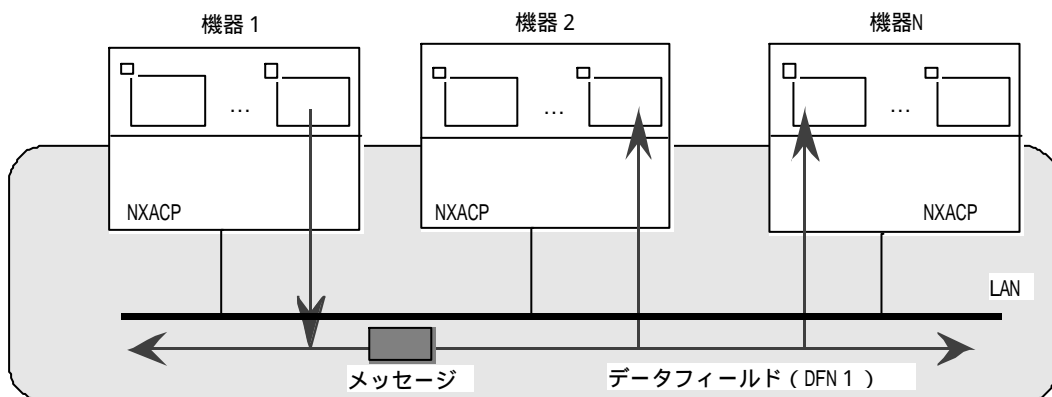


図1 - 5 データフィールド

データフィールドには、ローカルデータフィールドとリモートデータフィールドがあります。

ローカルデータフィールド：自ノードが直接接続するデータフィールドです。

リモートデータフィールド：ルータ、ゲートウェイを介して接続され、自ノードが間接的に接続されるデータフィールドです。

(例) データフィールド1に接続しているノード1から見たローカルデータフィールドは、データフィールド1で、リモートデータフィールドはデータフィールド2と3です。

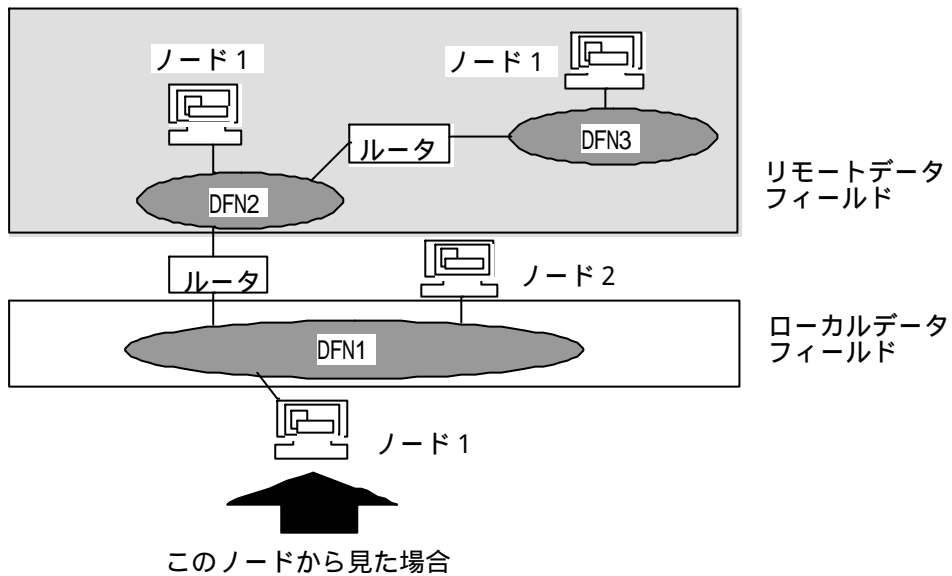


図1 - 6 ローカルデータフィールドとリモートデータフィールド

(2) メッセージ

NXACP間で交信するデータのかたまりの基本単位をメッセージと呼び、ユーザデータとNXヘッダにて構成されます。

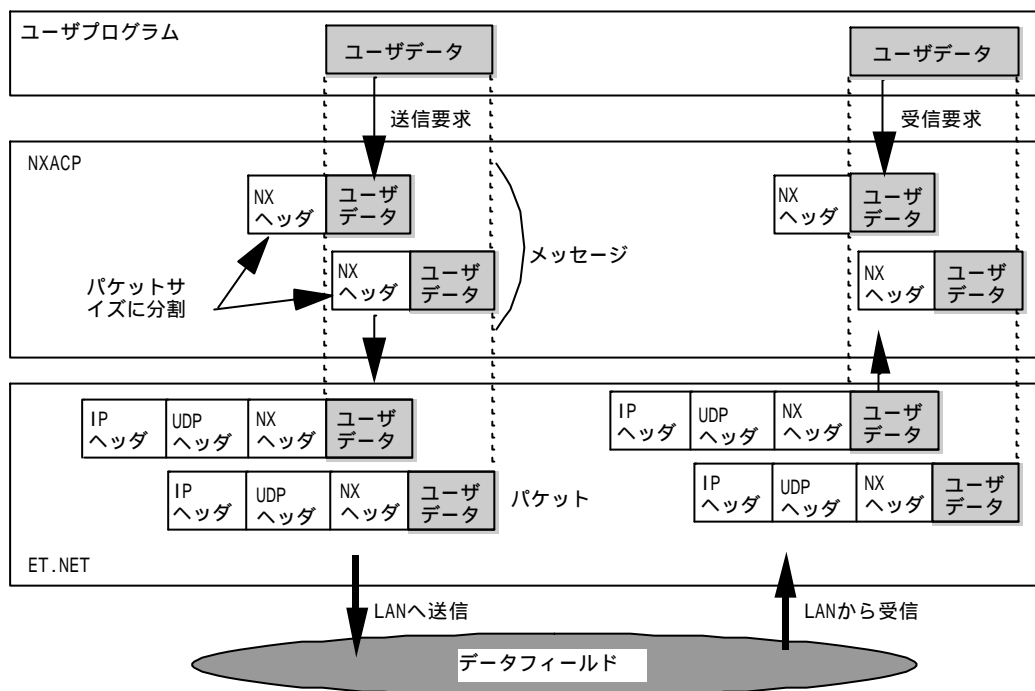


図 1 - 7 メッセージ

(3) トランザクションコード

トランザクションとは、商品やサービス、リアルタイムイベントなどに関する処理、取引を意味します。例えば、店で買い物をしたり、旅行会社で旅行に関するサービスを受けることは、取引であり、すべてトランザクションと言えます。トランザクション処理とは、そのような取引を実行、監視、記録することです。

つまり、アプリケーション業務がプラントデータ、証券取引データなどを受け入れると、データベースを更新する処理、他業務への通知処理などがすべてトランザクション処理ということになります。

このシステムでは、各データフィールドを流れるメッセージをトランザクションとして処理します。そのための識別子をトランザクションコード（TCD）と呼びます。

TCDとメッセージとの対応付けは、LNSPにてサポートされていた機能コード（FC）同様ユーザが任意に設定でき、ユーザプログラムは、TCDを指定してメッセージを送受信します。

ユーザプログラムが指定したTCDは、各メッセージのNXヘッダに格納されます。

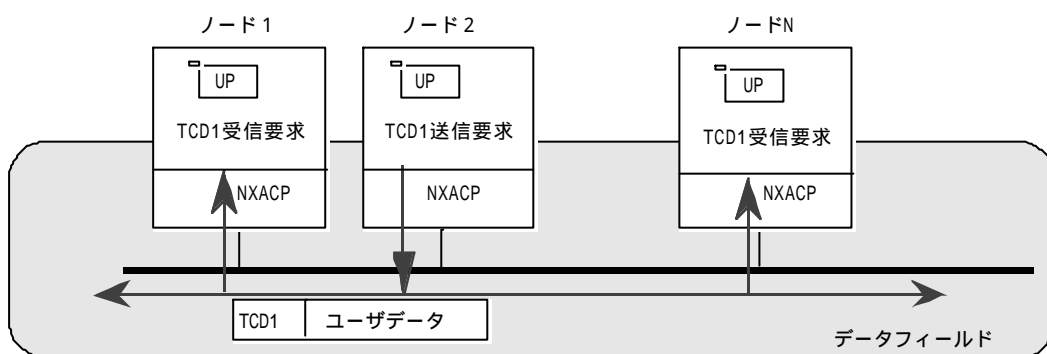


図 1 - 8 トランザクションコード

(4) 論理ノード

データフィールドに所属する機器を論理ノードと呼びます。データフィールド内で一意に論理ノードを識別するための番号が論理ノード番号（LNN）です。NXACPでは、CPUを論理ノードとして扱います。

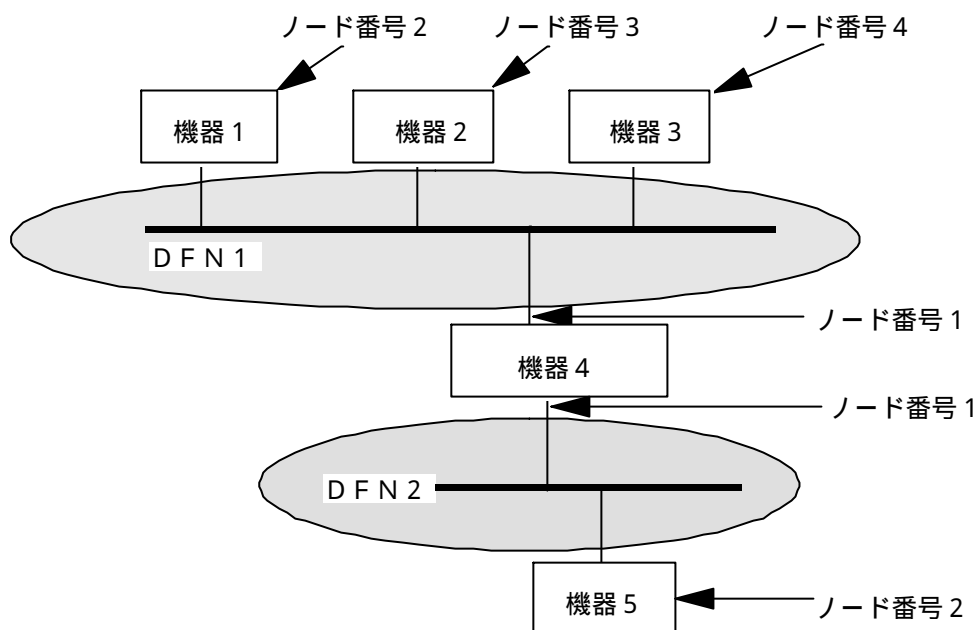


図 1 - 9 論理ノード

ノードごとにモードを所有し、オンライン稼働中のノードをオンラインノード、テスト中のノードをテストノードと呼びます。

(5) マルチキャストグループ

マルチキャストグループとは、ある論理ノードからデータフィールド内へブロードキャストしたデータを受信するかどうかをグループ化したものです。各データフィールドごとにマルチキャストグループを管理し、1つのデータフィールド内に複数のマルチキャストグループを構成でき、データフィールドに属する各論理ノードは、複数のマルチキャストグループに属することができます。

1つのマルチキャストグループは、1組のポート番号（オンライン/テストポート番号）に対応し、データフィールド内でマルチキャストグループを識別するための番号が、マルチキャストグループ番号（MGN）です。

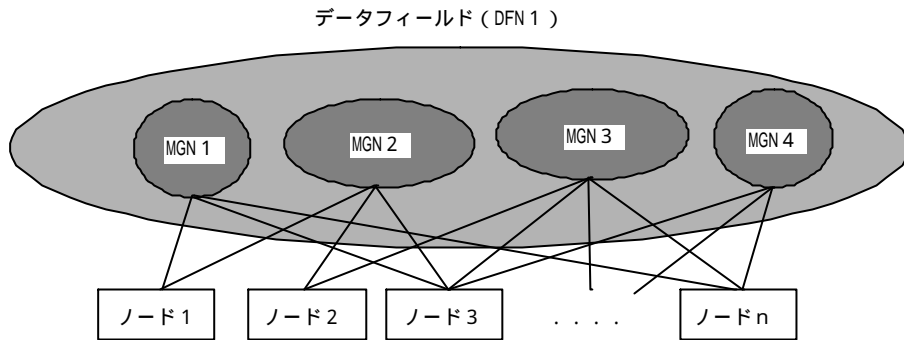


図 1 - 10 マルチキャストグループ

(6) ポート

物理的には単一の伝送路であるネットワークを使用してメッセージを送受信する場合、送受信元の属性を統一した方法で行うため、送信ポートという中継点と、宛先ポートという終点を設定します。マルチキャストグループに対応するポート番号とは、宛先ポート番号です。

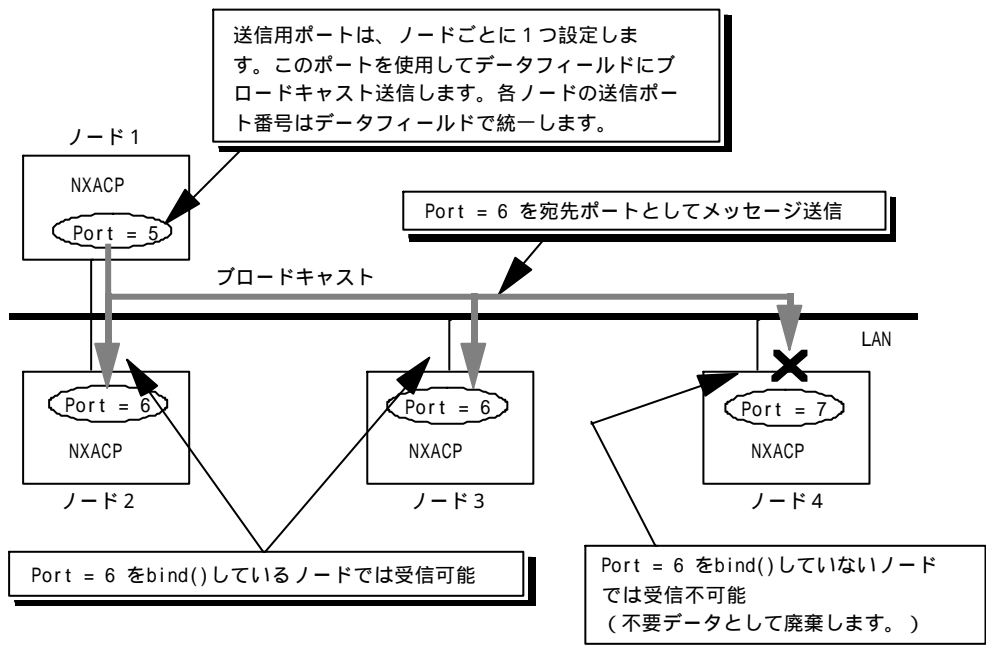


図 1 - 11 ポート

上記ポート番号 6 を、マルチキャストグループ 1 に対応させる場合、ノード 2、3 は、マルチキャストグループ 1 に属するノードということになります。マルチキャストグループ番号とポート番号の対応は、データフィールド内全ノードにて統一してください。

NXACPでは、送信マルチキャストグループとして宛先ポート番号を、受信マルチキャストグループとして受信ポート (bind()するポート) 番号を構築情報で指定します。

1. 3 機能概要

1. 3. 1 マルチキャスト通信機能

マルチキャスト通信は、NeXUSプロトコルのベースとなる通信方式で、以下の特長があります。

- ・複数ノードへ同時に同じメッセージを送信できるため、効率の良い送信ができる。
- ・メッセージの送受信においてノードの依存関係がなく、またコネクションの確立といった手順も不要なため、使い勝手が良い。

送信側では、送信先のデータフィールド、マルチキャストグループ、トランザクションコードなどを指定してメッセージを送信し、受信側の各ノードは自律的に必要なメッセージのみを取込みます。

したがって、“通信相手の認識”、“通信相手との同期などが不要”となり、各機器あるいは各ユーザプログラムの独立性を向上すると同時に、システムの拡張性を高めることができます。

マルチキャスト通信の送信範囲は、指定データフィールド番号に対応したネットワークセグメントにブロードキャストします。受信側ノードでは、マルチキャストグループを定義していない場合やトランザクションコードを定義していない場合は、不要メッセージとして廃棄します。

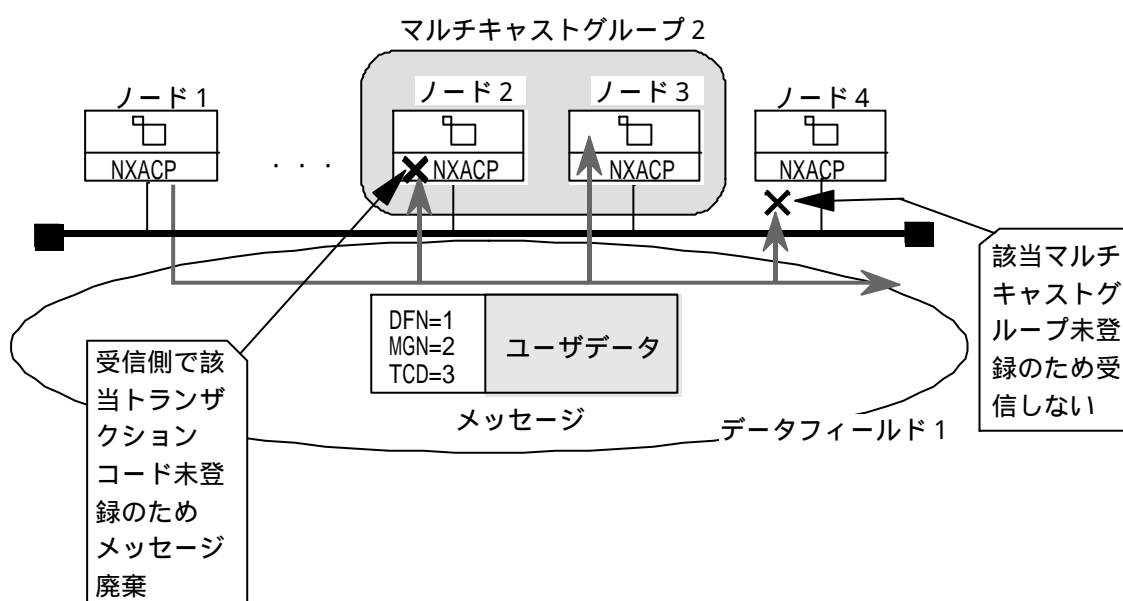


図 1 - 12 マルチキャスト通信

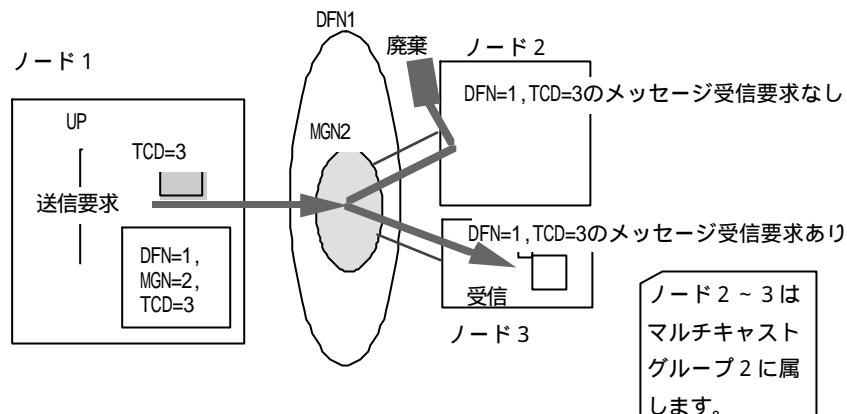


図 1 - 13 マルチキャストグループ

ユーザは、送信するTCDを指定してputran()、getran()マクロを発行することにより、メッセージ送受信を行えます。このとき、ユーザデータの扱いには、以下のような特長があります。

(1) 16Kバイトデータ送受信

UPでは、最大16Kバイトのデータを1回の送受信要求で指定できます。実際のネットワーク上を流れるパケットは、ユーザ指定サイズ(最大:MTUサイズ-64バイト)に分割して送信されますが、このときの分割・再生をNXACPIにて制御します。

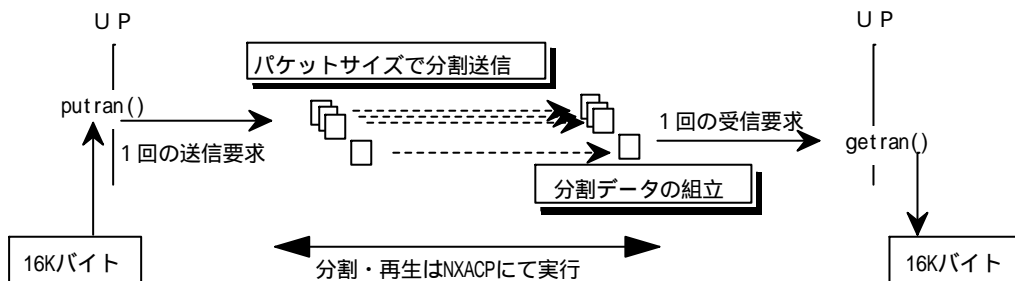


図 1 - 14 メッセージの分割・再生

(2) 送受信範囲

メッセージの送信範囲は、データフィールド番号により異なります。

DFN = 0を指定して送信要求した場合、自ノード内専用送信になります。

DFN = 0を指定して送信要求した場合、ネットワークおよび自ノード内への送信になります(自ノード内へ送信したデータを受信するには、受信TCDの定義が必要です)。

メッセージを受信する場合も同様に、自ノード内からのメッセージについては、DFN = 0を指定した場合に受信できます。ネットワークで接続された他ノードからメッセージを受信する場合は、DFN = 0を指定して受信します。

なお、データの送信時は、putran()マクロのパラメータにてデータフィールド番号を指定してください。データの受信時は、構築情報で指定してください。

【補足：フィルタリング機能について】

S10では、ET.NETにて不要メッセージの廃棄処理をします。これを、フィルタリング機能と呼びます。

フィルタリング機能とは、必要な受信メッセージの判別をハードウェアレベルで行い、不要なメッセージを廃棄する機能のことで、CPUの負荷低減を目的としています。

S10でのフィルタリング機能の対象は、ポート番号となります。したがって、NXACPを使用するユーザには、マルチキャストグループ番号がフィルタリング機能の対象となります。

NXACPでは、ユーザが構築情報にて定義したマルチキャストグループ番号をET.NETに登録します。したがって、登録していないマルチキャストグループ宛のメッセージについては、ET.NETレベルにて廃棄されます。

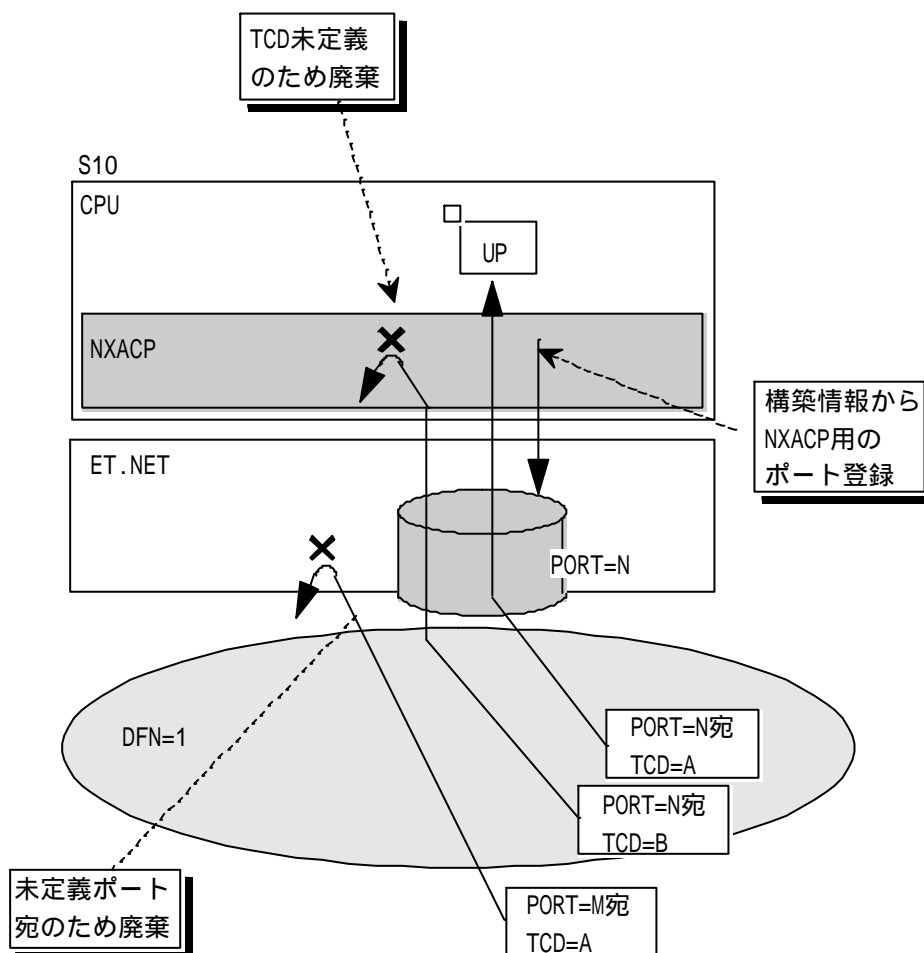


図 1 - 15 メッセージのフィルタリング

1 概 説

1. 3. 2 タスク起動機能

タスク起動機能とは、TCD受信時、該当TCDを受信するタスクを起動する機能です。

NXACPでは以下に示す6つの起動方法をサポートしています。ただし、(2)~(6)の起動時は、メッセージを該当タスクに渡しません。

(1) イベント起動

構築情報にて指定された該当TCDを受信するタスクをqueue()マクロを使用して起動します。

(2) 時間起動

構築情報にて指定された時間経過後、一度だけ起動します。

(3) 周期起動

構築情報にて指定された時間経過後、構築情報で指定された周期にて起動します。

(4) 時刻起動

構築情報にて指定された時刻に、一度だけ起動します。

なお、構築情報にて指定された時刻をすでに経過している場合には、起動されません。

(5) 時間、周期起動キャンセル

時間起動、および周期起動のタスクの動作をキャンセルします。

なお、タスクがすでに起動されて動作中の場合には、動作は中止されず以降の起動が行われなくなります。

(6) 時刻起動キャンセル

時刻起動のタスクの動作をキャンセルします。

なお、タスクがすでに起動されて動作中の場合には、動作は中止されません。

1. 3. 3 常駐バッファ機能

常駐バッファ機能とは、常に一番最後に受信したメッセージをNXACP内のバッファに保存しておく機能で、自ノード内通信(DFN=0)時のみ、使用できます。

常駐バッファ上のメッセージを取得する場合には、getmsg()マクロを使用します。

なお、常駐バッファは、TCD単位に設定できます。

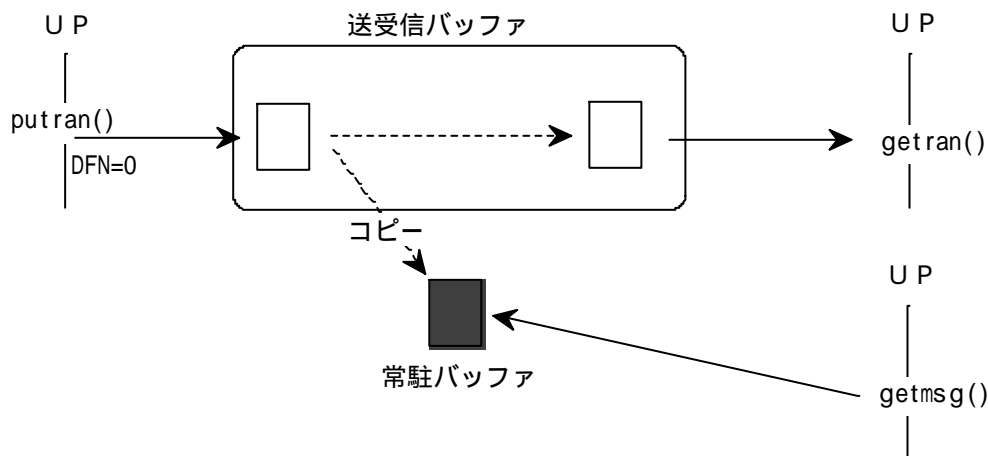


図1 - 16 常驻バッファ

1. 3. 4 データフィールド管理機能

データフィールド管理とは、データフィールドに接続しているノードの接続状態を監視または自ノードの状態を他ノードへ通知する機能です。

NXACPでは、データフィールド単位にそのデータフィールドに属する各ノードが、定期的に“生存信号”というメッセージを送信します。このメッセージにより、各ノードとデータフィールドの接続状態を他ノードから監視でき、システム全体の構成および制御ができるようになります。

生存信号の送出機能

NXACPでは、ユーザからの立上げ要求を受付けた時点から、生存信号メッセージの定周期送信をします。生存信号には、ノードモードやシステム立上げ時間、IPアドレス、生存信号モードなどの情報を付加して送信します。生存信号メッセージの送信周期は、構築情報にてユーザが指定します。

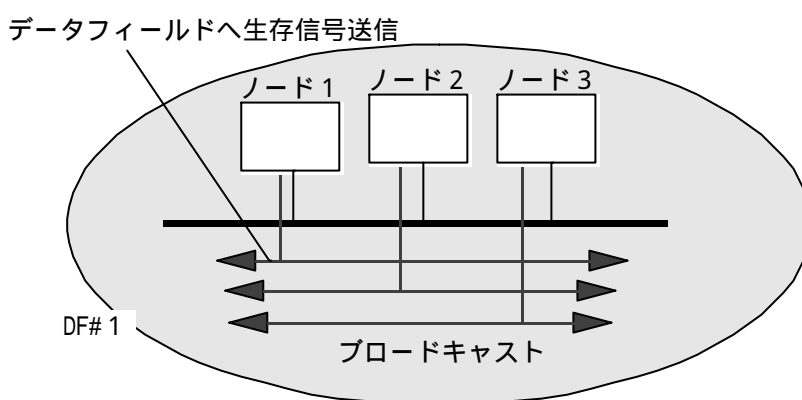


図 1 - 17 生存信号

ノード状態の監視機能

NXACPでは、他ノードからの生存信号によりノードの状態を管理する機能はありませんので、ネットワークシステム上にWSやPCを接続し、該当WSやPCにてノード状態を監視してください。

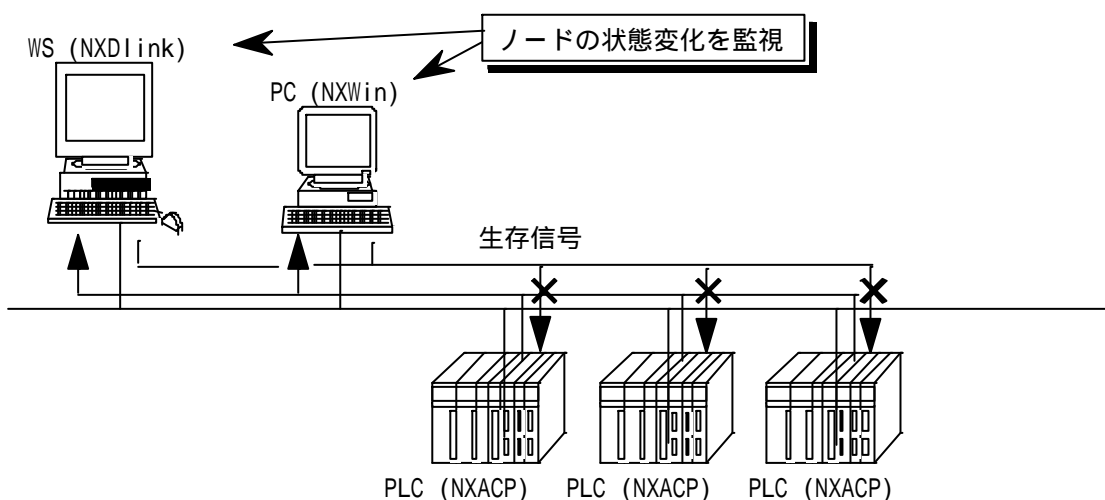


図 1 - 18 状態監視ノード

1. 3. 5 テスト機能

テスト機能とは、オンライン稼働中のシステム上で、論理ノードごとのテストを実現する機能です。

オンライン稼働中のノードをオンラインモードノード（以降、略してオンラインノード）、テスト中のノードをテストモードノード（以下テストノードと略します）と呼び、テスト対象ノードをテストモードに設定することで、オンライン系のノードに擾乱を与えないテストができるようになります。

これは、データフィールドを流れるメッセージにも、オンラインノードが送信するメッセージをオンラインモードメッセージ（以下オンラインメッセージと略します）、テストノードが送信するメッセージをテストモードメッセージ（以下テストメッセージと略します）というように、メッセージにもモードを付けることにより、以下のようなテスト形態を実現しています。

- ・テストノードから送信するメッセージは、テストメッセージとして送信され、オンラインノードでは、オンラインメッセージのみを受信します。したがって、オンライン環境でオンライン系のノードに擾乱を生じさせないテストが実施できます。
- ・オンラインメッセージをテストノードでは受信できると設定することで、オンラインデータを使用したテストが実施できるようになります。

NXACPでは、テストノードにて受信するメッセージモードを下記2つの中から選択でき、上記以外にも多様なテストシステム構成を構築できます。

- ・オンラインメッセージのみを受信するモード
- ・テストメッセージのみを受信するモード

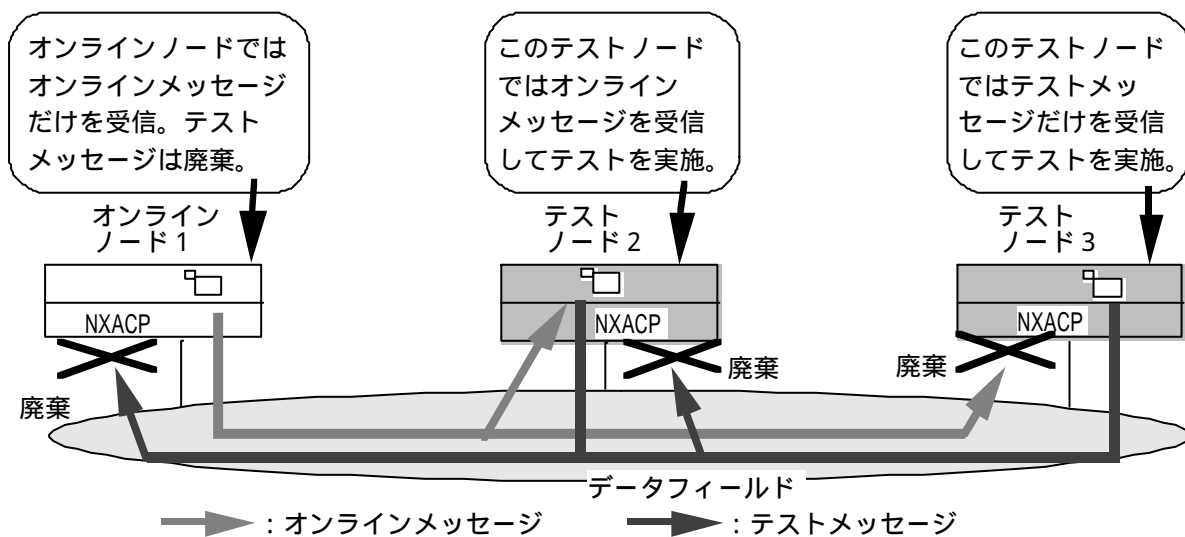


図 1 - 19 テスト機能

【補足】

オンライン系のノードへは、上記のとおりテストメッセージの入力を抑止するため、擾乱を与えませんが、ネットワークへは、テストメッセージが送出されますので、ネットワーク負荷を考慮して使用してください。

1. 3. 6 システム管理機能

このシステムには以下のシステム管理機能があります。

- ・ユーザタスク状態管理機能
- ・通信障害管理機能

ユーザタスク状態管理機能

メッセージを受信するユーザタスクの状態を監視します。これは、メッセージを受信できない場合は、バッファオーバーフローを防止するためのものです。

NXACPでは、ユーザタスクの状態をメッセージ到着時にチェックし、DORMANTまたはNON-EXISTの場合、該当タスクへメッセージを配信しません。また、ユーザタスクがDORMANT 遷移時でも、該当タスクが受信処理中および受信予定のメッセージを廃棄します。

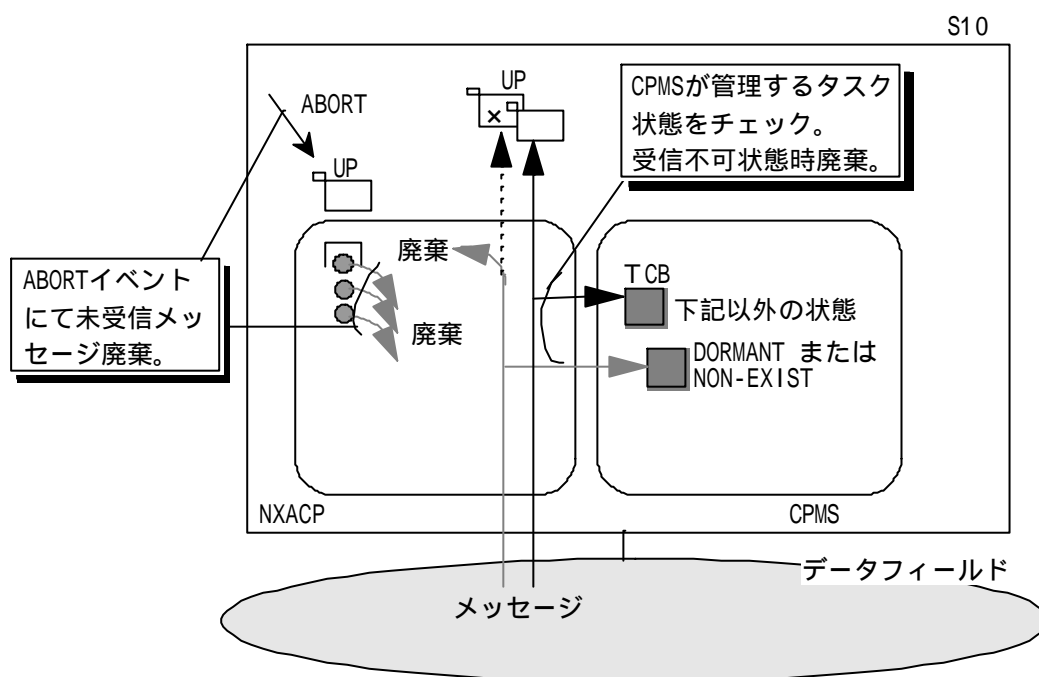


図 1 - 20 UPタスク管理

通信障害管理機能

NeXUSプロトコル障害発生時、NXACPはエラー情報をIRSUB番号38へ通知します。ユーザは、IRSUB番号38を組み込むことにより、障害情報をタイムリーにキャッチし、的確な対処を講じることができます。また、ソケット障害が発生した場合、該当障害はNXACPの制御トースバッファに逐次記録されます。

1. 3. 7 運用管理機能

NXACPのサービス開始/停止は、ユーザからの要求で制御できます。

NXACPの起動/停止は、`acpinit()` / `acpquit()`を発行することで行えます。

(1) NXACP起動

ユーザは、システム立ち上げ時に`acpinit()`を発行してください。`acpinit()`の発行により、データフィールドへのメッセージの送受信を開始できます。

(2) NXACP停止

システムを動作したままNXACPを停止したい場合は、`acpquit()`を発行してください。`acpquit()`を発行した場合、ローカルデータフィールド内の他ノードへ計画停止 (SHUTDOWNモードの生存信号) であることを通知してから送受信サービスを停止します。

`acpquit()`にてNXACPを停止させた後再起動する場合は、`acpinit()`を発行してください。

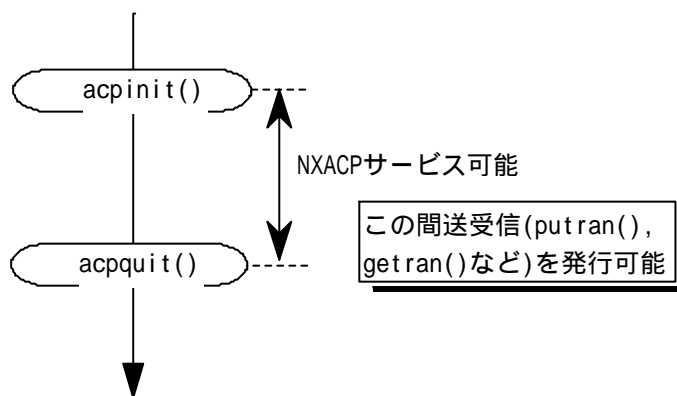


図 1 - 21 NXACP運用管理

1. 3. 8 システム構築機能

NXACPの制御情報構築は、パソコン（開発系）上にて構築情報を作成し、ターゲットとなるサイトのGLBエリアにRPDPコマンドを使用してローディングします。

ユーザは、構築情報をデータベースファイルに定義し、NXACP構築情報ローディング専用コマンドを起動してください。

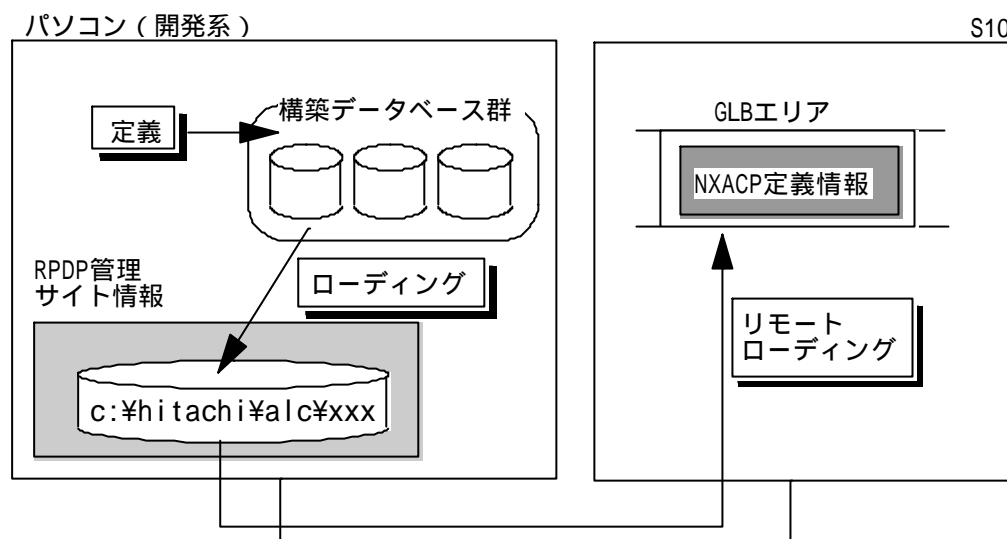


図 1 - 22 システム構築

このシステムには下記の構築情報があります。

- ・バッファ面数などのバッファ情報
- ・生存信号送出間隔、ポート番号などの生存信号情報
- ・ネットワークアドレス、ポート番号などのネットワーク情報
- ・TCD、MGN、送受信タスク番号などのトランザクション情報

これらの情報を、データベースファイルに定義し、NXACP構築情報ローディングコマンドを使用してコンパイルし、ターゲットとなるサイトにローディングします。NXACP構築情報ローディングコマンドは、RPDPコマンドを使用してGLBエリアにローディングしますので、システムジェネレーションが完了した後に実行してください。

なお、オンライン変更は未サポートです。構築情報をターゲットマシンに反映する場合、構築情報ローディングコマンドは、該当マシンのNXACPを一時停止させた状態にします。

この場合、SHUTDOWN予告の生存信号は送信しませんので注意してください。

2 機能ガイド

2. 1 マルチキャスト通信機能

2. 1. 1 通信の特長

このシステムで利用できる通信方式について説明します。

(1) メッセージ通信の単位

このマニュアルでは、ユーザが入出力する情報の固まりを「データ」と呼びます。

ユーザが指定したデータにNeXUSヘッダを付加し、NXACP内で1つの論理的な情報として扱うものを「メッセージ」と呼びます。また、ネットワーク上を流れる1つの情報単位を「パケット」と呼びます。

NXACPは、ユーザから指定されたデータにNeXUSヘッダを付加（メッセージ化）し、それをパケットに分割してネットワーク伝送します。受信側のNXACPでは、分割されたパケットをメッセージに組立て、NeXUSヘッダを削除してユーザに渡します。

ユーザが指定できる最大データ長は16Kバイトです。

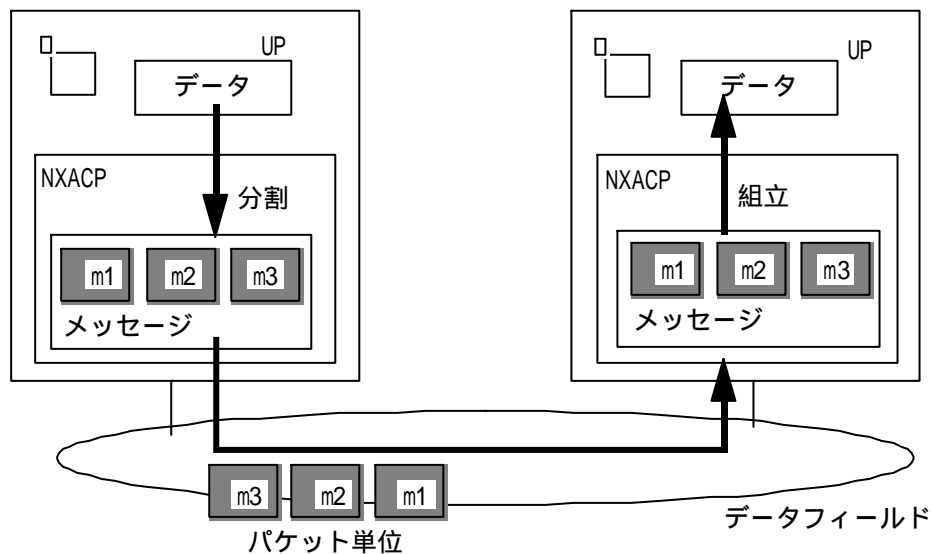
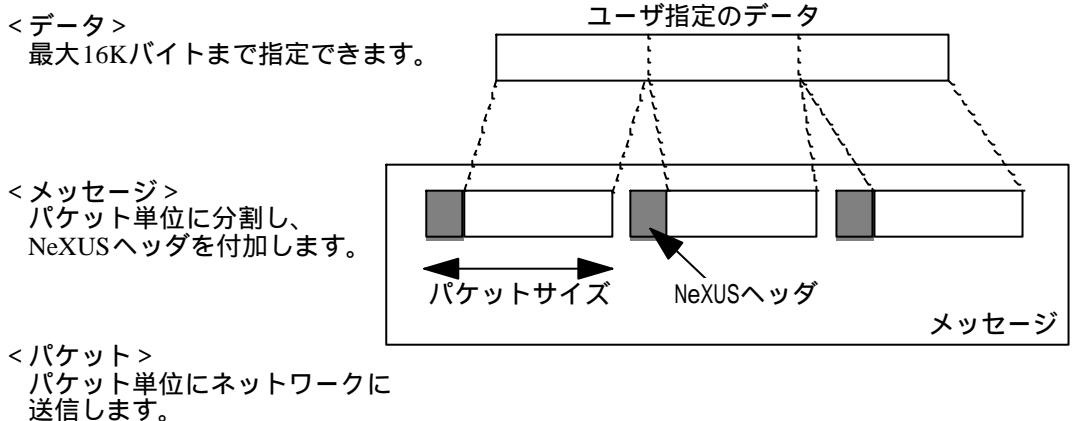


図 2 - 1 通信単位

(2) メッセージの分割・組立

ユーザが送信時に指定できる最大データ長は16Kバイトですが、NXACPでは、パケットサイズに分割して送信し、受信側で組立てます。

NXACPではバッファサイズは1408バイト、分割するパケットサイズはバッファサイズにNeXUSヘッダ(64バイト)を加えたサイズ=1472バイト固定となります。

16Kバイトの送信要求を発行した場合のパケット使用ケース数は以下ようになります。

$$\text{パケット使用ケース数} = \frac{(16384 - 1)}{1408} + 1 = 11 + 1 = 12 \text{ ケース}$$

【補足】

バッファ1ケースサイズはデータフィールドごとに全ノードで統一してください。

2 機能ガイド

(3) ユーザプログラム構造

ユーザは、データの送受信要求時、putran() / gettran() マクロを発行してください。

putran() / gettran() マクロは、以下に示す特長があります。なお、左記マクロは全タスクが発行できません。

(a) 送信要求

ユーザは、データ送信要求時、putran() マクロを発行してください。

NXACPでは、ユーザプログラムの送信処理がハードウェアの処理と非同期に実行できるよう、ユーザからの送信要求を受付けるサービスと、ネットワーク上にパケットを送信するサービスを分けています。また、ネットワーク上にパケットを送信するサービスは、周期起動（周期時間は構築情報）を採用し、スイッチング回数の低減を図っています。

したがって、putran()が正常終了しても、ネットワークに正常に送信されたことを保証するものではありません。ユーザプログラムからの送信要求（putran()）に対し、putran()マクロが発行元ユーザに返す情報は以下の情報に限定されます。なお、以下の障害検知時は、いずれも送信処理を中断します。

- ・動作環境チェックで異常を検知したときのエラーコード
- ・パラメータ異常を検知したときのエラーコード
- ・NXACPの送信バッファが満杯状態のときのエラーコード

ネットワーク上にパケットを送信するサービスを実行時にエラーを検出した場合は、制御トレースエリアに情報を記録します。

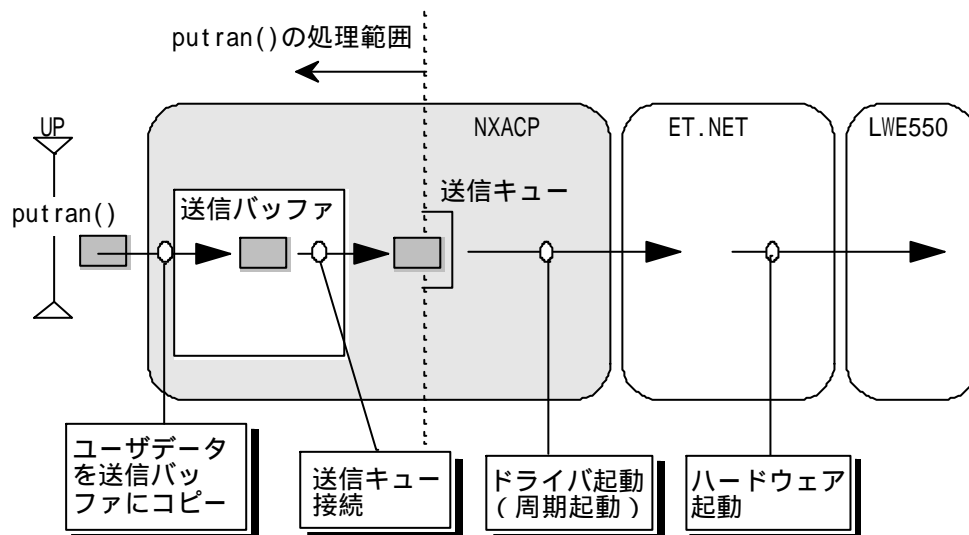


図 2 - 2 putran マクロ

(b) 受信要求

ユーザは、データ受信要求時、getran()マクロを発行してください。

受信処理も送信処理同様、ハードウェアからの割り込み処理とユーザプログラムの受信処理が非同期に実行できるよう、ユーザからの受信要求を受付けるサービスと、ネットワークからの割り込みサービスを分けて処理します。

したがって、受信要求時getran()発行元ユーザに障害情報として通知できるのは、以下の場合のみで、受信処理を中断します。

- ・動作環境チェックで異常を検知したときのエラーコード

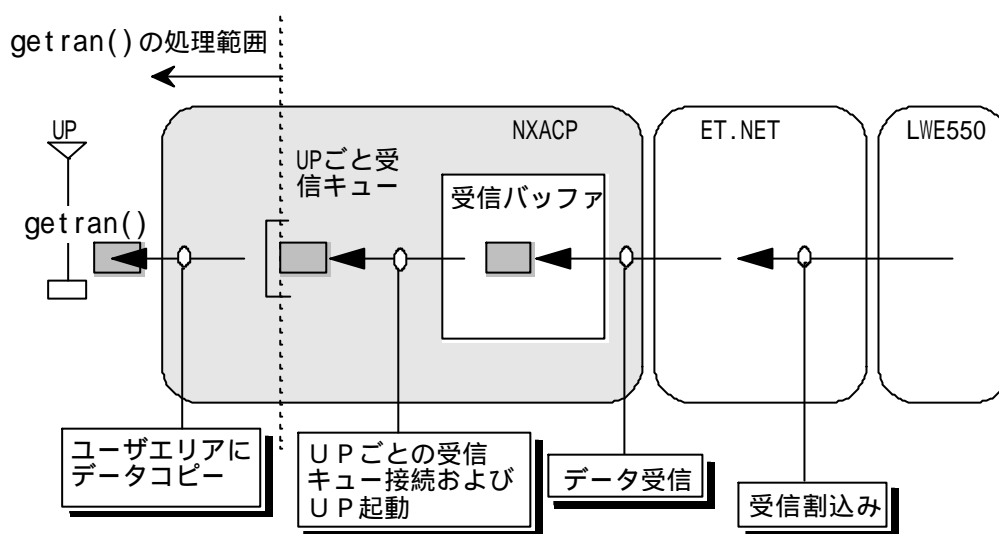


図 2 - 3 getranマクロ

getran()を発行するユーザタスクは走り切り型の構造を取ります。これは、データ受信時NXACPが該タスクをqueue()マクロにて起動するためです。

2 機能ガイド

(4) メッセージ処理順序

NXACPでのメッセージ送受信処理では、送受信ともメッセージ発生順番（FCFS）にてサービスします。したがって、メッセージ送信においては、同じタスクレベルでは、ユーザの送信要求順のサービスとなり、メッセージ受信においては、マルチキャストグループ単位での到着順のサービスとなります。

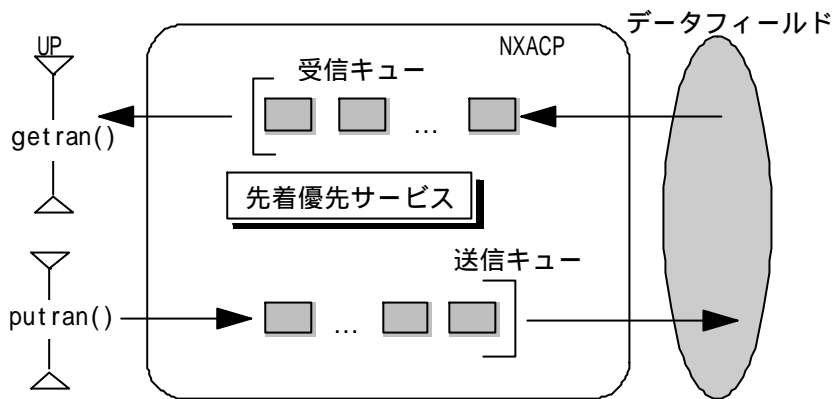


図 2 - 4 処理順序

【補足】

NXACPは、メッセージ優先制御は未サポートです。

NeXUSプロトコルには、「メッセージ優先レベル」といって、メッセージ送受信処理にプライオリティを付け、プライオリティの高いメッセージを優先してサービスするという機能があります。

この機能は、例えば、オンラインメッセージには高いプライオリティを割付け、ファイル転送データには低いプライオリティを割付けることにより、ファイル転送をオンラインメッセージ処理の空いた時間に実行できるという利点があります。しかし、コントローラでは、ファイル転送タイプのデータはないことから、この機能はサポートしていません。

NXACPは、ネットワークにメッセージ送信時、ヘッダ内送信レベルに0（ゼロ）を設定します。

送信レベル0（ゼロ）は、優先レベルをサポートしているNeXUSファミリ（NX Dlinkなど）では、受信した場合の優先レベルを任意に割付けることができます。詳細は、各機器に対応したNeXUSサポートパッケージのマニュアルを参照してください。

また、ネットワークからメッセージを受信した場合は、ヘッダ内優先レベルを評価せず、到着順に処理します。

【制限事項】

NX Dlinkと接続する場合は、NX Dlink側で設定するメッセージ優先レベルに以下の制限事項がありますので注意してください。

NX DlinkからNXACP宛にロングメッセージ（2パケット以上で分割・再生が必要なメッセージ）を送信する場合は、1つの優先レベルを使用する必要があります。

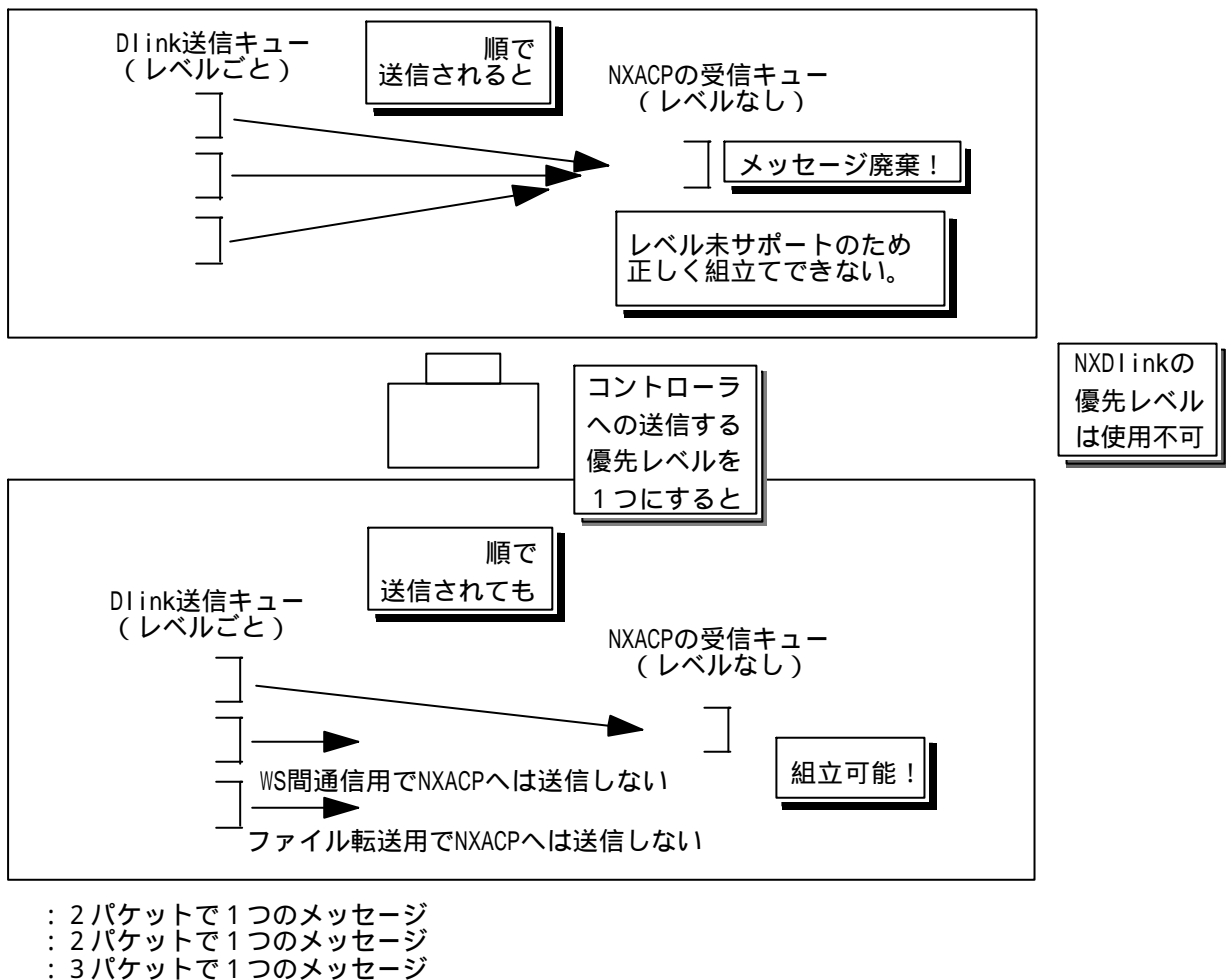


図2 - 5 優先レベル使用時の制限事項

2 機能ガイド

2.1.2 メッセージ送信機能

この節では、putran()マクロの送信機能について説明します。

ユーザタスクからputran()を発行する場合、送信先として“データフィールド番号 (DFN)”を指定する必要があります。

データフィールド番号に0以外を指定した場合、ネットワークに送信することを意味し、0を指定した場合は、自コントローラ内通信という特殊な意味を持っています。

ネットワーク送信 (DF = 0を指定した場合)

送信者と受信者が異なるノードに存在することをネットワーク送信と呼びます。

自ノード内送信 (DF = 0を指定した場合)

送信者と受信者が同じノード内に存在することを自ノード内送信と呼びます。

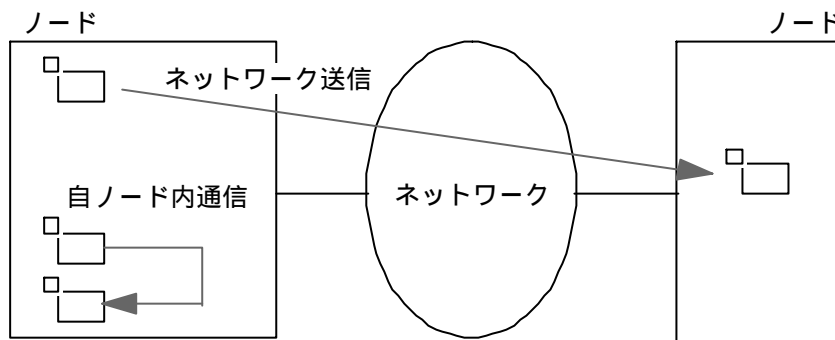


図2 - 6 データフィールド番号と送信範囲

(1) ネットワーク送信

(a) 送信範囲の指定方法

ネットワークに接続された他ノードへメッセージ送信するには、データフィールド番号に0以外の番号を指定して、送信要求する必要があります。

なお、送信に先立ち、構築情報にてデータフィールド番号とネットワークアドレス (INA) との対応付けが必要です。

つまり、送信したいネットワークセグメントのアドレスとデータフィールド番号を論理的に対応させ、送信時はネットワークアドレスでなくデータフィールド番号を指定します。下記例では、CTLからCPU11の接続されたネットワークにメッセージを送りたい場合、データフィールド番号は1を指定してputran()を発行します。

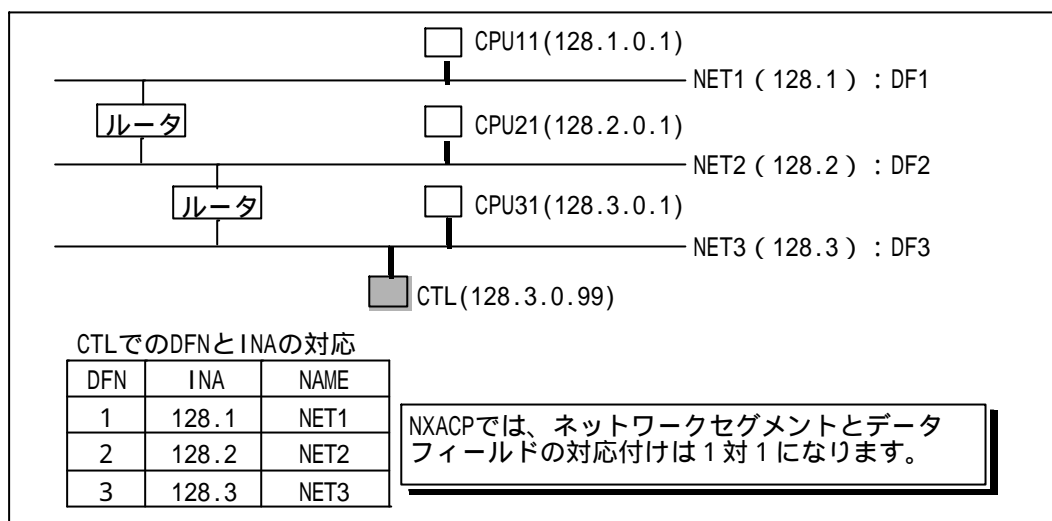


図2 - 7 データフィールド番号とネットワークセグメント

また、データフィールド番号の他に、マルチキャストグループ番号（MGN）とトランザクションコード（TCD）を指定する必要があります。

マルチキャストグループ番号も、以下のように、送信に先立ち、構築情報にてオンライン/テストモードごとに、ポート番号と対応付けておく必要があります。

表2 - 1 マルチキャストグループとポート

MGN	オンラインモード宛先ポート番号	テストモード宛先ポート番号
1	5 5 0 0 1	5 7 0 0 1
2	5 5 0 0 2	5 7 0 0 2
3	5 5 0 0 3	5 7 0 0 3
4	5 5 0 0 4	5 7 0 0 4

1つのマルチキャストグループには、モード別に2つのポートが対応する

NXACPでは、データフィールド番号で指定されたネットワークセグメントに、メッセージをブロードキャストします。ただし、このときの宛先ポート番号は、指定されたマルチキャストグループ番号に対応したポート番号です。

したがって、送信されたメッセージを受信できるのは、該当マルチキャストグループ番号に対応したポートを定義した（受信マルチキャストグループとして定義した）ノードだけとなります。ポートを定義していないノードでは、ハードウェアレベルにてメッセージを廃棄します。

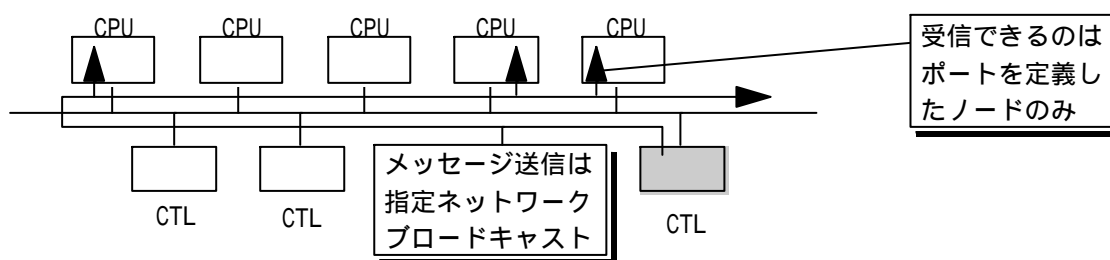


図2 - 8 マルチキャストと選択受信

2 機能ガイド

つまり、マルチキャストグループ番号を指定することにより、データフィールドに接続されるノードの中で、送信したいノードにだけメッセージが送信されます。なお、マルチキャストグループ番号とポート番号の対応付けは、データフィールドに接続される全ノードで統一してください。

トランザクションコードについては、受信先ノードでメッセージを渡す相手（NXACPでは、ユーザタスク）を判定するIDです。NXACPでのトランザクションコードとユーザタスクの関係については、「2.1.3 メッセージ受信機能」にて説明します。

(b) リモートデータフィールドへの送信

ルータを経由して接続されたリモートデータフィールドへメッセージを送信する場合も、データフィールド番号を指定して送信要求します。このとき、リモートデータフィールドへの送信経路は、1経路のみ設定できます。

図2 - 9において、ノードAからノードBにメッセージを送信する場合、データフィールド3経由で送信する経路とデータフィールド2経由で送信する経路がありますが、NXACPでは、どちらか一方をユーザの指定（構築情報）により静的に管理します。したがって、複数の経路を設定したり、障害発生時、異常を検出して経路を切替えることはできません。

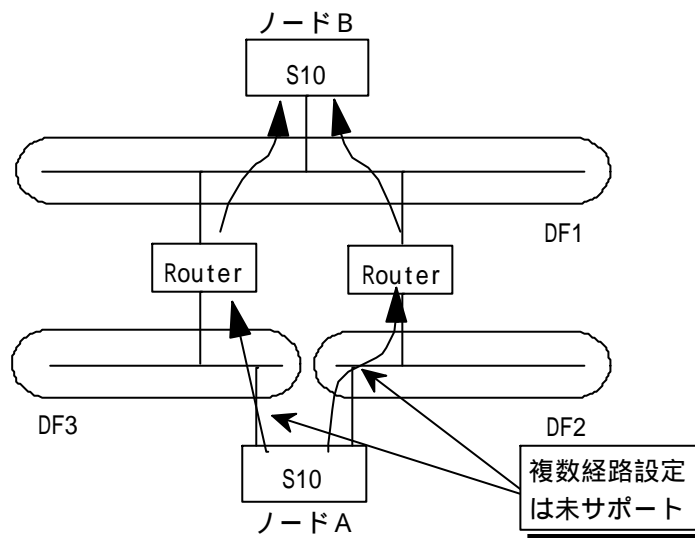


図2 - 9 リモートデータフィールド(1)

また、以下のような構築や送受信方法もできません。つまり、リモートデータフィールドは、1つのローカルデータフィールドに接続されている必要があります。

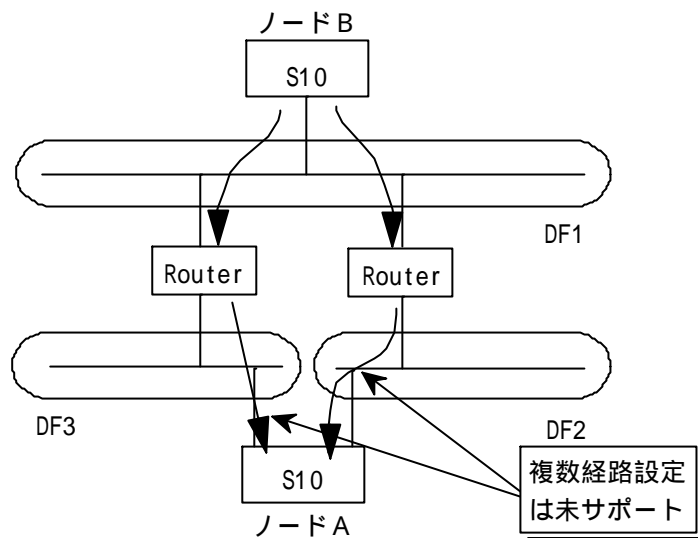


図 2 - 10 リモートデータフィールド (2)

リモートデータフィールドは、1つのローカルデータフィールドに対応し、経路を1つとしてください。

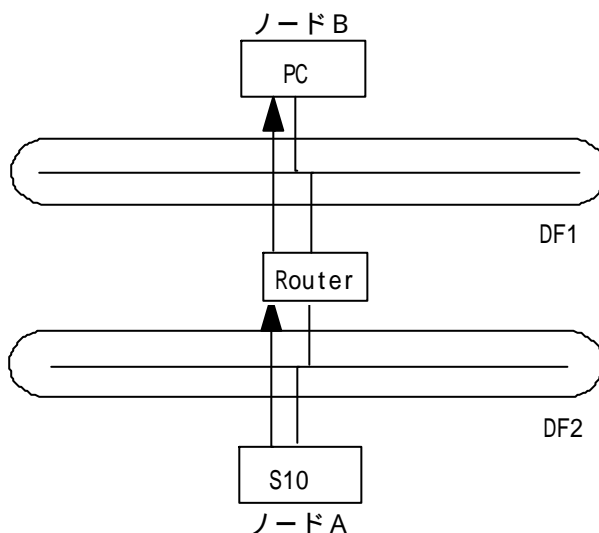


図 2 - 11 リモートデータフィールド (3)

【留意事項】

リモートデータフィールドは、システム設計時にリモート/ローカルデータフィールドのネットワークトラフィックや使用するルータの特性に注意して使用してください。

2 機能ガイド

(c) 自ノードへの折返しについて

データフィールド番号 0 へのメッセージ送信要求は、自ノードへの折返しもします。

自ノードへ折返したデータは、該当のTCDを受信するタスクが存在する場合、ユーザにデータを渡します（ただし、リモートデータフィールドへの送信要求に対する自ノードへの折返しはしません）。

(d) 送信バッファ管理

送信バッファは、データフィールド単位に管理します。したがって、1つのデータフィールドの送信バッファがビジー状態になっても、他のデータフィールドへの影響はありません。構築時は、データフィールド単位にバッファ面数を指定してください。

(e) 送信権チェック処理

NXACPを使用してユーザタスクがメッセージを送信するためには、構築時、ユーザタスクごとに送信するトランザクションコードを定義する必要があります。この定義情報に従いputran()では、送信元ユーザタスクと送信トランザクションコードが一致しているかチェックし、未定義トランザクションコードを送信しようとした場合、エラーとして処理を中断します。

データフィールドごとに、1TCDを送信できるユーザタスク数は最大16タスクです。

表2 - 2 トランザクションコードとユーザタスク対応

TCD	送信TN 1	送信TN 2	送信TN 3	...	送信TN16
3 3	2	4 1	2 5		7 0
4 8	4 1	-	-		-
7 6	3 7	8 0	5 5		-
9 9	2 5	3 9	4 0		4 1

ユーザタスクと送信トランザクションコードはデータフィールドごとに定義します。

(f) 特定ノードへの送信方法

NXACPでは、マルチキャスト送信のみサポートしているため、宛先ノードを指定した送信はできません。しかし、マルチキャストグループ番号、トランザクションコードを論理ノードと対応させて定義することにより、特定ノード宛送信にできます。

ここでは、以下に示す2とおりの例を示します。

(i) 宛先ノード番号 = マルチキャスト番号と定義する方法

マルチキャストグループごとに宛先ノードを対応するように定義することで、特定ノード宛送信を実現できます。

- ・各マルチキャストグループの受信ポート番号を各ノード専用に設定してください。二重化CPUの場合は、2つのノードに1つのマルチキャストグループを対応させてください。
- ・各ノード専用に定義したマルチキャストグループ番号を、データフィールド内の全ノードで統一してください。

表2 - 3 マルチキャストグループとノード対応

MGN	宛先ポート番号	対応ノード番号
1	5 5 0 0 1	A
2	5 5 0 0 2	B
3	5 5 0 0 3	C
4	5 5 0 0 4	D、E

全ノードで統一

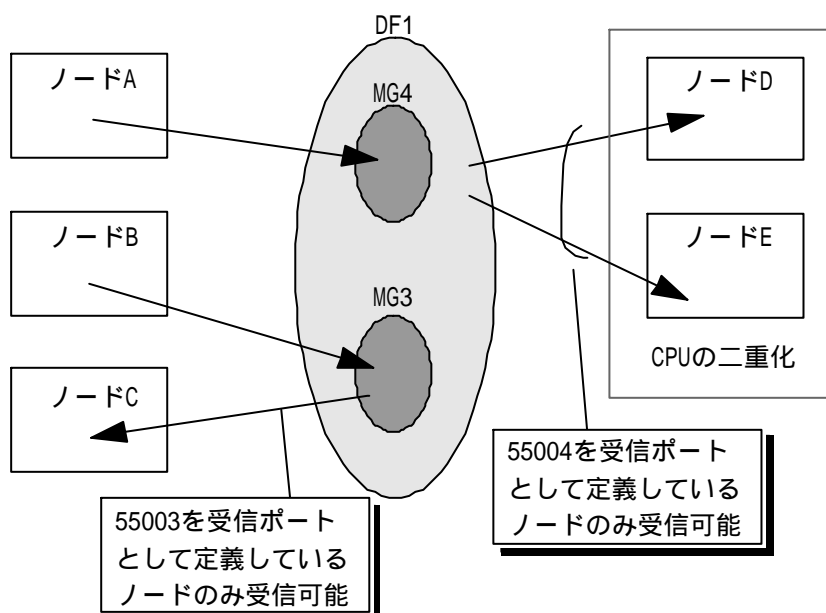


図2 - 12 特定ノード宛送信 (1)

NXACPでは、データフィールドごとに送信マルチキャストグループは最大16、受信マルチキャストグループは最大6グループまで定義できます。

2 機能ガイド

(ii) 宛先ノード番号 = トランザクションコードと定義する方法

トランザクションコードごとに宛先ノードとするように定義することで、特定ノード宛の送信ができます。

- ・各トランザクションコードを各ノード専用に設定してください。二重化CPUの場合は、2つのノードに1つのトランザクションコードを対応させてください。
- ・各ノード専用に定義したトランザクションコードは、データフィールド内の全ノードで統一してください。

表2 - 4 トランザクションコードとノード対応

MGN	TCD	対応ノード番号
1	1 ~ 1 0	A
1	1 1 ~ 2 0	B
1	2 1 ~ 3 0	C
2	3 1 ~ 4 0	D、E

全ノードで統一

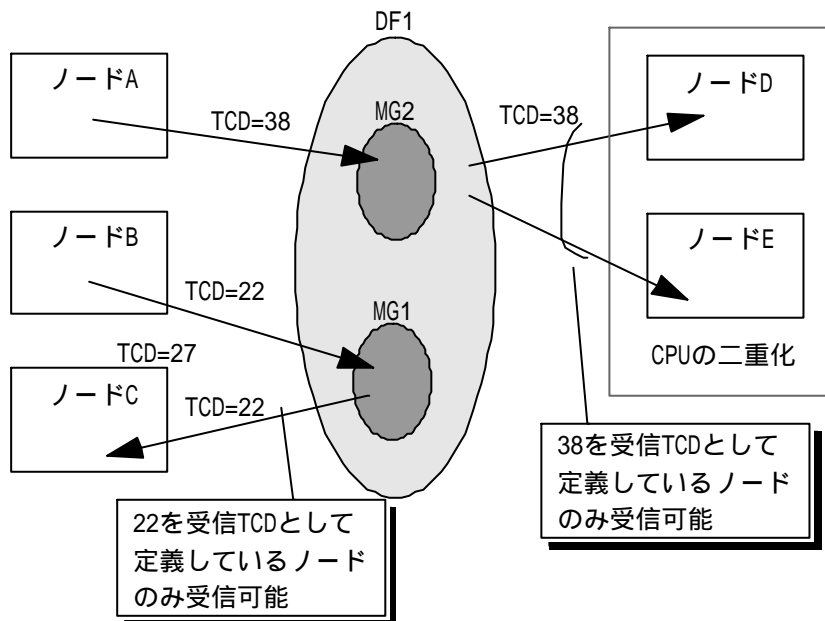


図2 - 13 特定ノード宛送信(2)

(i)と(ii)の方法を比較した場合、不要メッセージの廃棄を、(i)の場合ではハードウェアレベルで実施し、(ii)の場合はCPUのNXACPが実施します。したがって、CPU負荷を考慮した場合、(i)の使用方法を選択することを推奨します。

(2) 自ノード内送信

データフィールド番号0は、自ノード送信専用のデータフィールドです。

宛先データフィールド番号を0にしたメッセージを送信すると、自ノード内専用のバッファを使用しメッセージの受渡しをします。このデータフィールドは、外部の他ノードとのメッセージ送受信はできません。

データフィールド番号0を指定して送信する場合には、マルチキャストグループ番号は意味を持ちません。

トランザクションコードは、自ノード専用データフィールドのトランザクションコードとなります。これは、トランザクションコードがデータフィールド単位に一意であるためですが、受信タスクでは、他ノードからのメッセージと自ノード内メッセージを同じ手順にて受信できるため、ネットワークワイドで機能とトランザクションコードを割振ることを推奨します。

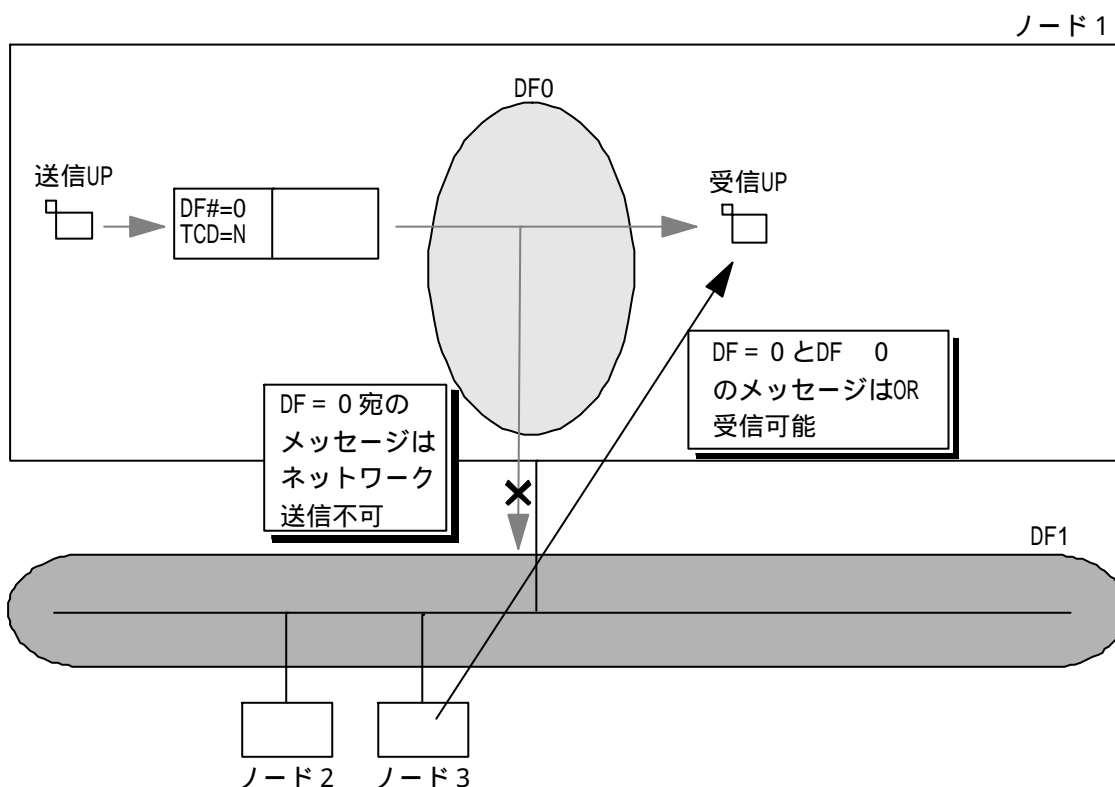


図2 - 14 自ノード内通信

自ノード内通信では、送受信バッファは1つで管理（受信バッファとして管理）します。

構築時、他データフィールド同様にバッファ面数を指定してください。なお、バッファサイズは1408バイト固定となります。

また、自ノード内通信では、常駐バッファ機能を使用して、一番最後に受信したメッセージを取得できます。常駐バッファにて使用するバッファサイズは構築時、TCD単位に設定してください。なお、常駐バッファは必ずしも定義する必要はありません。

2 機能ガイド

2. 1. 3 メッセージ受信機能

getran ()マクロの受信機能について説明します。

ユーザタスクからgetran ()を発行する場合、メッセージを受信するデータフィールドごとにトランザクションコードを定義しておく必要があります。この定義情報により、NXACPからメッセージを受け取ることができます。このとき、getran ()を発行するユーザタスクでは、自ノード内通信やノード間通信を意識することなく受信できます。

(1) メッセージ受信

getran ()でのデータ受信機能には、以下の特長があります。

- ・ 1つのユーザタスクで、受信するデータフィールドやトランザクションコードを複数定義できます。ただし、1回の受信要求で受信できるメッセージは1つだけです。
- ・ 1つの受信メッセージを複数のユーザタスクで共用受信できます。したがって、1つのメッセージに複数の業務データを格納し、伝送効率を上げることができます。

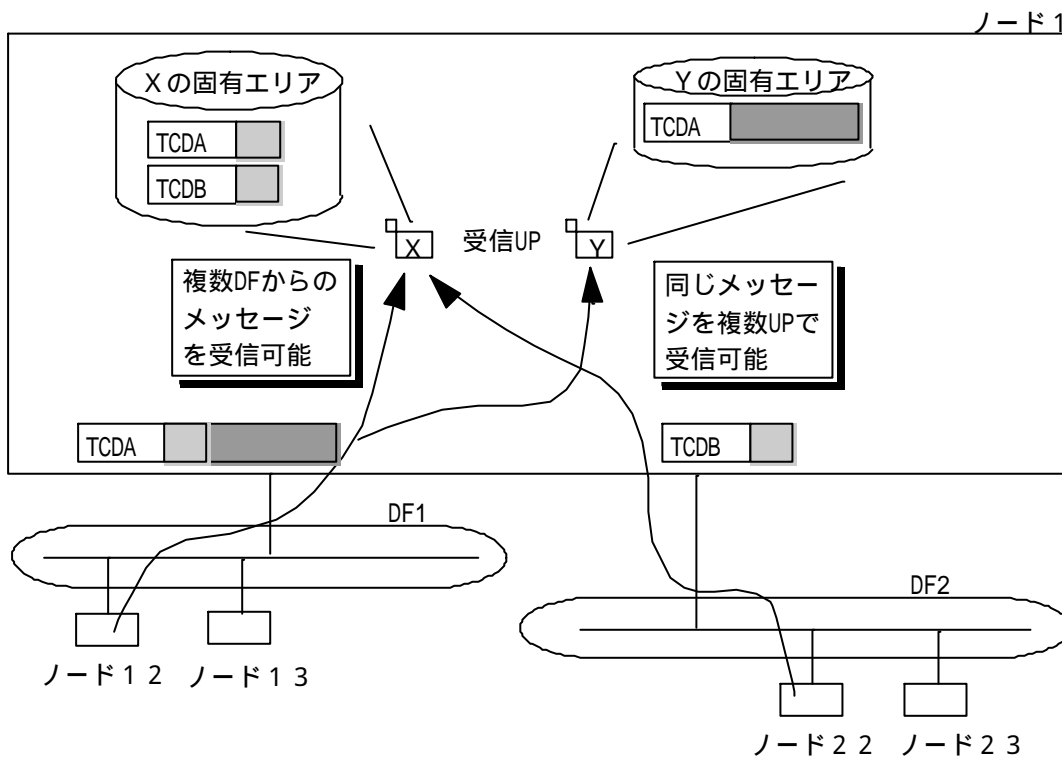


図 2 - 15 getran()の特長

(a) 定義情報と終了情報

NXACPを使用してユーザタスクがメッセージを受信するためには、構築時、ユーザタスクごとに受信するデータフィールド番号とトランザクションコードを定義する必要があります。

getran()のパラメータと分離しているのは、ユーザタスクと受信管理情報を分離させ、ユーザタスクの移植性を上げるためです。

データフィールドごとに、1ユーザタスクが受信できるトランザクションコード数の制限はありませんが、同じTCDを最大16タスクで受信することができます。また、1ユーザタスクで複数のデータフィールド番号にまたがった定義もできます。ただし、1回のgetran()発行にて受信できるメッセージは、1メッセージのみです。

受信したメッセージは先着優先にて処理されユーザタスクに渡されます。構築時指定したどのデータフィールドの、どのトランザクションコードかは、終了情報にてユーザに通知します。

表 2 - 5 トランザクションコードとユーザタスク対応

TCD	受信TN 1	受信TN 2	受信TN 3	...	受信TN16
3 3	1	4 1	2 5		7 0
4 8	4 1	-	-		-
7 6	3 7	8 0	5 5		-
9 9	2 5	3 9	4 0		4 1

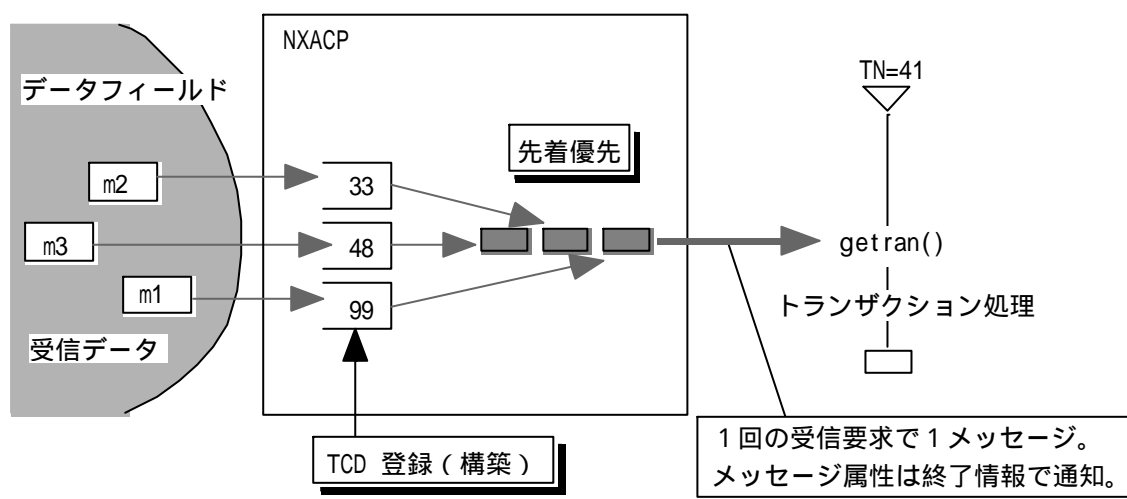


図 2 - 16 受信処理の流れ

2 機能ガイド

(b) トランザクションコード共有受信

1つのトランザクションコードメッセージを複数のユーザタスクにて受信できます。したがって、1つのメッセージに複数業務メッセージのデータを格納し、ネットワーク上での伝送効率を上げたり、ネットワークからの割込み発生回数を少なくすることで、CPUの負荷を下げるすることができます。また、ユーザタスクでも、複数業務メッセージを業務ごとに並列に実行できます。

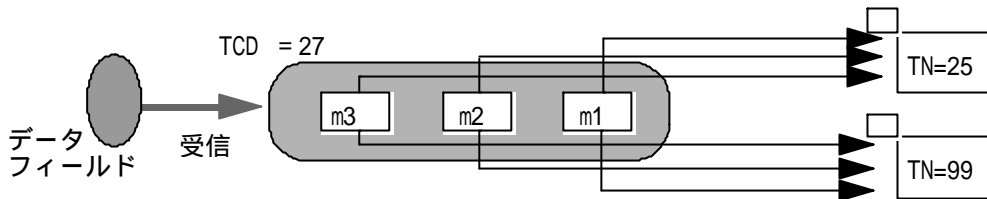


図 2 - 17 共有受信

【留意事項】

複数のユーザタスクに受信されるメッセージは、受信すべきすべてのユーザタスクが受信完了したときにバッファ解放します。したがって、処理時間やプライオリティに差があるユーザタスクにて同じトランザクションコードを受信する場合は、バッファ効率低下しいては受信バッファ満杯に伴うメッセージ廃棄を引き起こす可能性がありますので注意してください。

(c) 受信バッファ占有容量チェック

特定タスクの誤動作により受信バッファがオーバーフローしないよう、トランザクションコードごとにバッファの使用面数を定義できます。トランザクションコードごとに定義した面数を超過した場合、到着メッセージを廃棄します。この機能により、特定のトランザクションコードで受信バッファを占有することを防止します。

トランザクションコードごとに使用できるバッファの数は、構築時にユーザが指定します。廃棄されては困るトランザクションコードについては、バッファ面数を大きく確保してください。

(2) マルチキャストグループの定義

ネットワークセグメントに対応するデータフィールド（0のDF）では、受信するマルチキャストグループ番号と受信ポート番号の対応付けを定義する必要があります。この定義がない場合、該当マルチキャストグループ宛のメッセージを受信できません。この対応付けは、データフィールドに接続される全ノードにて統一する必要があります。

表 2 - 6 マルチキャストグループ番号と受信ポート番号

MGN	受信ポート番号
1	5 5 0 1
1 1	5 5 1 1

この対応付けを統一

送信時のマルチキャストグループ番号の定義にも宛先送信ポート番号を対応させますが、この宛先送信ポート番号と上記受信ポート番号は、同じマルチキャストグループ番号の場合、データフィールド内全ノードで統一する必要があります。

表 2 - 7 マルチキャストグループ番号と送受信ポート番号

MGN	宛先送信ポート番号
1	5 5 0 1
1 1	5 5 1 1

この対応付けを統一

MGN	受信ポート番号	宛先送信ポート番号
1	5 5 0 1	5 5 0 1
1 1	5 5 1 1	5 5 1 1

この対応付けを統一

受信ポートの定義は、MGN 1 を受信したいノードで定義します。宛先送信ポートの定義は、MGN 1 を送信したいノードで定義します。

他ノードでは、上記宛先送信ポート番号が受信ポートとして定義されていない場合、該当メッセージはハードウェアにて廃棄されます。宛先送信ポート番号と同じポート番号が受信ポートとして定義されているノードでのみメッセージは受信できます。

2 機能ガイド

ポートとマルチキャストグループ番号の関係を送受信ごとに示すと図 2 - 18 のようになります。

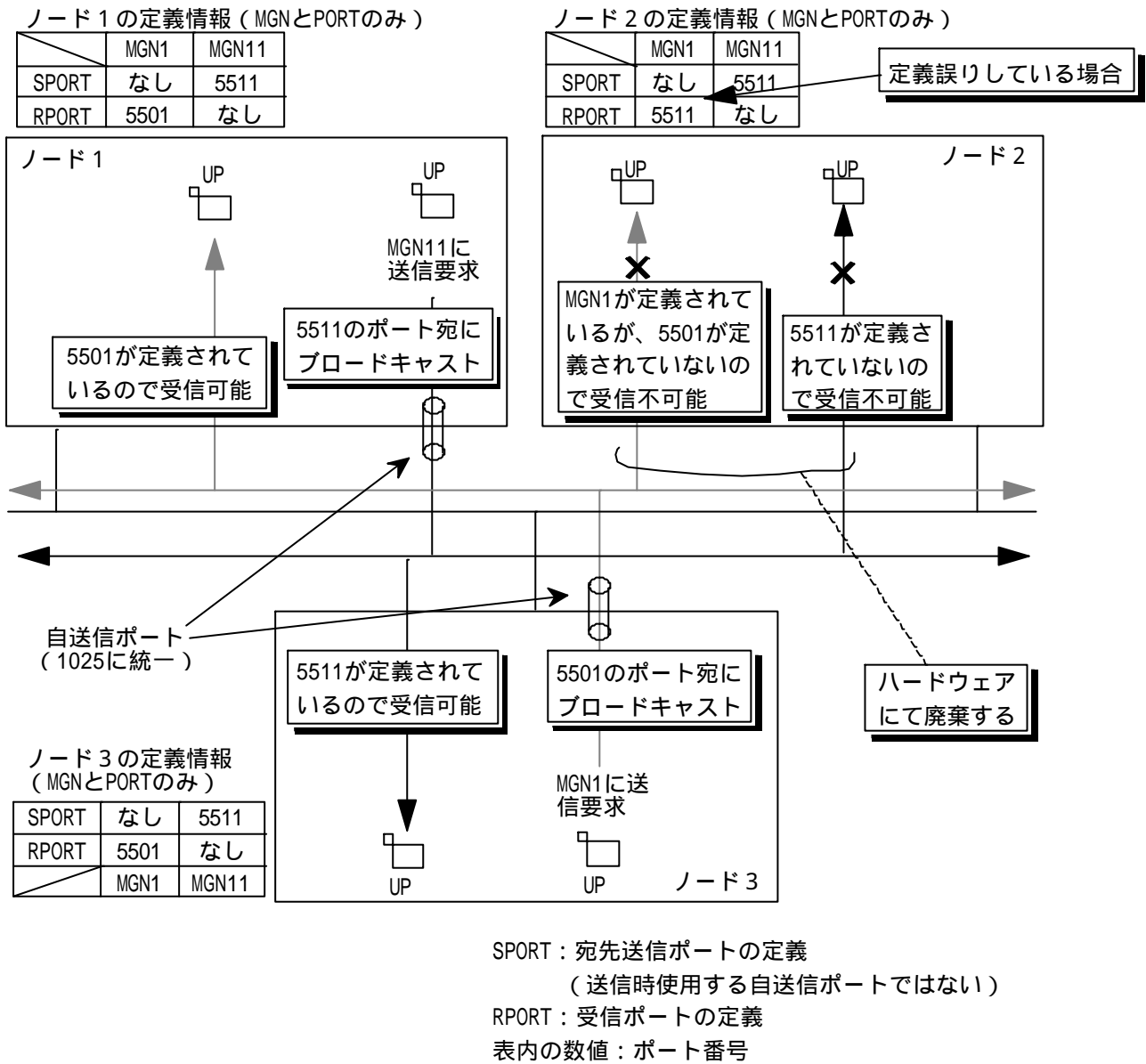


図 2 - 18 宛先送信マルチキャストグループと受信マルチキャストグループ

2. 1. 4 リモートデータフィールド制御機能

複数のLANをルータやノードで接続し、IP（Internet Protocol）のルーティング機能を利用することにより、大規模なネットワークにS10からアクセスできます。

(1) システム接続形態

大規模ネットワーク構成の場合、あるノードから見ると、自ノードが直接接続されているLANとルータなどを介して間接的に接続されるLANに分けることができます。前者をローカルデータフィールドと呼び、後者をリモートデータフィールドと呼びます。

リモートデータフィールドは、必ず1つのローカルデータフィールドに接続されていなければなりません。

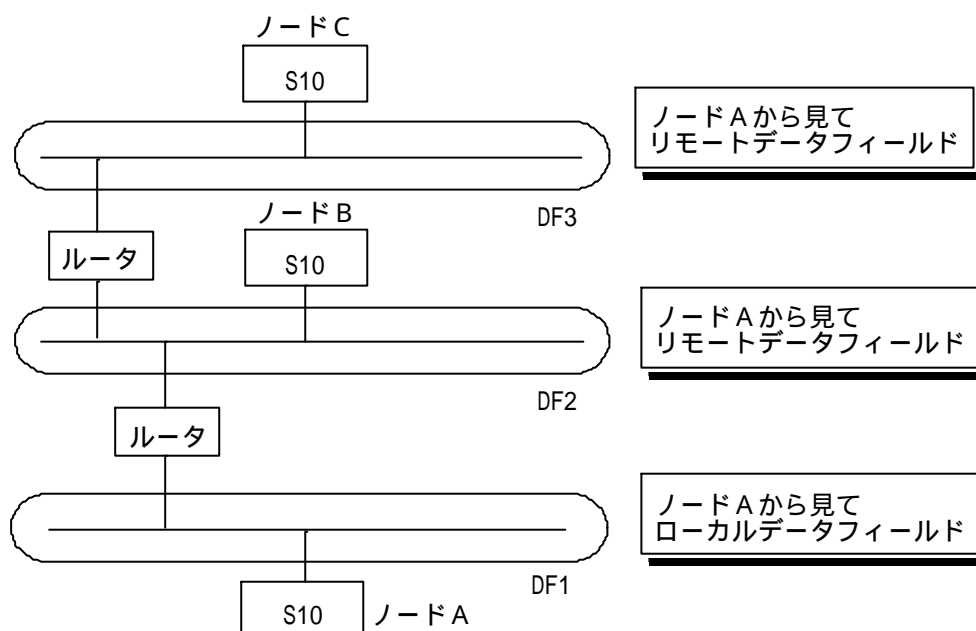


図2 - 19 データフィールドタイプ

【留意事項】

リモートデータフィールドへの送信は、IP（Internet Protocol）のルーティング機能を使用した機能です。

ルーティング機能は、NXACPではなくET.NETにて実現しています。

2 機能ガイド

リモートデータフィールドは、送信経路として、必ず1つのローカルデータフィールドに対応しなければなりません。リモートデータフィールドとローカルデータフィールドを1対Nに対応させることはできません。

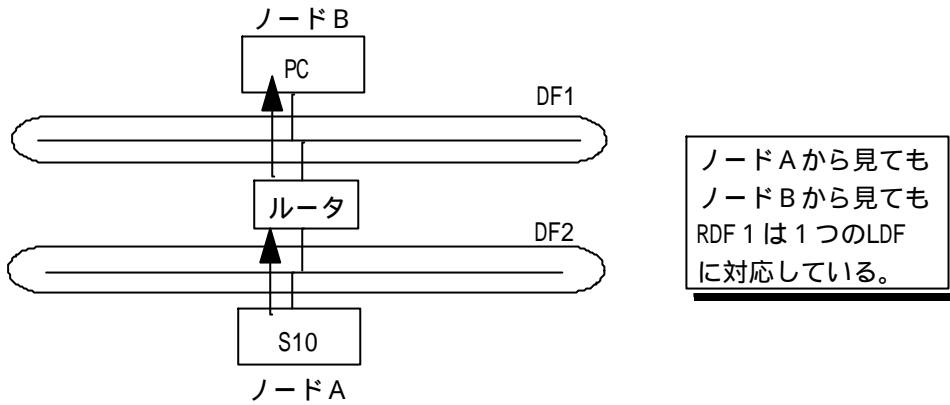


図 2 - 20 リモートデータフィールド (1)

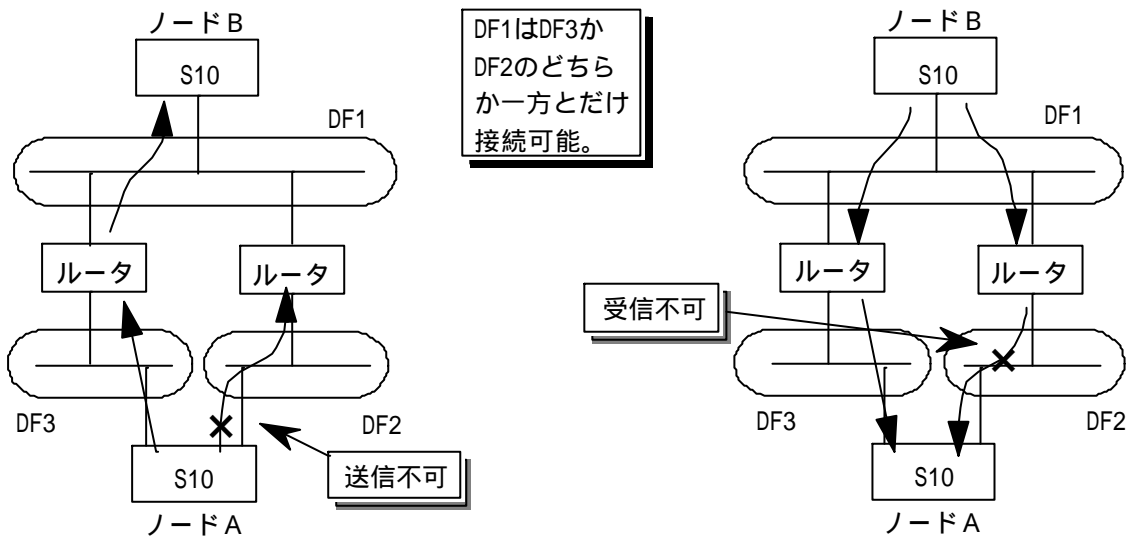


図 2 - 21 リモートデータフィールド (2)

(2) メッセージ送受信

(a) 通信の範囲

リモート/ローカルともに、そのデータフィールドで定義されているマルチキャストグループに対してメッセージを送信できます。ただし、1回の送信要求で複数のデータフィールドへ同時に送信できませんが、同じデータフィールド内の複数のマルチキャストグループへは送信できます。

受信に関して各ノードは、ローカルデータフィールドで定義されたマルチキャストグループには加入できますが、リモートのマルチキャストグループには加入できません。したがって、各ノードのローカルデータフィールドに送信されるメッセージは受信できますが、リモートデータフィールドに送信されたメッセージは受信できません。

(b) メッセージの送受信方法

メッセージを送信する場合、宛先データフィールドがリモート/ローカルに係わらず、宛先データフィールド番号とトランザクションコードを指定して送信要求 (`putran()`) を発行します。この場合、指定した宛先データフィールドでのみメッセージを受信できます。

ルータを幾つか介したリモートデータフィールドへ送信した場合でも、宛先データフィールド以外 (途中に存在するデータフィールドやローカルデータフィールド) では受信できません。

一方、受信メッセージは、送信元ノードがリモート/ローカルに係わらず、メッセージが送信されたローカルデータフィールドの受信キューに接続されます。したがって、メッセージを受信する場合は、ローカルデータフィールド番号とトランザクションコードを構築情報へ定義する必要があります。なお、この場合、受信できるメッセージはローカルデータフィールド宛に送信されたメッセージだけです。

なお、リモートデータフィールドへの送受信サービスは、リモートデータフィールドが属するローカルデータフィールドの処理プログラムが対応します。

2 機能ガイド

リモートデータフィールドとメッセージ送受信の関係をまとめると図 2 - 22 のようになります。

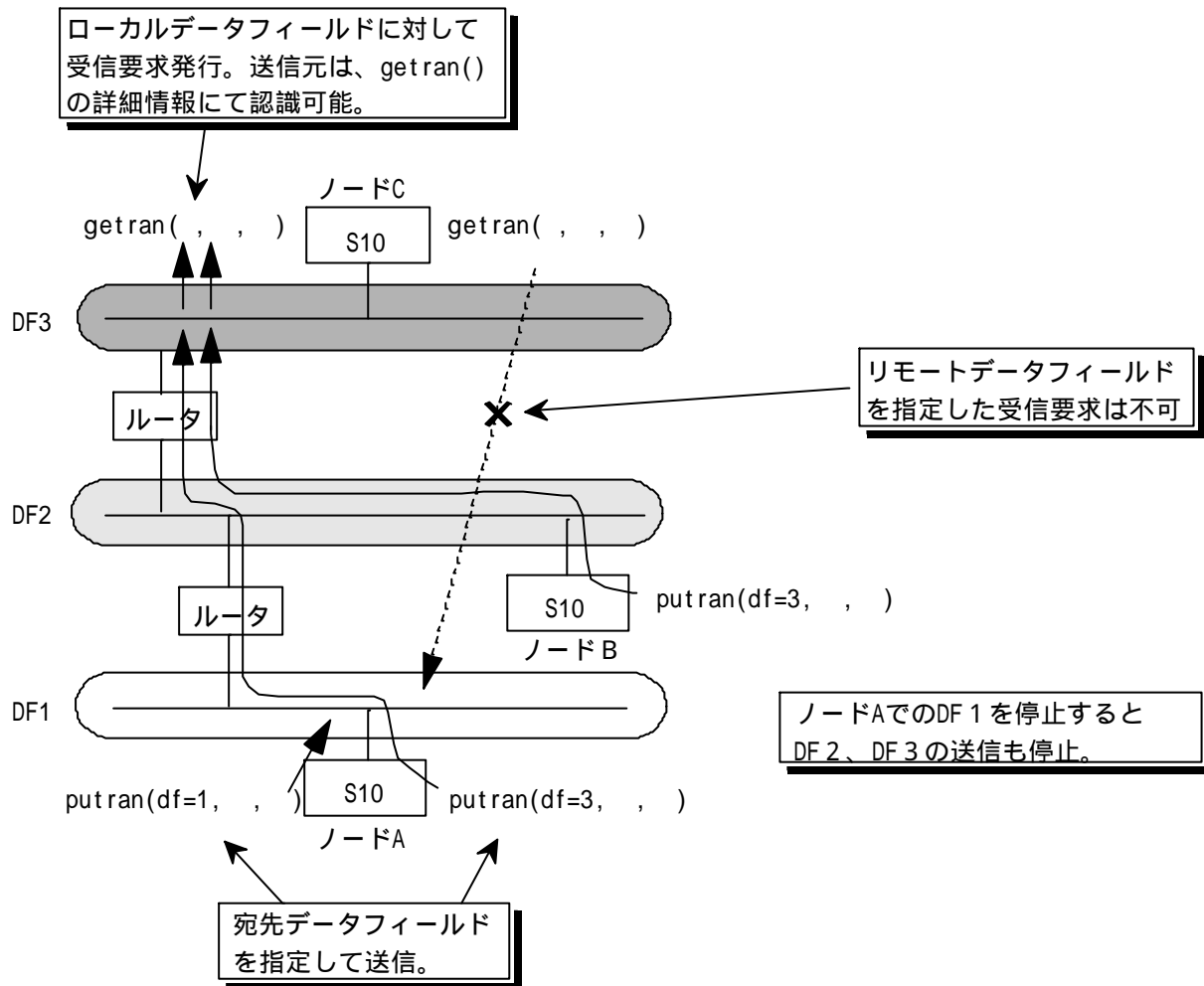


図 2 - 22 リモートデータフィールドのメッセージ送受信

2. 1. 5 バッファ管理

送受信バッファは、各データフィールドごとに送受信に分けて管理しますので、データフィールドで送受信ごとに定義してください。

表2 - 8 データフィールドタイプとバッファ定義

	ローカルデータフィールド (自ノード内専用)	ローカルデータフィールド (自ノード内除く)	リモートデータフィールド
送信バッファ	定義不要	定義要	定義不要
受信バッファ	定義要	定義要	定義不要

バッファオーバーフロー発生時には、生存信号内障害情報エリアにエラー情報を設定します。
ユーザはNx/Viewerなどにてエラー情報を検知できます。

2. 2 データフィールド管理機能

NXACPでは、データフィールドの状態管理機能として下記の機能があります。

- ・自ノードの生存信号送信

NeXUSを搭載した各ノードは、一定間隔にて、各ノードが直接接続するローカルデータフィールドに対して“生存信号”をブロードキャスト送信します。他ノードでは、この生存信号を受信することによりデータフィールドに接続した各ノードの生死状態を監視できます。

NeXUSプロトコルの生存信号には、以下の3つのモードがあります。

- ・生報告（通常モード）
- ・SHUTDOWN予告報告（計画的な機器停止）
- ・メンテナンス予告報告（メンテナンスのための一時的な機器停止）

NXACPでは、NeXUSプロトコル生存信号の3つのモードに対し、生報告とSHUTDOWN予告報告をサポートします。

2. 2. 1 生存信号送信機能

NXACPは、イニシャル要求を受付けると（acpinit()が発行されると）“生”報告の生存信号を、構築時指定された一定間隔にて送信します。また、停止要求を受付けると（acpquit()が発行されると）、“SHUTDOWN予告”報告の生存信号を送信します。この“SHUTDOWN予告”により、他ノードでは、計画停止なのか異常による停止なのかを判断できます。

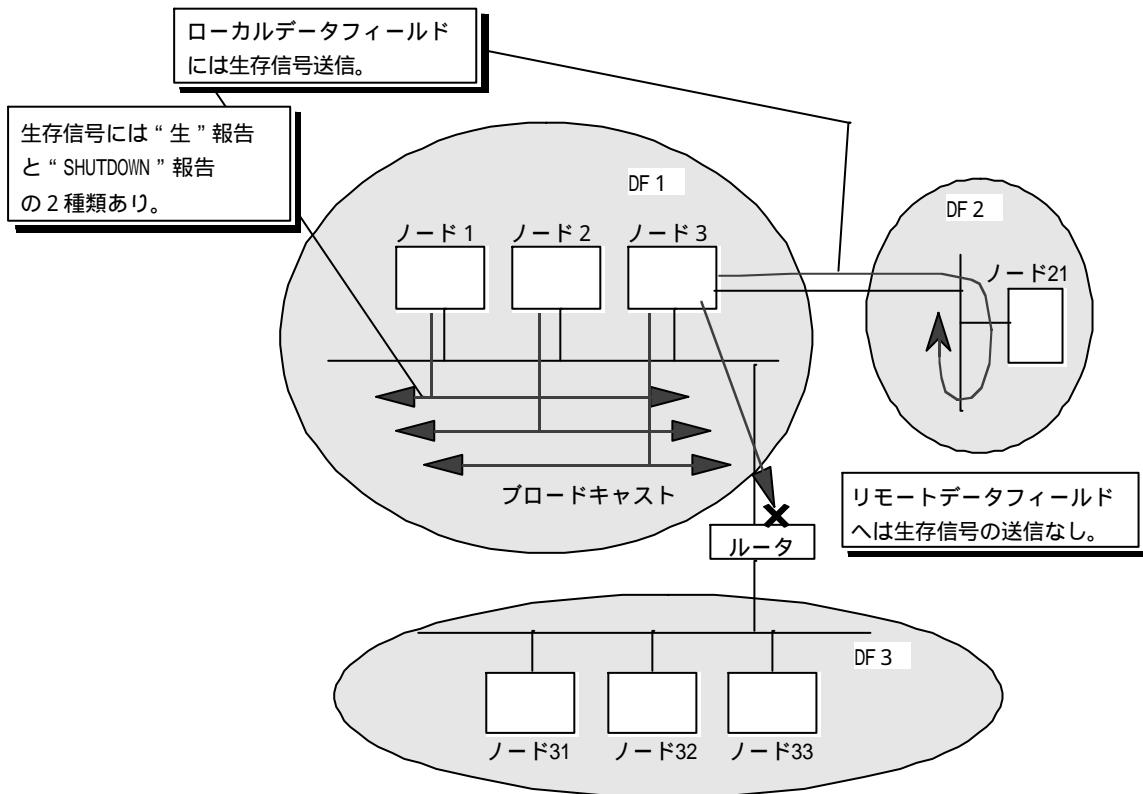


図 2 - 23 生存信号

生存信号により、データフィールド単位に他ノードのデータフィールドへの接続状態を監視することができます。ここで言う監視とは、システム内で特定ノードが動作しているかどうかではなく、業務のベースとなるデータフィールドから見て、各ノードがデータフィールドに接続され他ノードと協調しあいながら稼働できるかどうかの監視です。

例えば、図 2 - 23 に示す構成でノード 3 は、データフィールド 1 と 2 に接続されています。この構成でデータフィールド 1 とノード 3 の接点の通信機器（ケーブル、イーサモジュールなど）が故障した場合、データフィールド 1 に接続する他ノードは、ノード 3 が“生”から“死”状態に変化したと認識し、その旨をユーザに通知します。しかし、データフィールド 2 では、ノード 3 は他ノードからは“生”状態のままと認識され続けます。

NXACPの送信する生存信号のタイプを以下に示します。

表 2 - 9 生存信号タイプ

報告タイプ	生報告	SHUTDOWN 予告
送信開始タイミング	UPからのacpinit()要求	UPからのacpquit()要求
送信回数	停止要求まで無限	3回
送信周期	ユーザ指定（秒単位）	acpquit()発行直後と1秒後、2秒後

acpinit() / acpquit()（詳細説明は「2.5 運用管理機能」参照）は、NXACPの立上げ停止を要求するマクロです。

つまり、NXACPの立上げ要求をacpinit()にて発行すると同時に“生報告”の生存信号一定周期にて送信し始めます。

また、停止要求をacpquit()にて発行すると“SHUTDOWN 予告”の生存信号を、マクロ発行直後と1秒後、2秒後の3回だけ送信します。

“生報告”の送信周期は、ユーザが構築時に秒単位に指定できますが、ユーザが構築時に定義する情報としては他にも表 2 - 10 に示すものがあります。それぞれの情報は、生存信号に付加してブロードキャストされNX Martのシステム監視機能を使用した場合に参照できます。

表 2 - 10 生存信号付加情報

ユーザ定義情報	説明
ノード番号	データフィールド単位に付ける番号
ノード名称	データフィールド単位に付ける名称
生存信号タイムアウト監視時間	生存信号が未到達となってから状態が“死”に変化したと認識するまでの時間

2.3 テスト機能

テスト機能とは、ユーザプログラムを容易に種々のテスト形態にてテストするための支援機能です。システムを段階的に拡張していくうえで、既存のオンラインで動作しているコンポーネントを変更したり、テスト専用システムを作成することなく、ノード単位でテストできます。

NXACPでは、既存のコンポーネントに影響を与えることなくテストできるようにするために、ノードとメッセージのそれぞれにモードを付加し、このモードの組合せにより、ノード単位でのメッセージ入出力制御を行います。つまり、ノード単位に入出力するメッセージを限定することにより、システムの一部のコンポーネントだけを、論理的にシステムとは隔離して運用できます。

2.3.1 メッセージ入出力制御

メッセージ入出力制御方式は、ユーザが構築情報にて指定します。

メッセージ入出力制御の組合せで、ノードモードとメッセージモードの可能な組合せは以下のとおりです。

表2 - 11 ノードモードとメッセージモード

ノードモード	送信するメッセージモード	受信できるメッセージモード
オンライン	オンライン	オンライン
テスト	テスト	テスト
テスト	テスト	オンライン

オンラインノードでは、送信メッセージはオンラインメッセージのみとなり、受信できるメッセージもオンラインメッセージのみとなります。なお、テストメッセージ用のポートはイニシャライズしないため、ハードウェアレベルにて廃棄されますので、テストメッセージによるCPUへの擾乱はありません。

テストノードでは、送信メッセージはテストメッセージのみとなりますが、受信できるメッセージは以下のどちらかを選択できます。

- ・テストのみ
- ・オンラインのみ

どちらの入力方式を選択した場合でも、受信不要のモード用ポートをイニシャライズしないため、不要モードメッセージはハードウェアレベルにて廃棄されますので、受信不要のメッセージによる擾乱がコントローラに入ることはありません。

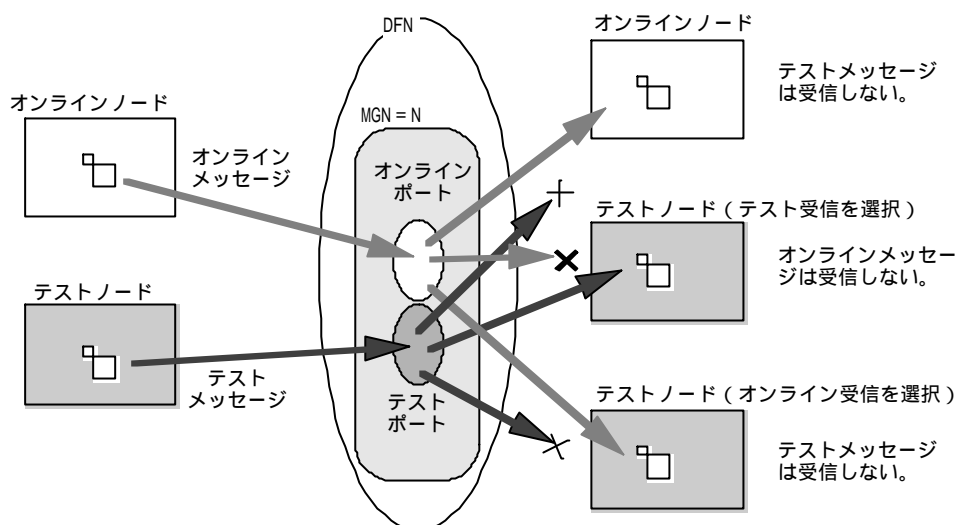


図 2 - 24 メッセージ入出力制御

図 2 - 24で、オンラインノードがNのマルチキャストグループにオンラインメッセージを送信する場合、NXACPは、Nのマルチキャストグループに対応するオンライン用ポート番号を送信宛先ポートとしてメッセージを送信します。したがって、ユーザはオンラインノードを構築時指定した場合、Nのマルチキャストグループに対応するオンライン用ポート番号を送信用として定義する必要があります。

また、Nのマルチキャストグループのオンラインメッセージを受信する場合、Nのマルチキャストグループに対応するオンライン用ポート番号を受信用として定義してください。

なお、上記Nのマルチキャストグループのオンライン用ポート番号は、データフィールドで同じ番号としてください。マルチキャストグループ1つに対し、オンライン用/テスト用と2つポート番号を対応させてください。

メッセージ入出力制御の組合せは、構築情報にて指定しますが、この場合、各マルチキャストグループに対応するオンライン/テストモード用のポート番号定義との対応を、データフィールドの全ノードで統一させ、このモードとマルチキャストグループ番号とポート番号の組合せがデータフィールドで統一されていることにより、必要メッセージの受信、ハードウェアレベルでの不要メッセージ廃棄ができるようになります。

NXACPが実際に使用するポートは、構築情報にてユーザが指定したモードに対応するポートだけです。

表 2 - 12 モードとポート番号

MGN	オンラインモードポート番号	テストモードポート番号
1	5 5 0 0 1	5 7 0 0 1
2 3	5 5 0 2 3	5 7 0 2 3
9 9	5 5 0 9 9	5 7 0 9 9
1 0 1	5 5 1 0 1	5 7 1 0 1

この対応をデータフィールド内の全ノードで統一する

2 機能ガイド

2.3.2 テスト形態

メッセージの入出力制御を利用したテスト形態の例を以下に示します。

(1) オンラインメッセージ受信によるテスト

S10のユーザプログラムを増設/変更する場合、テスト専用の新たなシステムを設定することなく、既存のオンラインメッセージのまま流用してテストできます。

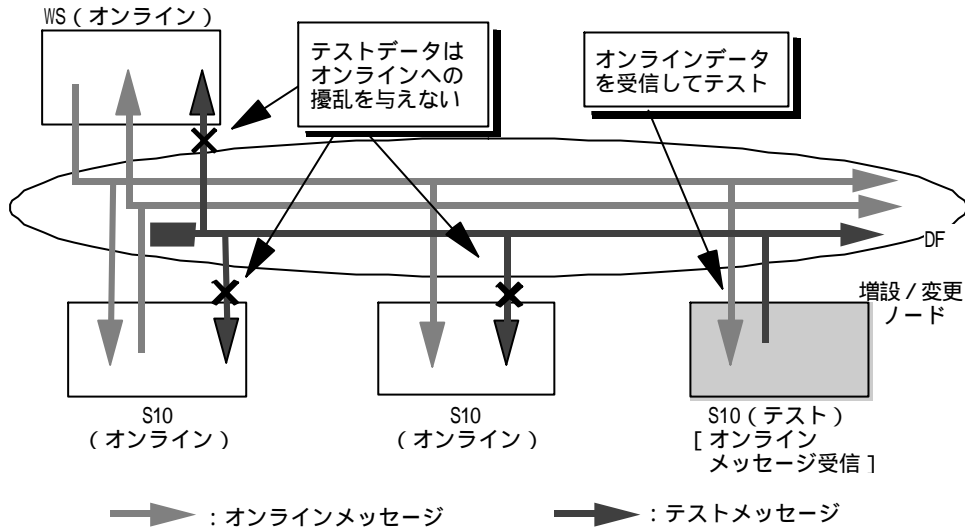


図 2 - 25 テスト形態 (1)

(2) テストメッセージ受信によるテスト

S10とWSを新規追加する場合などで、WSからテストメッセージを受信し、その返答をWSに返すような場合でも、新たなテスト環境を設定する必要がなく、オンラインネットワークに直接テスト用WSとS10を接続してテストできます。

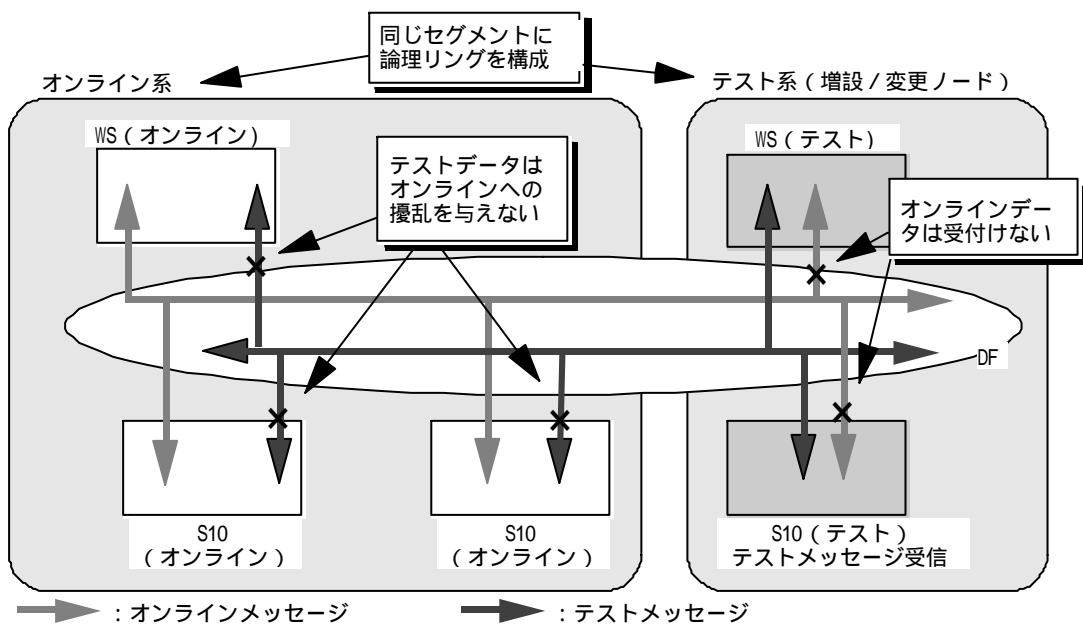


図 2 - 26 テスト形態 (2)

(3) テスト / オンラインメッセージ受信によるテスト

S10とWSを新規追加する場合で、オンラインWSからオンラインメッセージを受信し、その返答メッセージをテストWSに返すテストができます。NXDlinkのジャーナル機能を使用すれば、オンライン系のS10処理結果とテスト系のS10処理結果の照合も容易に行えます。

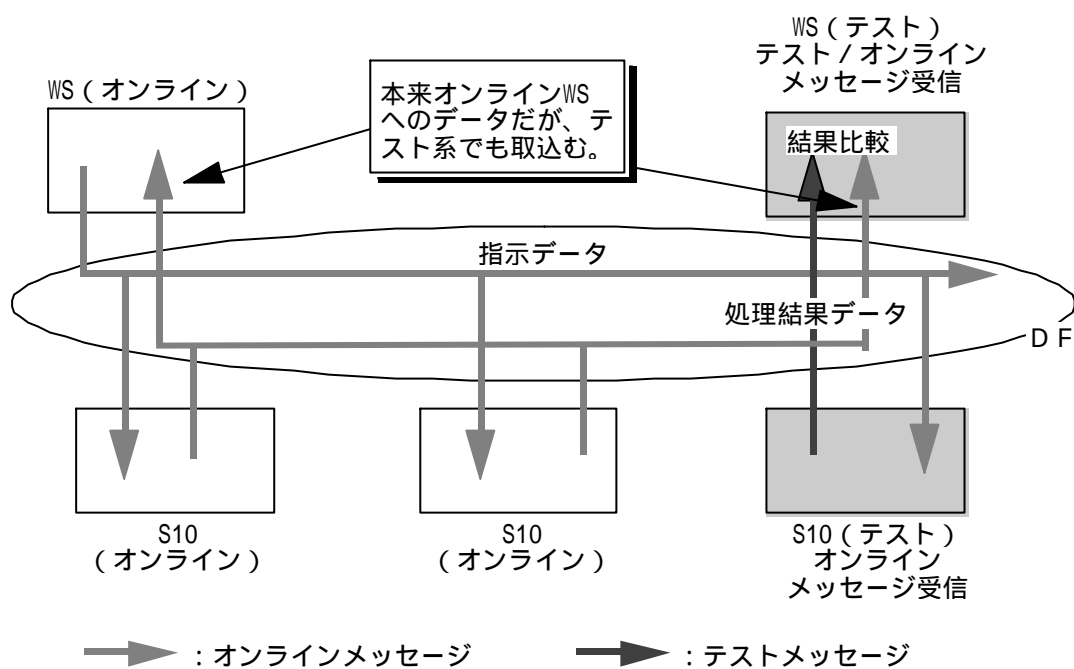


図 2 - 27 テスト形態 (3)

2 機能ガイド

2. 3. 3 リモートデータフィールドとモード

ノードモードはデータフィールド単位に設定できます。したがって、データフィールドに送信される送信メッセージモードもデータフィールド単位に設定できます。

例えば、下図のようにローカルデータフィールド（DF1）でのノードモードをオンラインに設定し、リモートデータフィールド（DF2）でのノードモードをテストに設定した場合、DF1にブロードキャストされるメッセージモードはオンラインモードになり、DF2にブロードキャストされるメッセージモードはテストモードになります。

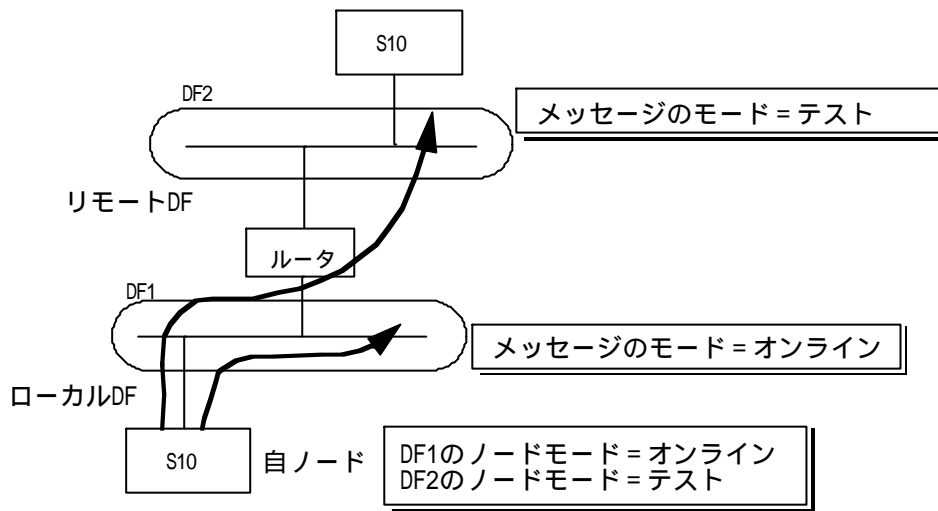


図2 - 28 リモートデータフィールドとモード(1)

リモートデータフィールドからのメッセージを受信するノードでは、リモートデータフィールドの送信元ノードのノードモード（すなわち送信されるメッセージのモード）と、自ノードが受信するメッセージのモードを一致させる必要があります。

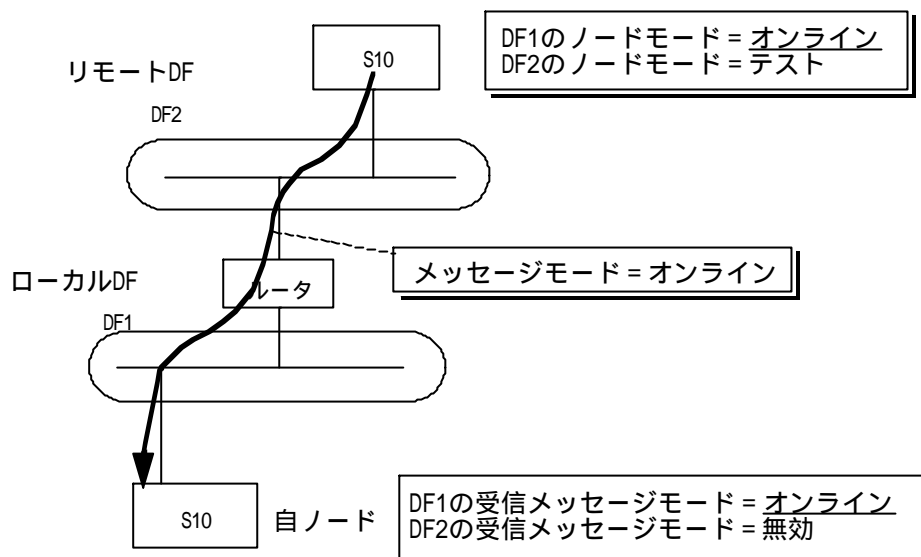


図 2 - 29 リモートデータフィールドとモード (2)

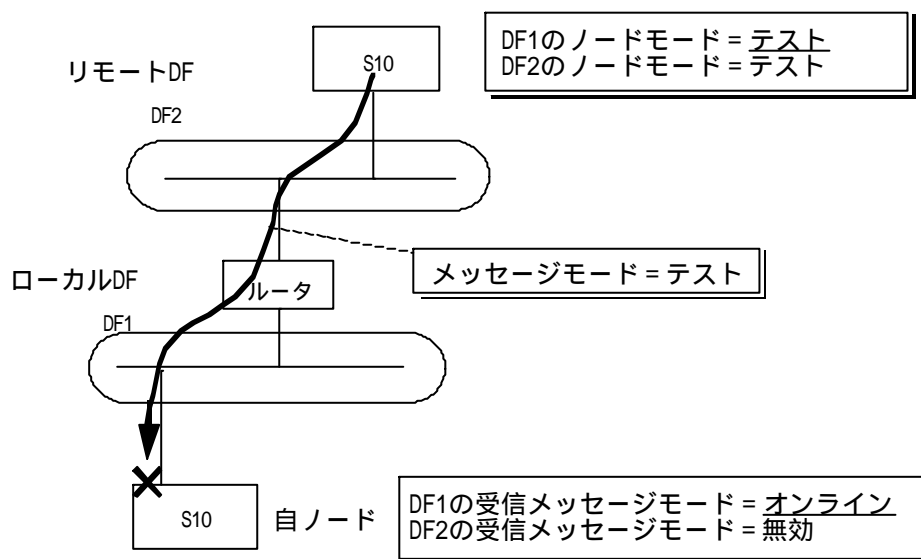


図 2 - 30 リモートデータフィールドとモード (3)

2.4 システム管理機能

NXACPでは、異常NeXUSプロトコルヘッダ受信時、ヘッダの内容をIRSUB#38に知らせます。これを障害通知機能と呼びます。また、ユーザタスク管理機能として、メッセージを受信するユーザタスクの状態を監視し、メッセージを受信できない状態の場合は、NXACPIにてメッセージを廃棄します。

これらの機能を纏めてシステム管理機能と呼びます。

また、上記以外にも、メッセージトレース機能があります。

2.4.1 障害通知機能

(1) プロトコル異常

プロトコル異常とは、送信する際に各メッセージに付加されるNeXUSヘッダ異常のことです。これは、送信/受信ノードでの構築誤りなどにより発生する可能性があります。

NXACPでは、上記障害を検知した場合、検知したヘッダ情報をそのままユーザに報告します。通知するフォーマットについては「付録C 異常NXヘッダ通知フォーマット」を参照してください。

[条件]

ヘッダ情報に異常を検知した場合に通知します。

[検知タイミング]

ネットワークからメッセージ受信時、ヘッダチェック処理にて検知します。

[条件を満たしたときの処理]

異常を検出したメッセージは、IRSUB 番号38番へ報告後にNXACPにて廃棄します。

IRSUB 番号38番への登録例は「付録C 異常NXヘッダ通知フォーマット」を参照してください。

[ユーザ対処]

送信元ノードとの構築情報を見直してください。

2.4.2 ユーザタスク管理機能

受信バッファは、ネットワークからデータ受信時に確保し、ユーザタスクが`getran()`を発行し受信したタイミングにて解放します。したがって、ユーザタスクが`getran()`を発行しない場合、受信バッファはオーバーフローしてしまいます。

NXACPでは、受信バッファのオーバーフローを防止するため、ユーザタスクの状態を監視し、ユーザタスクが`getran()`を発行できない状態の場合は、該当タスクの受信待ちメッセージや到着メッセージを廃棄します。

(1) メッセージ到着時のチェック

メッセージ到着時、該当トランザクションコードを入力するユーザタスクの状態をチェックし、DORMANT / NON-EXIST状態の場合、メッセージ配信をしません。また、入力する予定のユーザタスクがすべて上記状態の場合は、到達したメッセージを廃棄します。

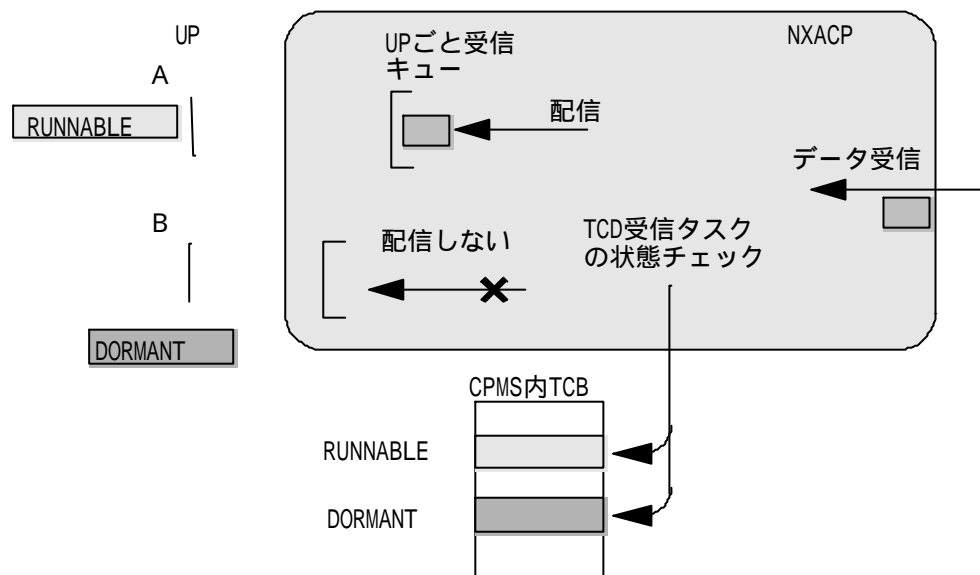


図 2 - 31 メッセージ到着時のタスク状態チェック

RUNNABLE状態のタスクAにはメッセージを配信しますが、DORMANT状態のタスクBには配信しません。

2 機能ガイド

(2) DORMANT 遷移時のチェック

ユーザタスクがDORMANTに遷移時、該当ユーザタスクの未受信メッセージをチェックし、未受信メッセージがあれば受信キューからパージします。また、入力する予定のユーザタスクがすべて受信完了している場合、またはすべてDORMANTになった場合は、メッセージを廃棄します。

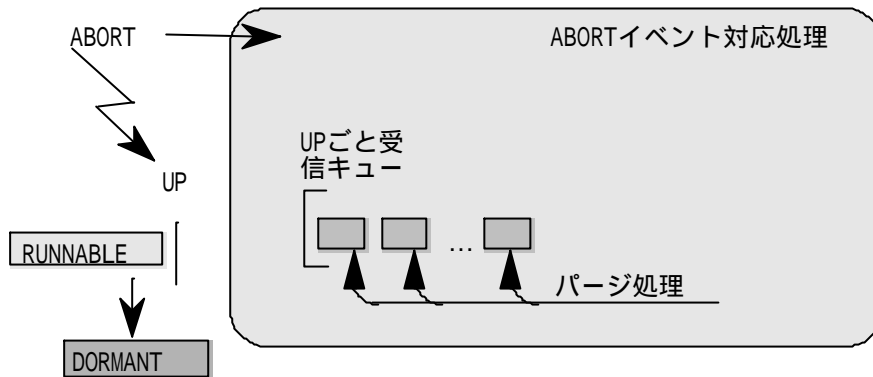


図 2 - 32 ABORT後処理

ユーザタスクがIDLEに遷移したタイミングでの上記受信キューのパージは行いません。

(3) EXIT 時のチェック

ユーザタスクが終了 (exit) したとき、NXACPでは該当ユーザタスクに対する未受信メッセージをチェックし、未受信メッセージがあれば組み込みサブルーチン (EXS) から該当のユーザタスクを再起動します。

2. 4. 3 メッセージトレース機能

NXACPには、デバグなどのためにメッセージトレース機能があります。

メッセージトレース機能は、NXACP内のトレースバッファをサイクリックに使用します。

メッセージトレースを取得するモードには、

- ・通信時の制御情報のみを取得するモード (最大取得ケース数：5)
- ・通信時の制御情報 + ユーザデータを取得するモード (最大取得ケース数：5)

の2つがあります。

ただし、通信時の制御情報 + ユーザデータを取得するモードを選択時は、トレースバッファを有効に使用するため、ユーザデータは先頭から1024バイトのみを取得します。

2.5 運用管理機能

NXACPの運用を以下に示します。

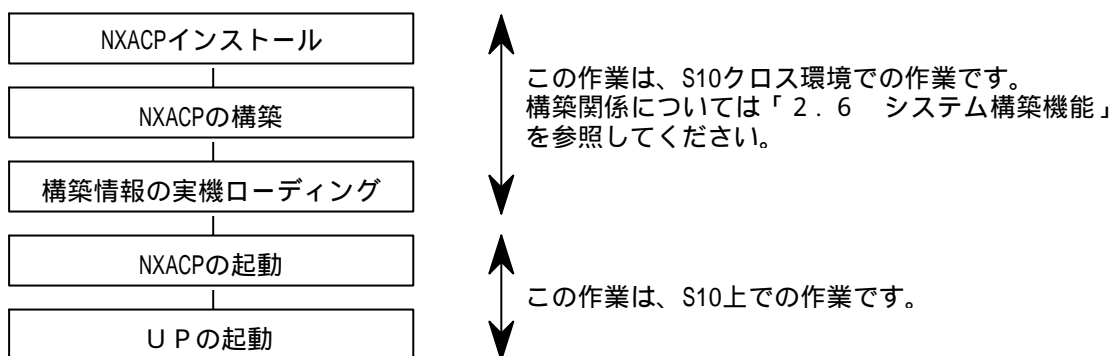


図2 - 33 NXACPの運用

S10実機でのNXACPの運用を以下に示します。

- ・NXACPの起動 / 停止

NeXUS機能を使用したサービス開始に先だって、NXACP本体を起動する必要があります。NXACP本体の起動は、システム立ち上げ時にUP起動の前に一度だけ発行してください。

2.5.1 NXACPとデータフィールドの運用管理

NXACPの起動 / 停止は、それぞれacpinit(), acpquit()マクロを用いて行います。

acpinit()マクロを発行することにより、データフィールドの運用を開始できるようになります。

acpquit()マクロを発行することにより、すべてのNXACPのサービスが停止します。

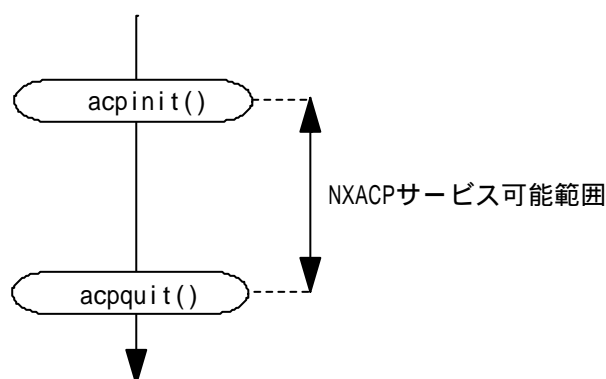


図2 - 34 各マクロと運用

2.6 システム構築機能

NXACPのインストールとシステム構築の手順を説明します。

NXACPは、ドライバ処理をするCPMSタスク（以下NXACPタスクと略します）とユーザインタフェース処理をするIRSUB（以下NXACPマクロと略します）にて動作し、オブジェクト本体はF/Dにて提供します。したがってNXACP本体は、ターゲットとなるS10のサイト情報を所有する開発系のパソコンからRPDPのコマンドを使用してインストールします。

また、NXACP使用にあたり、動作環境を構築コマンドにて設定する必要があります。

この環境設定も、ターゲットとなるS10のサイト情報を所有する開発系のパソコンにて実施し、S10のGLBエリアにローディング（以下NXACPテーブルと略します）します。

以下にNXACPのシステム構築手順とそれに対応するコマンド名称を示します。なお、インストールおよびシステム構築は、すでにサイト情報の構築が終わっていることを前提としています。

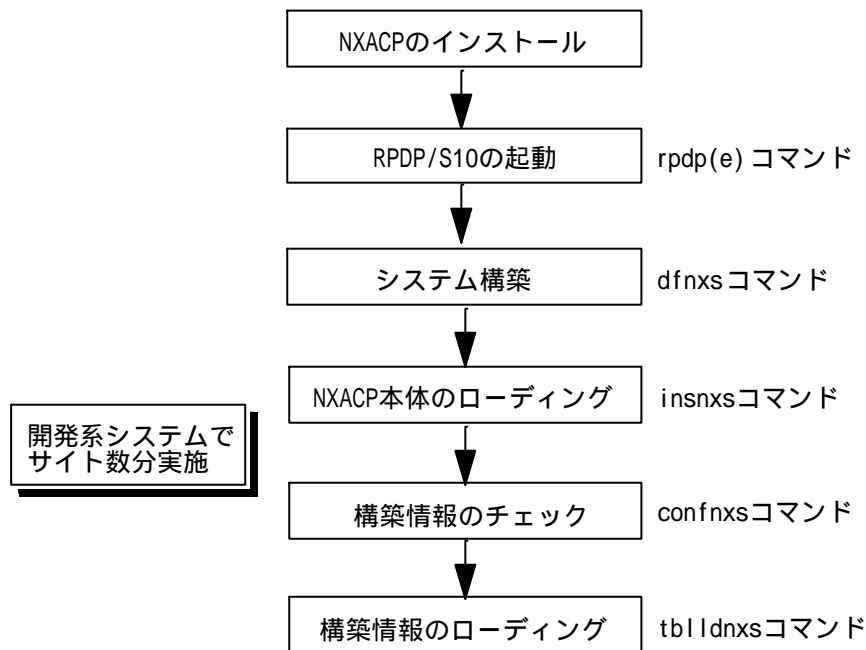


図 2 - 35 インストールと構築手順

NXACPの各リソースは、各インストールコマンドにより、以下のエリアにインストールされます。

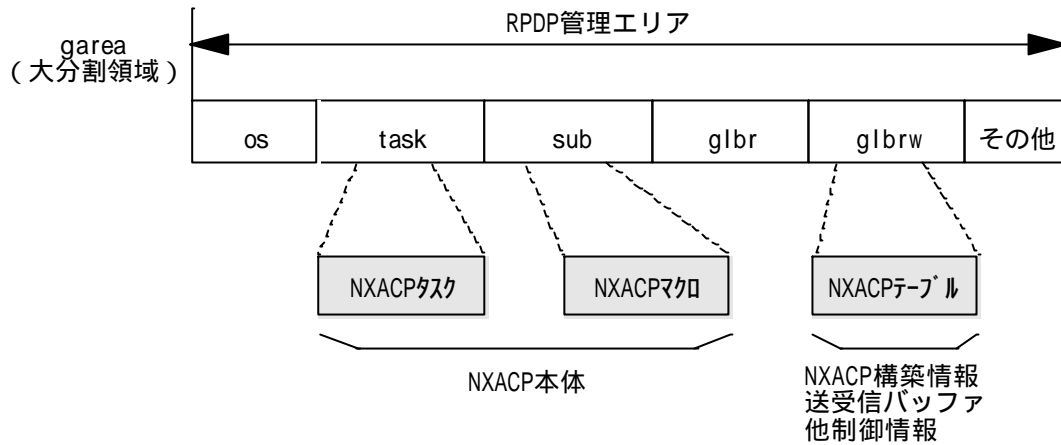


図 2 - 36 リソース配置

2. 6. 1 NXACPのインストール

F/DからNXACPをインストールする手順を以下に示します。

- (1) NX/ACP-S10のF/DをパソコンのF/Dドライブに挿入します。
- (2) 「エクスプローラ」、または「コントロールパネル」内の「アプリケーションの追加と削除」を起動し、F/D内のsetup.exeを起動します。
- (3) インストールディレクトリを変更したい場合は、そのディレクトリを指定してください。
(インストールディレクトリはRPDP/S10と同じディレクトリにしてください。)
- (4) インストール終了後、パソコンを再起動してください。

【注意事項】

NX/ACP-S10をインストールする前にRPDP/S10をインストールしてください。

2 機能ガイド

2.6.2 システム構築

NXACPを使用するためには、サイト単位に動作環境を構築しておいてください。

サイト単位の名義情報は、以下のコマンドにより初期設定してください。なお、下記コマンドは、初期構築時の1回だけ実行してください。

```
# df nxs サイト名称 <ret>
```

<エラー表示とその内容>

パラメータ数誤り：以下のエラーメッセージを表示します。

```
parameter number out of range
usage:df nxs site
```

このコマンドにより、指定サイト専用の下記ディレクトリに構築情報ファイルを配置します。
なお、この項ではドライブを、C：ドライブにて説明します。

ディレクトリ名称：“ c:¥hitachi¥alc¥サイト名称¥nxacps10 ”

ファイル名称：“ nxacp.conf.txt ”（共通情報）

“ df.txt ”（データフィールド情報）

“ mcg.txt ”（マルチキャストグループ情報）

“ revtcd.txt ”（受信トランザクションコード情報）

“ sndtcd.txt ”（送信トランザクションコード情報）

2.6.3 NXACP本体のローディング

NXACPタスク、NXACPマクロなどのNXACP本体をサイト情報下のバックアップファイルへローディングするには、以下のコマンドを起動します。

```
# insnxs サイト名称 [OFF] [-t TASKNO] <ret>
```

このコマンドにより、サイト名称として指定したS10のバックアップファイルに、NXACP本体をローディングします。ターゲットとなるS10への反映は、RPDPのリモートローディングコマンド（srpl）を使用してください。

なお、オプション機能は以下となります。

（オプション機能を未指定時は、NXACPが自動で登録します。）

- OFF : NXACP用組み込みサブルーチン（ABS, EXS）を組み込まない場合に指定します。この場合、ユーザがNXACP用の組み込みサブルーチンを登録する必要があります。
- t TASKNO : NXACPが使用するタスクの先頭タスク番号を指定します。tオプションが指定されない場合は、109～114にNXACP用のタスクを登録します。

<エラー表示とその内容>

パラメータ数誤り：以下のエラーメッセージを表示します。

```
parameter number out of range
usage:insnxs site [OFF] [-t TASKNO]
```

実行中のエラー：RPDP関連のエラーを表示した場合、「ソフトウェアマニュアル オペレーション RPDP/S10 For Windows®」（マニュアル番号 SAJ-3-133）を参照してください。

<実行中の表示とその内容>

コマンドが正常に動作している場合、このコマンドは以下のメッセージを表示します。

```
*** NX/ACP-S10 INSTALL START(サイト名称) ***
```

```
| ← 実行コマンドを表示します。
```

```
*** NX/ACP-S10 INSTALL END ***
```

2 機能ガイド

【参考】以下にNXACPのリソース割付けを示します。

表 2 - 13 リソース割付け

プログラム名称	タスク番号	IRサブ番号	ULサブ番号	タスクレベル	格納エリア名称 (容量)
nx_timer	1 0 9	-	-	1	task/nxacpt (3 2 Kバイト)
nx_upexe	1 1 0	-	-	1	
nx_rcv1	1 1 1	-	-	1	
nx_rcv2	1 1 2	-	-	1	
nx_snd1	1 1 3	-	-	1	
nx_snd2	1 1 4	-	-	1	
acpinit	-	2	-	-	sub/nxacpm (3 2 Kバイト)
acpquit	-	3	-	-	
putran	-	4	-	-	
getran	-	5	-	-	
getmsg	-	6	-	-	
sdtimer	-	7	-	-	
stimer	-	8	-	-	
swake	-	9	-	-	
nx_abs	-	1 0	-	-	
nx_exs	-	1 1	-	-	
nx_ulxs	-	-	4 (EXS)	-	sub/nxacps (3 2 Kバイト)
nx_ulabs	-	-	5 (ABS)	-	

【補足】

TASK NO : 109 ~ 114は、NXACP用に空けてください。

IRSUB NO : 1 ~ 38は、NXACPにてリザーブしています。

IRGLB NO : 1 ~ 50は、NXACPにてリザーブしています。

【留意事項】

NXACPを使用するユーザタスクは、NXACPタスクより低いタスクレベルを使用してください。

2.6.4 構築情報とローディング

(1) 構築情報の設定

サイト単位の各定義情報は、dfnxsコマンドにて指定サイト専用ディレクトリに格納された各定義原本ファイル名称を変更し、定義情報を設定することで行います。各ファイルの名称はcopyコマンドなどにて変更してください。

各ファイルの内容設定はnotepadなどのエディタで編集してください。

表2 - 14 構築ファイル一覧

分類	原本ファイル名称	作成ファイル名称	データベース管理内容
NXACP 共通定義	nxacp.conf.txt	nxacp.conf.txt	NXACPの基本的な設定値を管理します。
DF定義	df.txt	dfN.txt	データフィールドごとの設定値を管理し、ノード番号などの情報を定義してください。Nはデータフィールド番号を指定し、0~255まで指定できます。
MGN定義	mcg.txt	mcgN.txt	マルチキャストグループ番号ごとのオンライン/テスト用UDP宛先ポート番号を管理します。Nはデータフィールド番号を指定し、1~255まで指定できます。
TCD定義	rcvtcd.txt sndtvd.txt	rcvtcdN.txt sndtcdN.txt	トランザクションコードごとの設定値を管理します。rcvtcdN.txtは、TCDメッセージを受信するユーザタスク番号を定義します。sndtcdN.txtは、TCDメッセージを送信するユーザタスク番号を定義します。Nはデータフィールド番号を指定し、0~255まで指定できます。

各々の定義情報ファイルについて説明します。説明に先だって定義ファイルの記述方法を以下に示します。

定義ファイルは、以下の規則に従って作成してください。

< 作成規則 >

複数行定義の扱い

各定義情報は1行内で完結させ、複数行にまたがらないようにしてください。

(¥による行継続はできません。)

定義できる文字

半角の英数字(数字は10進数)のみ定義できます。空白、タブ、空行は読み飛ばします。

行の先頭文字が#(シャープ)の場合、該当行をコメントと判断します。

記号、全角文字は使用できません。

定義順番

昇順、降順を問わずランダム定義できます。

2 機能ガイド

(a) 共通定義ファイル (nxacp.conf.txt) 設定情報

共通定義ファイルには、以下の情報を設定してください。

個々の設定が不要の場合は、“未使用時”の欄に示す数値を設定してください。また、“未使用時”の欄が“-”の項目については、最適な値を設定してください。

認識子	内容	指定できる範囲	未使用時
MAXDFCNT	登録データフィールド総数	1 ~ 5	-
DFNO	データフィールド番号	0 ~ 255(*1)	-
TRCMODE	メッセージトレースモード 0 : 未取得 5 : 制御情報のみ取得 15 : 制御情報 + メッセージを取得	0、5、または15	0
NCBUFFG	常駐バッファ使用フラグ 0 : 使用しない 1 : 使用する	0 ~ 1	0
RLEASMODE	NXACP立上げ時の ユーザタスク起動 (releas) モード 0 : releas しない 1 : releas する	0 ~ 1	0

(*1) 定義したデータフィールド番号にて後述する“dfN”を作成してください。下記定義例では、“DFNO 1”に対して“df1.txt”を作成してください。

< 定義例 >

MAXDFCNT	1
DFNO	1
TRCMODE	0
NCBUFFG	0
RLEASMODE	1

(b) データフィールド定義ファイル (dfN.txt) 設定情報

データフィールド定義ファイルには、以下の情報を設定してください。

右端の I / L / R は、データフィールドのタイプ (I : 自ノード内, L : ローカル, R : リモート) と指定の有無 (: 指定要, × : 不要 (指定無効)、 : ローカルDFの定義に従う) を示しています。

個々の設定が不要の場合は、“未使用時”の欄に示す数値を設定してください。また、“未使用時”の欄が“-”の項目については、最適な値を設定してください。

認識子	内容	指定できる範囲	未使用時	L	R	I
DFTYPE	データフィールドタイプ 0 : ローカル、 0 : リモート(*1)	0, 1 ~ 255	-			×
NODENAME	自ノード名称	ASCII9文字以内	-		×	×
NODENO	自ノード番号	1 ~ 4095	-		×	×
NODEMODE	ノードモード 0 : オンライン、 1 : テスト	0 ~ 1	-			×
SSPORT	送信用自ポート番号	1 ~ 65535	-			×
ALPORT	生存信号宛先ポート番号	1 ~ 65535	-		×	×
ALSCYC	生存信号送信周期 (秒)	1 ~ 3600	-		×	×
ALTOUT	生存信号タイムアウト時間 (秒)	1 ~ 3600	-		×	×
DTSCYC	メッセージ送信サービス周期 (ミリ秒)	5 ~ 1000	-			×
LANMODE	LANモジュール番号 1 : メインモジュール、 2 : サブモジュール	1 ~ 2	-		×	×
MINSNMCG	最小送信マルチキャストグループ番号	1 ~ 255	0			×
MAXSNMCG	最大送信マルチキャストグループ番号	1 ~ 255	0			×
SNMCGCNT	送信マルチキャストグループ総数	1 ~ 16	0			×
MINRCVMCG	最小受信マルチキャストグループ番号	1 ~ 255	0		×	×
MAXRCVMCG	最大送信マルチキャストグループ番号	1 ~ 255	0		×	×
RCVMCGCNT	受信マルチキャストグループ総数	1 ~ 6	0		×	×
MINUSETCD	最小送受信 T C D 番号	1 ~ 59999	-			
MAXUSETCD	最大送受信 T C D 番号	1 ~ 59999	-			
USETCDCNT	送受信 T C D 総数	1 ~ 500	-			
SBUFCNT	送信バッファケース数	1 ~ 256	0		×	×
RBUFCNT	受信バッファケース数	32 ~ 256	0		×	×
MINNODENO	メッセージ受信可能最小ノード番号	1 ~ 4095	0			×
MAXNODENO	メッセージ受信可能最大ノード番号	1 ~ 4095	0			×
R_INA1	L A N ネットワークアドレス	XX.XX.XX.XX	0	×		×
R_NMASK1	L A N ネットマスク	XX.XX.XX.XX	0	×		×
R_GTWAYA1	ゲートウェイマシンのIPアドレス	XX.XX.XX.XX	0	×		×
R_METRIC1	メトリック数	1 ~ 16	0	×		×

(*1) データフィールドタイプがリモートデータフィールドとなる場合には、該当のローカルデータフィールド番号を指定してください。

<メモリの有効な利用方法>
 受信する T C D , M G N は、各ノードごとに連続して使用してください。
 またノード番号は、各データフィールドごとに連続して設定してください。

【留意事項】

各認識子の留意事項を以下に示します。

- MINUSETCD : 送受信するTCDの最小値を定義してください。
- MAXUSETCD : 送受信するTCDの最大値を定義してください。
- USETCDCNT : 送受信するTCDの総数を定義してください。
- R_INA1 : リモートデータフィールドのネットワークアドレスを10進の“.”(ドット表記)で指定してください。
- R_NMASK1 : リモートデータフィールドのネットマスクを10進の“.”(ドット表記)で指定してください。
- R_GTWAYA1 : 自ネットワークに接続されたゲートウェイ(ルータ)のIPアドレスを10進の“.”(ドット表記)で指定してください。
- R_METRIC1 : リモートデータフィールドへデータを送るためのメトリック数(ホップ数)を指定してください。
- DTSCYC : メッセージ送信サービス周期は、NXACPのシステムタスクがメッセージサービスをする周期です。下記に送信タスクの処理概要を示します。
- SBUFCNT,
RBUFCNT : 自ノード内データフィールドでRBUFCNT=0の指定はできません。リモートデータフィールドへの送信は、該当ローカルデータフィールドのSBUFCNTの定義が必要となります。
- ALSCYC,
ALTOUT : この2つのパラメータは、“ALSCYC < ALTOUT”の関係になるように設定してください。目安としては、ALSCYC値に対するALTOUT値は3~4倍となるようにしてください。

[送信処理]

送信処理を以下に示します。NXACP送信タスクは、ローカルDF単位に1タスク対応し、DTSCYC時間の周期で送信キューに接続されているデータを処理します。リモートDFへの送信メッセージもリモートDFが属するローカルDFの(DFTYPE)送信周期により送信します。

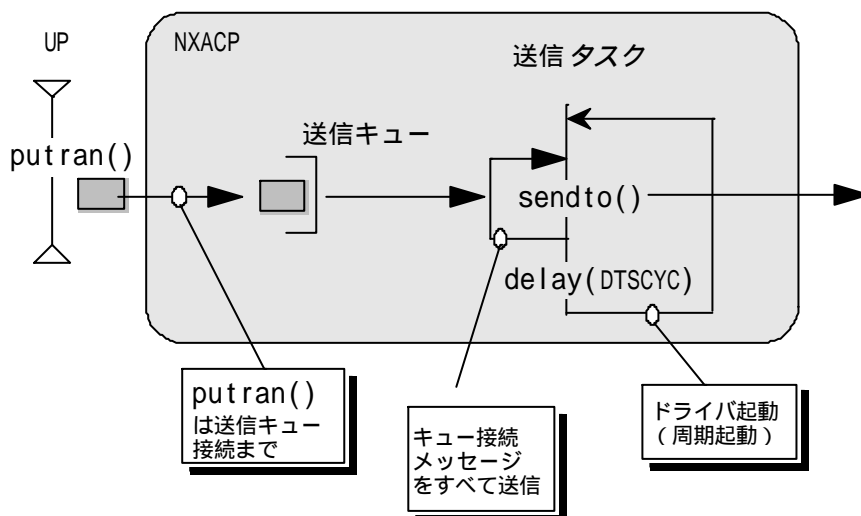


図 2 - 37 送信周期

< ローカルデータフィールド定義例 >

DFTYPE	0
NODENAME	OOMIKA
NODENO	3
NODEMODE	0
SSPORT	1025
ALPORT	600
ALSCYC	10
ALTOUT	30
DTSCYC	100
MINSNMCG	1
MAXSNMCG	4
SNDMCGCNT	4
MINRCVMCG	5
MAXRCVMCG	8
RCVMCGCNT	4
MINUSETCD	1
MAXUSETCD	64
USETCDCNT	32
SBUFCNT	32
RBUFCNT	64
MINNODENO	1
MAXNODENO	64

< 自ノード内通信データフィールド定義例 >

DFTYPE	0
MINUSETCD	1
MAXUSETCD	64
USETCDCNT	32
RBUFCNT	48

< リモートデータフィールド定義例 >

DFTYPE	1
NODEMODE	0
MINSNMCG	1
MAXSNMCG	4
SNDMCGCNT	4
MINUSETCD	1
MAXUSETCD	64
USETCDCNT	32
MINNODENO	1
MAXNODENO	64
R_INA1	192.168.1.0
R_NMASK1	255.255.255.0
R_GTWAYA1	192.168.0.2
R_METRIC1	1

2 機能ガイド

(c) マルチキャストグループ定義ファイル (mcgN.txt) 設定情報

このファイルには、以下の情報を設定してください。

右端の I / L / R は、データフィールドのタイプ (I : 自ノード内, L : ローカル, R : リモート) と指定の有無 (: 指定要, × : 不要 (指定無効)、 : ローカルDFの定義に従う) を示しています。

個々の設定が不要の場合は、“未使用時”の欄に示す数値を設定してください。また、“未使用時”の欄が“-”の項目については、最適な値を設定してください。

認識子	内容	指定できる範囲	未使用時	L	R	I
SMGN	送信マルチキャストグループ番号	1 ~ 255(*1)	-			×
RMGN	受信マルチキャストグループ番号	1 ~ 255(*1)	-		×	×
OPORT	オンライン用ポート番号(*2)	1 ~ 65535	0			×
TPORT	テスト用ポート番号(*2)	1 ~ 65535	0			×
TMODE	メッセージ受信モード 0 : オンライン、1 : テスト	0 ~ 1	-		×	×

(*1) 送受信マルチキャストグループの登録個数は、dfN.txtに登録したSNDMCGCNT、RCVMCGCNTまでが有効です。

(*2) 送信時は宛先送信ポート番号で、受信時はbind()するポート番号です。
オンライン用/テスト用のポート番号は、送受信別に定義してください。

< 定義例 >

#SND	MGN	OPORT	TPORT	
SMGN	1	55001	57001	
SMGN	2	55002	57002	
SMGN	3	55003	57003	
SMGN	4	55004	57004	
#RCV	MGN	OPORT	TPORT	TMODE
RMGN	5	55005	57005	0
RMGN	6	55006	57006	0
RMGN	7	55007	57007	0
RMGN	8	55008	57008	0

オンライン、またはテストのみ
定義したい場合は、未使用の
モードに対するポート番号は0
を指定してください。

【留意事項】

- ・ df N.txtのSSPORT, ALPORT で定義したポート番号をmcgN.txtに定義しないでください。
- ・ NXDlinkと通信する場合、NXDlinkのdf fieldN定義情報内RCVLEVELは“1”を指定してください。
- ・ テストモードのポートを使用して、NXDlink (WS、PC、制御サーバ用NX通信パッケージ) と通信する場合、NXDlinkのdf fieldN定義情報内LNSYSTYPEは“0”を指定してください。
- ・ ポート番号は、サブシステムによりリザーブされている場合がありますので、各サブシステムのマニュアルを参照してください。

(d) 受信トランザクションコード定義ファイル (rcvtcdN.txt) 設定情報

このファイルには、以下の情報を設定してください。

右端の I / L / R は、データフィールドのタイプ (I : 自ノード内, L : ローカル, R : リモート) と指定の有無 (: 指定要, x : 不要 (指定無効), : ローカル DF の定義に従う) を示しています。

個々の設定が不要の場合は、“未使用時”の欄に示す数値を設定してください。また、“未使用時”の欄が“-”の項目については、最適な値を設定してください。

認識子	内容	指定できる範囲	未使用時	L	R	I
TCD	受信トランザクションコード番号	1 ~ 59999(*1)	-		x	
CNT	最大受信バッファケース	1 ~ 1024(*2)	-		x	
TN	受信タスク番号	1 ~ 128(*3)	-		x	
QUTYPE	起動タイプ	0 ~ 5(*4)	-		x	
QUPARA	起動パラメータ	表2 - 15 参照	-		x	
NCBUFSZ	常駐バッファサイズ	256 ~ 16384	0	x	x	

(*1) TCDの登録個数は、dfN.txtに登録したUSETCDCNTまでです。また、TCDの指定範囲は、dfN.txtに登録したMINUSETCD ~ MAXUSETCDまでです。

(*2) 最大バッファケースは、dfN.txtに登録したRBUFCNTまでです。0指定時はdfN.txtのRBUFCNTの値となります。

(*3) 1つのTCDに対して登録できるタスク数は、最大16タスクです。

(*4) 起動タイプおよびパラメータについては表2 - 15を参照してください。

表2 - 15 起動タイプとパラメータ

起動タイプ	パラメータ1	パラメータ2	パラメータ3	パラメータ4
0 : イベント起動	0 固定	256 ~ 16384(*a)	-	-
1 : 時間起動	1 ~ 86400000(*b)	-	-	-
2 : 周期起動	1 ~ 86400000(*b)	1 ~ 86400000(*c)	-	-
3 : 時刻起動	1900 ~ 2199(*d)	1 ~ 12(*d)	1 ~ 31(*d)	0 ~ 86399(*d)
4 : 時間、周期起動 キャンセル	0 固定	-	-	-
5 : 時刻起動 キャンセル	0 固定	-	-	-

(*a) 常駐バッファサイズは、起動タイプ = イベント起動かつ、nxacp.conf.txt内NCBUFG = 1のときのみ指定できます。

(*b) 該当タスクを起動するまでの時間をmsにて指定してください。

(*c) 該当タスクを起動する周期時間をmsにて指定してください。

(*d) 起動する年、月、日、時間(秒)を指定してください。

2 機能ガイド

< ローカルデータフィールド定義例 >

#RCV	TCD	CNT	TN1-TN16	QUTYPE:QUPARA,NCBUFSZ
TCD	1	16	2 3	0:0
TCD	4	0	4	1:300
TCD	5	0	5	2:100:300
TCD	6	0	6	3:1998:9:6:10
TCD	7	0	4 5	4:0
TCD	8	0	6	5:0

< 自ノード内通信データフィールド定義例 >

#RCV	TCD	CNT	TN1-TN16	QUTYPE:QUPARA,NCBUFSZ
TCD	1	0	2 3	0:0
TCD	4	0	4	0:0,1024

< パラメータ指定時の注意事項 >

受信タスクに対する起動パラメータを指定する場合には、通常コロン(:)を使用してパラメータを区切りますが、常駐バッファを使用する場合には、カンマ(,)を使用してパラメータを区切ってください。

(e) 送信トランザクションコード定義ファイル(sndtcdN.txt) 設定情報

このファイルには、以下の情報を設定してください。

右端の I / L / R は、データフィールドのタイプ(I : 自ノード内, L : ローカル, R : リモート)と指定の有無(: 指定要, × : 不要(指定無効)、 : ローカルDFの定義に従う)を示しています。

個々の設定が不要の場合は、“未使用時”の欄に示す数値を設定してください。また、“未使用時”の欄が“-”の項目については、最適な値を設定してください。

認識子	内容	指定できる範囲	未使用時	L	R	I
TCD	送信トランザクションコード番号	1 ~ 59999(*1)	-			
TN	送信タスク番号	1 ~ 128(*2)	-			
MCG	送信マルチキャストグループ番号	1 ~ 255(*3)	-			×
SNDMODE	送信モード 0 : 自ノード内 1 : 自ノード外	0 ~ 1	-		×	×

(*1) TCDの登録個数は、dfN.txtに登録したUSETCDCNTまでです。また、TCDの指定範囲は、dfN.txtに登録したMINUSETCD~MAXUSETCDまでです。

(*2) 1つのTCDに対して登録できるタスク数は、最大16タスクです。

(*3) 送信マルチキャストグループには、mcgN.txtにて定義したマルチキャストグループを指定してください。また、指定できる最大マルチキャストグループは6つです。

< ローカル/リモートデータフィールド定義例 >

#SND	TCD	TN1-TN16	MCG1-MCG6	SNDMODE
TCD	10	10	1:0	1
TCD	11	11 12	2:3	1

< 自ノード内通信データフィールド定義例 >

#SND	TCD	TN1-TN16	MCG1-MCG6	SNDMODE
TCD	10	10		
TCD	11	11 12		

(2) 構築情報のコンパイル

サイト単位の各定義情報は、以下のコマンドによりコンパイルしてください。

なお、下記コマンドは、構築情報変更時も必ず実行してください。

```
# conf nxs サイト名称 [-i IRGLBNO] [-t TASKNO] <ret>
```

オプション機能を以下に示します。

- i IRGLBNO : ローディングするIRGLB番号を指定します。このオプション未指定時は、IRGLB番号1からローディングします。
- t TASKNO : insnxsコマンドにてtオプションを指定したときには、このオプションも指定する必要があります。タスク番号には、同じ番号を指定してください。
(insnxsコマンドにてtオプションを未指定時は、このオプションを指定しないでください。)

< エラー表示とその内容 >

パラメータ数誤り : 以下のエラーメッセージを表示します。

```
parameter number out of range
usage:conf nxs site
```

なお、confnxsコマンド実行中にコンパイラなどのエラーメッセージが出力された場合は、構築情報を見直してください。

< 実行中の表示とその内容 >

コマンドが正常に動作している場合、このコマンドは以下のメッセージを表示します。

```
*** NX/ACP-S10 GENERATION START(サイト名称) ***
|
*** NX/ACP-S10 GENERATION END ***
```


2 機能ガイド

(3) 構築情報のローディング

confnxs コマンドにて出力された情報は、tblldnxsコマンドにて指定サイトにローディングします。

```
# tblldnxs サイト名称 <ret>
```

このコマンドにより、構築情報をサイト名称として指定したS10のメモリーイメージファイルにローディングします。S10実機へのローディングは、RPDPのリモートローディングコマンド (srpl) を使用して一括ローディングしてください。

<エラー表示とその内容>

パラメータ数誤り：以下のエラーメッセージを表示します。

```
parameter number out of range  
usage:tblldnxs site
```

実行中のエラー：RPDP関連のエラーを表示した場合、「ソフトウェアマニュアル オペレーション RPDP/S10 For Windows®」(マニュアル番号 SAJ-3-133)を参照してください。

<実行中の表示とその内容>

コマンドが正常に動作している場合、このコマンドは以下のメッセージを表示します。

```
*** NX/ACP-S10 TABLE LOAD START(サイト名称) ***
```

```
|
```

```
*** NX/ACP-S10 TABLE LOAD END ***
```

← 実行コマンドを表示します。

2.6.5 容量見積り

NXACPでは、各種最大値、テーブル数などを様々なシステム形態に適合させるためにシステム構成情報をパラメータ化しています。パラメータの指定にあたって、NXACPを構築するために必要なメモリ容量の算出方式を示します。なお、以下に示す認識子は、構築情報の認識子です。

表2 - 16 容量算出シート

エリア名称	1 ケースバイト数 (対応構築ファイル名)	ケース数 (対応構築ファイル名)	合計(*2) (バイト)
共通固定部			15,008
自 ノ ド 内	受信バッファ	1476	RBUFCNT(df0)
	常駐バッファ	4+ 指定バッファサイズ (rcvtcd0)	指定バッファ数(rcvtcd0)
	T C D 管理部	32	1
		96	USETCDCNT(df0)
口 カ ル D F	送信バッファ	1476	SBUFCNT(dfN)
	受信バッファ	1476	RBUFCNT(dfN)
	T C D 管理部	32	1
		96	USETCDCNT(dfN)
	送信M C G 管理部	16	MAXSNMCG(dfN) - MINSNMCG(dfN) + 1
	受信M C G 管理部	256	1
		16	RCVMCGCNT(dfN)
	ノード管理部	32	MAXNODENO(dfN) - MINNODENO(dfN) + 1
16		RCVMCGCNT(dfN) × 定義ノード数(*1)	
リ モ ト D F	T C D 管理部	32	1
		96	USETCDCNT(dfN)
	送信M C G 管理部	16	MAXSNMCG(dfN) - MINSNMCG(dfN) + 1
	ノード管理部	32	MAXNODENO(dfN) - MINNODENO(dfN) + 1
		16	RCVMCGCNT(*3) × 定義ノード数(*1)
トレース部	32	512	
メッセージトレースバッファ	5456	1	
メッセージトレース退避バッファ	5456	1	
合計			

(*1) 定義ノード数とは、MAXNODENO(dfN) - MINNODENO(dfN) + 1です。

(*2) 合計は、1 ケースバイト数 × ケース数の値です。

(*3) dfN内DFTYPEで指定したソースDFの値です。

3 マクロ仕様

3 マクロ仕様

3.1 総説

3.1.1 マクロの種類と一覧

NXACPの提供するマクロは、IRSUB形態ですので、ユーザタスクにてスタックサイズを確保してください。また、リターンコードは、数値を直接返します。

NXACPのマクロ一覧を表3 - 1に示します。

表3 - 1 マクロ一覧

(1) マルチキャスト通信マクロ

マクロ名称	機能概要	スタックサイズ(バイト)
putran	マルチキャストメッセージ送信	768
getran	マルチキャストメッセージ受信	512
getmsg	常駐バッファからのメッセージ取得	512

(2) 運用マクロ

マクロ名称	機能概要	スタックサイズ(バイト)
acpinit	NXACPの起動	1280
acpquit	NXACPの停止	1280

(3) タイマ起動パラメータ変更マクロ

マクロ名称	機能概要	スタックサイズ(バイト)
sdtimer	時間起動パラメータの変更	512
stimer	周期起動パラメータの変更	512
swake	時刻起動パラメータの変更	512

3. 2 マルチキャスト通信マクロ

3. 2. 1 putran

【名前】

putran - 1メッセージの送信

【形式】

```
long putran (dfn, tcd, msg, msglen)
```

```
short *dfn;
```

```
short *tcd;
```

```
char *msg;
```

```
long *msglen;
```

【パラメータ】

dfn[in] 送信先データフィールド番号を指定します。

tcd[in] 送信するトランザクションコードを指定します。

msg[in] 送信すべきユーザデータの先頭アドレスを指定します。

msglen[in] 送信すべきユーザデータのサイズ(バイト単位)を指定します。

【機能説明】

putran()は、マルチキャスト方向のみをサポートし、1回の発行により1つのメッセージを1つのデータフィールド内にブロードキャスト送信します。なお、ブロードキャストは、ユーザが構築情報で指定したマルチキャストグループ分(回数)行います。

1回の要求で送信できるユーザデータサイズは0~16Kバイトです。

自ノード内のみに、データを送信したい場合にはDF番号=0を指定して、putran()を発行します。そのときは、DF=0用受信バッファを使用して送信します。なお、常駐バッファがある場合に、常駐バッファへのメッセージコピーは、putran()にて行います。

putran()は、NXACPの内部キューに送信データを格納した時点でリターンします。

putran()は、ノンブロック送信のみサポートします。キュー満杯などの原因によりキュー確保が失敗した場合には、キュー確保リトライはせず、そのままエラーリターンします。

3 マクロ仕様

【戻り値】

putran 処理が正常終了すると、値 0 を返します。
異常終了の場合は、次の 0 の値を返します。

- 0 × 0 0 1 : NXACPが停止状態です。
- 0 × 0 0 3 : NXACPのテーブルが未ロードです。
- 0 × 1 0 1 : tcdが指定できる範囲 (1 ~ 5 9 9 9 9) 外です。
- 0 × 1 0 2 : msglen が指定できる範囲 (0 ~ 1 6 3 8 4) 外です。
- 0 × 1 0 3 : dfnが指定できる範囲 (0 ~ 2 5 5) 外です。
- 0 × 1 0 4 : tcdが未定義です。
- 0 × 1 0 5 : dfnが未定義です。
- 0 × 1 0 7 : 送信マクロの発行権がありません。
- 0 × 2 0 1 : 送信バッファがオーバーフロー状態です。

【補足】

送信宛先として指定するデータフィールドの情報、マルチキャストグループの情報およびトランザクションコードの情報をあらかじめ構築時に設定してください。

putran()のエラーコードで返すのは、NXACP内部の送信キュー接続までの異常情報のみです。

NXACPは、「同じタスクレベルからの送信要求順に処理し、その順序を保証してネットワークに送出」します。もし、順序保証を厳密に行いたい場合には、putran()がリターンしたことを確認してから次のデータの送信要求putran()を発行してください。

3. 2. 2 getran

【名前】

getran - 1メッセージの受信

【形式】

```
long getran (dfn, tcd, msg, msglen)
```

```
short *dfn;
```

```
short *tcd;
```

```
char *msg;
```

```
long *msglen;
```

【パラメータ】

dfn[out] 受信データの送信先データフィールド番号が設定されます。

tcd[out] 受信データのトランザクションコードが設定されます。

msg[out] 受信データ格納エリアの先頭アドレスを指定します。

このエリアに受信したデータが格納されます。

msglen[out] 受信データのサイズ(バイト)が設定されます。

【機能説明】

getran()は、構築時に指定したTCD付きメッセージを1つ受信します。1回のgetran()で受信できるユーザデータサイズは、0～16Kバイトです。

ユーザタスクがどのメッセージを受信するかは構築時に指定しますので、getran()でその中の特定のメッセージを指定して受信することはできません。また、該当タスクはNXACPからのqueue()マクロ発行により起動します。そのときの起動要因(FACT)には16を固定で設定します。

ネットワークから受信した順にユーザタスクキューに接続し、先入後出(FIFO)でユーザタスクにサービスします。

3 マクロ仕様

【戻り値】

getran 処理が正常終了すると値 0 を返し、msglen に受信したデータサイズ (0 以上) を設定します。
異常終了の場合は、 0 の値を返します。

0 × 0 0 1 : NXACP が停止状態です。

0 × 0 0 3 : NXACP のテーブルが未ロードです。

0 × 0 5 0 : 該当タスクが受信するデータはありません。また、該当タスクに対する受信 TCD が未定義です。

【補足】

受信情報としてデータフィールドの情報、マルチキャストグループの情報およびトランザクションコードの情報をあらかじめ構築時に設定しておく必要があります。

受信データ格納エリアサイズと受信データサイズの比較は行っていないため、

「受信データ格納エリアサイズ < 受信データサイズ」時に受信データ格納エリアを破壊する可能性がありますので、十分な大きさの受信データ格納エリアを確保後、getran() を発行してください。

NXACP は getran() が発行されるまで受信データを保持しますので、ユーザは NXACP からのデータ受信による起動時は、必ず getran() を発行し受信データを取得してください。

3. 2. 3 getmsg

【名前】

getmsg - 常駐バッファからのメッセージの取得

【形式】

```
long getmsg (dfn, tcd, msg, msglen)
```

```
short *dfn;
```

```
short *tcd;
```

```
char *msg;
```

```
long *msglen;
```

【パラメータ】

dfn[in] 受信データの送信先データフィールド番号を指定します（0固定）。

tcd[in] 受信データのトランザクションコードを指定します。

msg[out] 受信データ格納エリアの先頭アドレスを指定します。

このエリアに受信したデータが格納されます。

msglen[out] 受信データのサイズ（バイト）が設定されます。

【機能説明】

getmsg()は、構築時に指定したTCD付きメッセージを1つ常駐バッファから取得します。

1回のgetmsg()で受信できるユーザデータサイズは、0～16Kバイトです。

3 マクロ仕様

【戻り値】

getmsg処理が正常終了すると値0を返し、msglenに受信したデータサイズ(0以上)を設定します。

異常終了の場合は、0の値を返します。

0 × 0 0 1 : NXACPが停止状態です。

0 × 0 0 3 : NXACPのテーブルが未ロードです。

0 × 0 0 5 : 指定されたTCDは常駐メッセージとして定義されていません。

0 × 1 0 1 : tcdが指定できる範囲(1 ~ 5 9 9 9 9)外です。

0 × 1 0 4 : tcdが未定義です。

0 × 1 0 5 : dfmが未定義です。

0 × 1 0 6 : dfmが指定できる範囲(0)外です。

【補足】

受信情報としてデータフィールドの情報、およびトランザクションコードの情報をあらかじめ構築時に設定しておく必要があります。

受信データ格納エリアサイズと受信データサイズの比較は行っていないため、

「受信データ格納エリアサイズ < 受信データサイズ」時に受信データ格納エリアを破壊する可能性がありますので、十分な大きさの受信データ格納エリアを確保後、getmsg()を発行してください。

常駐バッファにユーザが受信すべきデータがない(常駐メッセージの送信が未発行)場合には、msglen: - 1、リターンコード: 0でリターンします。

3.3 運用マクロ

3.3.1 acpinit

【名前】

acpinit - NXACPの立上げ

【形式】

long acpinit ()

【パラメータ】

なし。

【機能説明】

acpinit()は、NXACPの処理立上げ、および共通テーブルを初期化します。
ユーザは、システム立上げ時、最初にこのマクロを発行してください。

【戻り値】

acpinit()が正常終了すると、値0を返します。

異常終了の場合は、0の値を返します。

0 x 0 0 2 : NXACPはすでに動作中です。

0 x 0 0 3 : NXACPのテーブルが未ロードです。

0 x 0 0 4 : NXACPのタスクが未ロード、またはタスク起動が失敗しました。

0 x 3 0 1 : MAINモジュールの回線接続 (socket) が失敗しました。

0 x 3 0 2 : SUBモジュールの回線接続 (socket) が失敗しました。

0 x 3 1 1 : MAINモジュールに接続されている、イーサネットボードの型式はLWE550ではありません。

0 x 3 1 2 : SUBモジュールに接続されている、イーサネットボードの型式はLWE550ではありません。

0 x 3 2 1 : MAINモジュールの初期化が完了していません。

0 x 3 2 2 : SUBモジュールの初期化が完了していません。

0 x 3 3 1 : MAINモジュールのネットワーク情報取得 (getconfi g) が失敗しました。

0 x 3 3 2 : SUBモジュールのネットワーク情報取得 (getconfi g) が失敗しました。

0 x 3 4 1 : MAINモジュールに対するルーティング情報登録 (route_add) が失敗しました。

0 x 3 4 2 : SUBモジュールに対するルーティング情報登録 (route_add) が失敗しました。

【補足】

NXACPの構築情報 (nxacp.conf.txt) で「RLEASMODOE」が1のときは、acpinit()にてユーザタスクの起動 (releas) をします。

3 マクロ仕様

3.3.2 acpquit

【名前】

acpquit - NXACPの停止

【形式】

long acpquit()

【パラメータ】

なし。

【機能説明】

acpquit()は、NXACPを停止状態にし、SHUTDOWNモードの生存信号を送信して他ノードに状態変化を通知します。

【戻り値】

acpquit()は正常終了時、値0を返します。

異常終了の場合は、次の値を返します。

0 x 0 0 1 : NXACPはすでに停止状態です。

0 x 0 0 3 : 構築情報のテーブルが未ロードです。

【補足】

acpquit()処理中、ポートなどのCLOSE処理などに異常を検知した場合でも異常終了しません。

SHUTDOWNモードの生存信号を一定時間に3回送信しますので、終了するまでに5 ~ 10秒程度かかります。

3. 4 タイマ起動パラメータ変更マクロ

3. 4. 1 sdtimer

【名前】

sdtimer - 時間起動に対する起動パラメータの変更

【形式】

```
long sdtimer(dfn, tcd, t)
```

```
short *dfn;
```

```
short *tcd;
```

```
long *t;
```

【パラメータ】

dfn[in] 起動パラメータを変更するデータフィールド番号を指定します。

tcd[in] 起動パラメータを変更するトランザクションコードを指定します。

t[in] 変更する時間 (1 ~ 8 6 4 0 0 0 0) をmsで指定します。

【機能説明】

sdtimer()は、構築時に指定したTCDごとの時間起動に対する起動パラメータを変更します。

時間起動とは、時間起動用TCD (ユーザが構築時に定義)を受信時に、ユーザタスクを起動パラメータ (t) で指定された時間経過後、1回だけ起動するものです。

【戻り値】

sdtimer()が正常終了すると、値0を返します。

異常終了の場合は、0の値を返します。

0 x 0 0 1 : NXACPが停止状態です。

0 x 0 0 3 : NXACPのテーブルが未ロードです。

0 x 1 0 1 : tcdが指定できる範囲 (1 ~ 5 9 9 9) 外です。

0 x 1 0 3 : dfnが指定できる範囲 (0 ~ 2 5 5) 外です。

0 x 1 0 4 : tcdが未定義です。

0 x 1 0 5 : dfnが未定義です。

0 x 1 0 8 : tcdの起動タイプが不一致です。

0 x 1 0 9 : tが指定できる範囲 (1 ~ 8 6 4 0 0 0 0) 外です。

【補足】

データフィールドの情報、およびトランザクションコードの情報をあらかじめ構築時に設定しておく必要があります。

すでに起動され、実行待ち状態にあるタスクに設定された時間は変更されません。

NXACPテーブルをローディングした場合、sdtimer()にて設定された値は無効となります (変更前の値に戻ります)。

3 マクロ仕様

3.4.2 stimer

【名前】

stimer - 周期起動に対する起動パラメータの変更

【形式】

```
long stimer(dfn, tcd, t, cyt)
```

```
short *dfn;
```

```
short *tcd;
```

```
long *t;
```

```
long *cyt;
```

【パラメータ】

dfn[in] 起動パラメータを変更するデータフィールド番号を指定します。

tcd[in] 起動パラメータを変更するトランザクションコードを指定します。

t[in] 変更する時間 (1 ~ 8 6 4 0 0 0 0 0) をmsで指定します。

cyt[in] 起動する周期時間 (1 ~ 8 6 4 0 0 0 0 0) をmsで指定します。

【機能説明】

stimer()は、構築時に指定したTCDごとの周期起動に対する起動パラメータを変更します。

周期起動とは、周期起動用TCD (ユーザが構築時に定義)を受信時に、ユーザタスクを起動パラメータ (t) で指定された時間経過後、周期的 (cyt) に起動するものです。

【戻り値】

stimer()が正常終了すると、値0を返します。

異常終了の場合は、0の値を返します。

0 x 0 0 1 : NXACPが停止状態です。

0 x 0 0 3 : NXACPのテーブルが未ロードです。

0 x 1 0 1 : tcdが指定できる範囲 (1 ~ 5 9 9 9 9) 外です。

0 x 1 0 3 : dfnが指定できる範囲 (0 ~ 2 5 5) 外です。

0 x 1 0 4 : tcdが未定義です。

0 x 1 0 5 : dfnが未定義です。

0 x 1 0 8 : tcdの起動タイプが不一致です。

0 x 1 0 9 : t、またはcytが指定できる範囲 (1 ~ 8 6 4 0 0 0 0 0) 外です。

【補足】

データフィールドの情報、およびトランザクションコードの情報をあらかじめ構築時に設定しておく必要があります。

すでに起動され、実行待ち状態にあるタスクに設定された時間は変更されません。

NXACPテーブルをローディングした場合、stimer()にて設定された値は無効となります (変更前の値に戻ります)。

3.4.3 `swake`

【名前】

`swake` - 時刻起動に対する起動パラメータの変更

【形式】

long `swake(dfn, tcd, sec, day, month, year)`

short *dfn;
 short *tcd;
 long *sec;
 short *day;
 short *month;
 short *year;

【パラメータ】

`dfn[in]` 起動パラメータを変更するデータフィールド番号を指定します。
`tcd[in]` 起動パラメータを変更するトランザクションコードを指定します。
`sec[in]` 起動する時間 (1 ~ 8 6 4 0 0 0 0 0) を `ms` で指定します。
 (0 時 0 分 0 秒からの相対)
`day[in]` 起動する日付 (1 ~ 3 1) を指定します。
`month[in]` 起動する月 (1 ~ 1 2) を指定します。
`year[in]` 起動する年 (1 9 0 0 ~ 2 1 9 9) を指定します。

【機能説明】

`swake()` は、構築時に指定した TCD ごとの時刻起動に対する起動パラメータを変更します。
 時刻起動とは、時刻起動用 TCD (ユーザが構築時に定義) を受信時に、ユーザタスクを起動パラメータ (`year, month, day, src`) で指定された時刻に 1 度だけ起動するものです。

【戻り値】

`swake()` が正常終了すると、値 0 を返します。
 異常終了の場合は、 0 の値を返します。
 0 x 0 0 1 : NXACP が停止状態です。
 0 x 0 0 3 : NXACP のテーブルが未ロードです。
 0 x 1 0 1 : `tcd` が指定できる範囲 (1 ~ 5 9 9 9 9) 外です。
 0 x 1 0 3 : `dfn` が指定できる範囲 (0 ~ 2 5 5) 外です。
 0 x 1 0 4 : `tcd` が未定義です。
 0 x 1 0 5 : `dfn` が未定義です。
 0 x 1 0 8 : `tcd` の起動タイプが不一致です。
 0 x 1 0 9 : `year, month, day, sec` のいずれかが指定できる範囲外です。

【補足】

データフィールドの情報、およびトランザクションコードの情報をあらかじめ構築時に設定しておく必要があります。

すでに起動され、実行待ち状態にあるタスクに設定された時間は変更されません。

NXACP テーブルをローディングした場合、`swake()` にて設定された値は無効となります (変更前の値に戻ります) 。

付 録

付録 A リターンコード詳細

(1) マクロリターンコード

コード	内 容	ユーザ対処方法
0x0001	NXACP停止中	NXACPを初期化してください。
0x0002	NXACP稼働中	NXACPIはすでに初期化済みのため、初期化要求は不要です。
0x0003	NXACPテーブル未ロード	構築情報をローディングしてください。
0x0004	NXACPタスク未ロード	NXACPプログラムをローディングしてください。
0x0005	常駐メッセージとして未定義	パラメータに指定したTCD番号と構築情報を見直してください。
0x0050	受信メッセージなし	特になし。
0x0101	TCD設定可能範囲外	パラメータに指定したTCD番号を見直してください。
0x0102	メッセージ長設定可能範囲外	パラメータに指定したメッセージ長を見直してください。
0x0103	指定DF番号設定可能範囲外	パラメータに指定したDF番号を見直してください。
0x0104	指定TCD未定義	パラメータに指定したTCD番号と構築情報が合っているか見直してください。
0x0105	指定DF番号未定義	パラメータに指定したDF番号と構築情報が合っているか見直してください。
0x0106	指定DF番号が0以外	パラメータに指定できるDF番号は0のみです。
0x0107	送信マクロ発行権なし	構築情報に該当タスクが送信するTCD番号を定義してください。
0x0108	指定TCD起動タイプ不一致	構築情報で指定したTCD番号と起動タイプを見直してください。
0x0109	指定TCD起動パラメータ不一致	パラメータで指定された時間/周期/時刻の値を見直してください。
0x0201	送受信バッファオーバーフロー	リトライしてください。
0x0301	回線 (socket) 接続失敗 (MAIN)	システム管理者に通知してください。
0x0302	回線 (socket) 接続失敗 (SUB)	システム管理者に通知してください。
0x0311	ハードウェア異常 (MAIN)	ハードウェアの型式 (LWE550のみサポート) を確認してください。
0x0312	ハードウェア異常 (SUB)	ハードウェアの型式 (LWE550のみサポート) を確認してください。
0x0321	初期化未完了 (MAIN)	リトライしてください。リトライしても同様のときは、システム管理者に通知してください。
0x0322	初期化未完了 (SUB)	リトライしてください。リトライしても同様のときは、システム管理者に通知してください。
0x0331	ネットワーク情報取得 (getconfi g) 失敗 (MAIN)	システム管理者に通知してください。
0x0332	ネットワーク情報取得 (getconfi g) 失敗 (SUB)	システム管理者に通知してください。
0x0341	ルーティング情報登録 (route_add) 失敗 (MAIN)	システム管理者に通知してください。
0x0342	ルーティング情報登録 (route_add) 失敗 (SUB)	システム管理者に通知してください。

<リターンコードとマクロの対応>

コード	acpinit	acpquit	putran	getran	getmsg	sdtimer	stimer	swake
0x0001	-							
0x0002		-	-	-	-	-	-	-
0x0003								
0x0004		-	-	-	-	-	-	-
0x0005	-	-	-	-		-	-	-
0x0050	-	-	-		-	-	-	-
0x0101	-	-		-				
0x0102	-	-		-	-	-	-	-
0x0103	-	-		-	-			
0x0104	-	-		-				
0x0105	-	-		-				
0x0106	-	-	-	-		-	-	-
0x0107	-	-		-	-	-	-	-
0x0108	-	-	-	-	-			
0x0109	-	-	-	-	-			
0x0201	-	-		-	-	-	-	-
0x0301		-	-	-	-	-	-	-
0x0302		-	-	-	-	-	-	-
0x0311		-	-	-	-	-	-	-
0x0312		-	-	-	-	-	-	-
0x0321		-	-	-	-	-	-	-
0x0322		-	-	-	-	-	-	-
0x0331		-	-	-	-	-	-	-
0x0332		-	-	-	-	-	-	-
0x0341		-	-	-	-	-	-	-
0x0342		-	-	-	-	-	-	-

: エラーリターンあり - : エラーリターンなし

(2) 構築コマンドエラーコード一覧

詳細を以下の表に示します。

< 共通エラー >

コード	エラーメッセージ	エラー内容	ユーザの対応
1100	error EEE site XXXX not found	サイト (XXXX) が存在しません。	dfnxsコマンドを起動してサイト構築環境を作成してください。
1101	error EEE environment variable XXXX undefiend	環境変数 (XXXX) が設定されていません。	環境変数を設定してください。
1102	error EEE file XXXX not open(YY)	ファイル (XXXX) が存在しません。	システム管理者に通知してください。
1103	error EEE NXACP Home path not found	NXACPのホームパスが見つかりませんでした。	システム管理者に通知してください。
1104	error EEE directory(XXXX) not found	ディレクトリ (XXXX) が存在しません。	システム管理者に通知してください。
1105	error EEE file XXXX not read	ファイル (XXXX) に対してデータの読み込みができません。	システム管理者に通知してください。
1106	error EEE file XXXX not write	ファイル (XXXX) に対してデータの書き込みができません。	システム管理者に通知してください。

< 範囲、重複、未定義エラー >

コード	エラーメッセージ	エラー内容	ユーザの対応
1301	FFF, line LLL: error EEE Unexpected Symbol Value: (XXXX)	コメント以外の異常な値を検出しました。	異常な値をコメントにするか、削除してください。
1302	FFF, line LLL: error EEE Symbole Multiple Defined: (XXXX)	重複しているシンボルがあります。	重複しているシンボルを修正してください。
1303	FFF, line LLL: error EEE Value Multiple Defined: (XXXX)	重複している値があります。	重複している値を修正してください。
1304	FFF, line LLL: error EEE Undefined Symbol: (XXXX)	シンボルが未定義です。	シンボルと値を定義してください。
1305	FFF, line LLL: error EEE Request Symbol: (XXXX) Overcount	指定したシンボルの数が範囲を超えています。	指定したシンボルの定義数を見直してください。
1306	FFF, line LLL: error EEE Request Value: (XXXX) Overcount	指定した値の数が範囲を超えています。	値の設定数を見直してください。
1307	FFF, line LLL: error EEE Symbol: (XXXX) No search value	シンボルに対する値が未定義です。	値を見直してください。
1308	FFF, line LLL: error EEE Symbol value illegal or out of range: (XXXX)	値が異常、または範囲外です。	値を見直してください。
1309	FFF, line LLL: error EEE Symbol: (XXXX) value illegal	値が異常です。	設定値を見直してください。
1310	FFF, line LLL: error EEE Value: (XXXX) illegal format	値のフォーマットが合っていません。	値のフォーマットを見直してください。
1311	FFF, line LLL: error EEE define be not enough	必須項目の定義情報がありません。	定義情報を追加してください。

未定義エラー時のline LLLは、ファイル検索の結果として最終行を表示します。

コード	エラーメッセージ	エラー内容	ユーザの対応
1330	FFF, line LLL: error EEE Specified source DFTYPE(XXXX) is undefined	リモートDFのシンボル (DFTYPE)において、ロー カルDFNoの定義がされませ んでした。	シンボル (DFTYPE) を見 直してください。
1331	FFF, line LLL: error EEE TRCMODE (XXXX) not rationality:	シンボル (TRCMODE) にお いて、非合理的な値を検出しま した。	シンボル (TRCMODE) を 見直してください。
1332	FFF, line LLL: error EEE MINUSETCD(XXXX),MAXUSETCD (XXXX) not rationality	シンボル (MINUSETCD) と (MAXUSETCD) において、 非合理的な値を検出しました。	シンボル (MINUSETCD) と (MAXUSETCD) を見直 してください。
1333	FFF, line LLL: error EEE DFTYPE(XXXX) not rationality	シンボル (DFTYPE) にお いて、非合理的な値を検出しま した。	シンボル (DFTYPE) を見 直してください。
1336	FFF, line LLL: error EEE SSPORT(XXXX),ALPORT(XXXX) not rationality	シンボル (SSPORT) と (ALPORT) において、非合 理的な値を検出しました。	シンボル (SSPORT) と (ALPORT) を見直してく ださい。
1337	FFF, line LLL: error EEE ALSCYC(XXXX), ALTOUT(XXXX) not rationality	シンボル (ALSCYC) と (ALTOUT) において、非合 理的な値を検出しました。	シンボル (ALSCYC) と (ALTOUT) を見直してく ださい。
1338	FFF, line LLL: error EEE LANMODE(XXXX) multidefine	シンボル (LANMODE) の 値が他のDF定義と重複し ています。	シンボル (LANMODE) を 見直してください。
1339	FFF, line LLL: error EEE MINSNDMCG(XXXX),MAXSNDMCG (XXXX) not rationality	シンボル (MINSNDMCG) と (MAXSNDMCG) におい て、非合理的な値を検出しま した。	シンボル (MINSNDMCG) と (MAXSNDMCG) を見直 してください。
1340	FFF, line LLL: error EEE MINSNDMCG(XXXX),MAXSNDMCG (XXXX),SNDMCGCNT(XXXX) Resource data unmatch	シンボル (MINSNDMCG) と (MAXSNDMCG) と (SNDMCGCNT) のいづれ かが非合理的な値となってい ます。	シンボル (MINSNDMCG) と (MAXSNDMCG) と (SNDMCGCNT) を見直 してください。
1341	FFF, line LLL: error EEE MINRCVMCG(XXXX),MAXRCVMCG (XXXX) not rationality	シンボル (MINRCVMCG) と (MAXRCVMCG) におい て、非合理的な値を検出しま した。	シンボル (MINRCVMCG) と (MAXRCVMCG) を見直 してください。

コード	エラーメッセージ	エラー内容	ユーザの対応
1342	FFF, line LLL: error EEE MINRCVMCG(XXXX), MAXRCVMCG (XXXX), RCVMCGCNT(XXXX) Resource data unmatched	シンボル (MINRCVMCG) と (MAXRCVMCG) と (RCVMCGCNT) のいずれかが非合理的な値となっています。	シンボル (MINRCVMCG) と (MAXRCVMCG) と (RCVMCGCNT) を見直してください。
1343	FFF, line LLL: error EEE SBUFSZ(XXXX), SBUFCNT(XXXX) not rationality	シンボル (SBUFSZ) に対して (SBUFCNT) が非合理的な値です。	シンボル (SBUFSZ) と (SBUFCNT) を見直してください。
1344	FFF, line LLL: error EEE RBUFSZ(XXXX), RBUFCNT(XXXX) not rationality	シンボル (RBUFSZ) に対して (RBUFCNT) が非合理的な値です。	シンボル (RBUFSZ) と (RBUFCNT) を見直してください。
1345	FFF, line LLL: error EEE MINNODENO(XXXX), MAXNODENO (XXXX) not rationality	シンボル (MINNODENO) と (MAXNODENO) において、非合理的な値を検出しました。	シンボル (MINNODENO) と (MAXNODENO) を見直してください。
1346	FFF, line LLL: error EEE SSPORT(XXXX) to SMGN(XXXX) : OPORT(XXXX) and TPORT(XXXX) not rationality	シンボル (SSPORT) に対して SMGN の (OPORT) と (TPORT) が非合理的な値です。	シンボル (SSPORT) に対して SMGN の (OPORT) と (TPORT) を見直してください。
1347	FFF, line LLL: error EEE ALPORT(XXXX) to SMGN(XXXX) : OPORT(XXXX) and TPORT(XXXX) not rationality	シンボル (ALPORT) に対して SMGN の (OPORT) と (TPORT) が非合理的な値です。	シンボル (ALPORT) に対して SMGN の (OPORT) と (TPORT) を見直してください。
1348	FFF, line LLL: error EEE SSPORT(XXXX) to RMGN(XXXX) : OPORT(XXXX) and TPORT(XXXX) not rationality	シンボル (SSPORT) に対して RMGN の (OPORT) と (TPORT) が非合理的な値です。	シンボル (SSPORT) に対して RMGN の (OPORT) と (TPORT) を見直してください。
1349	FFF, line LLL: error EEE ALPORT(XXXX) to RMGN(XXXX) : OPORT(XXXX) and TPORT(XXXX) not rationality	シンボル (ALPORT) に対して RMGN の (OPORT) と (TPORT) が非合理的な値です。	シンボル (ALPORT) に対して RMGN の (OPORT) と (TPORT) を見直してください。
1350	FFF, line LLL: error EEE MCG(XXXX): MINSNDMCG(XXXX)- MAXSNDMCG(XXXX) out of range	MCG は (MINSNDMCG) から (MAXSNDMCG) の範囲外です。	範囲内に納めるよう MCG を見直してください。
1351	FFF, line LLL: error EEE MCG(XXXX): SMGN out of range	MCG は (SMGN) の指定範囲外です。	範囲内に納めるよう MCG を見直してください。

コード	エラーメッセージ	エラー内容	ユーザの対応
1352	FFF, line LLL: error EEE NODEMODE(XXXX), OPORT(XXXX) Resource data unmatched	受信ポートモードが、オンラインポートのみ有効にも関わらず、オンラインポートに設定がありません。	オンラインポート (OPORT) を 0 以外に設定してください。
1353	FFF, line LLL: error EEE NODEMODE(XXXX), TPORT(XXXX) Resource data unmatched	受信ポートモードが、テストポート有効にも関わらず、テストポートに設定がありません。	テストポート (TPORT) を 0 以外に設定してください。
1354	FFF, line LLL: error EEE NODEMODE(XXXX), MODE(XXXX) OPORT(XXXX) Resource data unmatched	(NODEMODE) に対して、(MODE) と (OPORT) が非合理的な値です。	(MODE) と (OPORT) を見直してください。
1355	FFF, line LLL: error EEE MGN(XXXX) S:OPORT(XXXX) R:OPORT(XXXX) Resource data unmatched	マルチキャストグループ (MGN) に対し送信オンラインポート (OPORT) と受信オンラインポート (OPORT) が一致しません。	マルチキャストグループ (MGN) に対し送信オンラインポート (OPORT) と受信オンラインポート (OPORT) を同じ値に変更してください。
1356	FFF, line LLL: error EEE MGN(XXXX) S:TPORT(XXXX) R:TPORT(XXXX) Resource data unmatched	マルチキャストグループ (MGN) に対し送信テストポート (TPORT) と受信テストポート (TPORT) が一致しません。	マルチキャストグループ (MGN) に対し送信テストポート (TPORT) と受信テストポート (TPORT) を同じ値に変更してください。
1357	FFF, line LLL: error EEE MINSNDMCG(XXXX), MAXSNDMCG (XXXX), SNDMCGCNT(XXXX), SBUFCNT(XXXX) Resource data unmatched	シンボル (MINSNDMCG), (MAXSNDMCG), (SNDMCGCNT), (SBUFCNT) が非合理的な値です。	シンボル (MINSNDMCG), (MAXSNDMCG), (SNDMCGCNT), (SBUFCNT) を見直してください。 (= 0、 0 の値に統一してください。)
1358	FFF, line LLL: error EEE MINRCVMCG(XXXX), MAXRCVMCG (XXXX), RCVCMCGCNT(XXXX), RBUFCNT(XXXX), MINNODENO (XXXX), MAXNODENO(XXXX) Resource data unmatched	シンボル (MINRCVMCG), (MAXRCVMCG), (RCVCMCGCNT), (RBUFCNT), (MINNODENO), (MAXNODENO) が非合理的な値です。	シンボル (MINRCVMCG), (MAXRCVMCG), (RCVCMCGCNT), (MINNODENO), (MAXNODENO) を見直してください。 (= 0、 0 の値に統一してください。)
1359	FFF, line LLL: error EEE TN:(XXXX):is already defined NXACP TASKNO(XXXX- XXXX)	設定したタスク番号は、登録するNXACPタスクの番号と重複しています。	タスク番号を登録するNXACPタスクの番号と重複しない値で設定してください。

<警告>

コード	エラーメッセージ	エラー内容	ユーザの対応
1360	FFF, line LLL: warning EEE Illegal Symbol: (XXXX)	不当なシンボルを検出しました。	削除するかコメントにしてください。
1361	FFF, line LLL: warning EEE not 4byte data align:(XXXX)	値が4バイトデータ境界になっていません。	値を4の倍数にしてください。

FFF : ファイル名称
 LLL : 障害発生行番号
 EEE : コード
 XXXX : 障害発生トリガ
 YY : エラー番号

付録 B 組み込みサブルーチンの登録

NXACPでは、組み込みサブルーチンとしてEXSとABSを使用します。

ユーザが前記の組み込みサブルーチンを使用しない場合は、NXACPプログラムローディング時に標準で組み込みサブルーチンを登録します（登録についてはinsnxsコマンドを参照してください）。

組み込みサブルーチンをユーザとNXACPで共有する場合には、ユーザが組み込みサブルーチンを登録してください。

ABS	EXS	組み込みサブルーチンの登録者
		ユーザ
	×	ユーザ (*1)
×		ユーザ (*2)
×	×	NXACP

：ユーザが組み込みサブルーチンを使用することを示します。
 ×：ユーザが組み込みサブルーチンを使用しないことを示します。

(*1) ユーザはNXACPが使用するEXSも登録してください。
 (*2) ユーザはNXACPが使用するABSも登録してください。

以下に組み込みサブルーチンの登録例を示します。

(1) EXS (タスクEXIT用組み込みサブルーチン)

(a) プログラム例

```
usr_exs( )
{
    nx_exs( );
    xxx( );
}
```

：メインルーチンのプログラム名称です（名称は任意です）。
 : NXACPのプログラム名称です（パラメータはありません）。
 : ユーザのサブプログラム名称です。

(b) ローディング

(a)で作成したプログラムの登録方法を示します。

```
$ mcc68k -c -S -Fsm プログラム名称<ret>
$ asm68k -l -f " case,-t" プログラム名称.lst プログラム名称.src<ret>
$ sload プログラム名称 -a エリア名称 -f コマンドファイル名称 -w スタックサイズ +U<ret>
$ sbuild プログラム名称 -p 4<ret>
```


(2) A B S (タスクABORT用組み込みサブルーチン)

(a) プログラム例

```
usr_abs( )
{
    nx_abs( );           : メインルーチンのプログラム名称です (名称は任意です)。
    xxx( );             : NXACPのプログラム名称です (パラメータはありません)。
                       : ユーザのサブプログラ名称です。
}
```

(b) ローディング

(a)で作成したプログラムの登録方法を示します。

```
$ mcc68k -c -S -Fsm プログラム名称<ret>
$ asm68k -l -f " case,-t" プログラム名称.lst プログラム名称.src<ret>
$ sload プログラム名称 -a エリア名称 -f コマンドファイル名称 -w スタックサイズ +U<ret>
$ sbuild プログラム名称 -p 5<ret>
```

付録C 異常NXヘッダ通知フォーマット

NXACPでは異常NXヘッダを受信した場合、該当ヘッダ情報をIRSUB番号38へ通知します。

ユーザは、該当情報をグローバルエリアなどに保存してください。

なお、IRSUB番号38はユーザが登録してください。

以下に通知フォーマット、エラー情報および登録方法を示します。

<通知フォーマット>

0	通 知 コ ー ド	通知コード	: 1 固定です。
+4	送 信 元 D F 番 号	送信元DF番号	: 該当データを送信したDF番号です。
+8	受 信 元 ポ ー ト 番 号	受信元ポート番号	: 該当データを受信したポート番号です。
+12	エ ラ ー コ ー ド	エラーコード	: 異常としたコードです。
+16	受 信 し た N X ヘ ッ ダ		
+80			

<エラーコード>

No.	エラーコード	エラー要因
1	1	未定義DFからデータを受信しました。
2	2	受信タスクがサービスしているDF番号と受信したデータ内DF番号が不一致です。
3	3	ヘッダタイプが不正です。
4	4	メッセージ送信制御情報が不正です。
5	5	受信データ長が不正です。(受信データ>受信バッファサイズ)
6	6	受信MCGが不正です。(NXヘッダ内情報とNXテーブル情報不一致)
7	7	受信MCGが未定義です。(将来用)
8	8	メッセージ送信通番が不正です。
9	9	メッセージモードが不正です。
10	10	ロングメッセージ組立後のメッセージサイズが不正です。
11	11	システムTCD受信時のカレントフラグメントブロック番号が不正です。
12	12	システムTCD受信時のトータルフラグメントブロック番号が不正です。
13	13	システムTCD受信時のメッセージサイズが不正です。
14	14	ローカルDF以外からシステムTCDを受信しました。

<ユーザのプログラム例>

```
nx_smsg( buf )
long *buf;
{
    extern long err_g[20];
    long i;

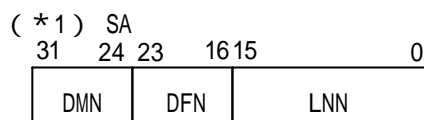
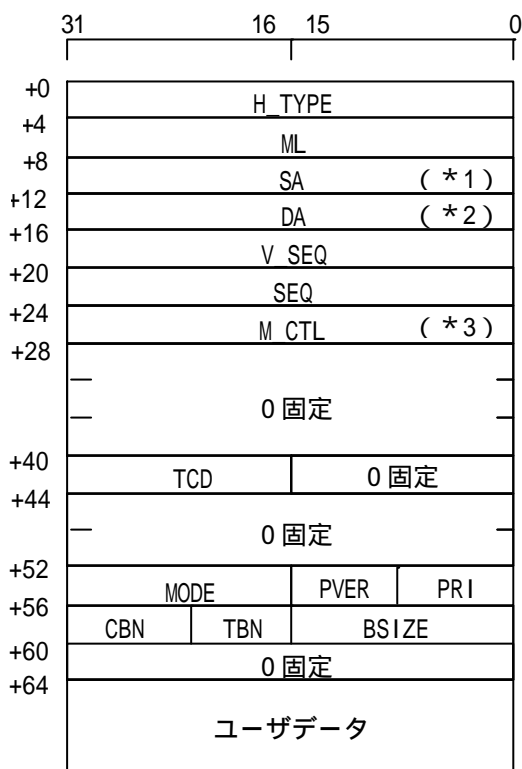
    for( i=0; i<20; i++ ) {
        err_g[i] = buf[i];
    }

    return;
}
```

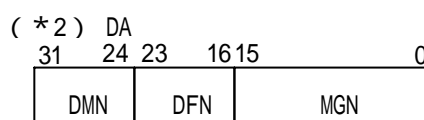
<ユーザのプログラムの登録例>

```
$ mcc68k -c -S -Fsm プログラム名称<ret>
$ asm68k -l -f " case,-t" プログラム名称.lst プログラム名称.src<ret>
$ sload プログラム名称 -a エリア名称 -f コマンドファイル名称 -w スタックサイズ +S<ret>
$ sbuild 38 プログラム名称 -S<ret>
```

付録D メッセージヘッダフォーマット



DMN : ドメイン番号 (現状 0 固定)
 DFN : データフィールド番号
 LNN : 送信元ノード番号

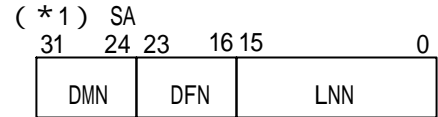
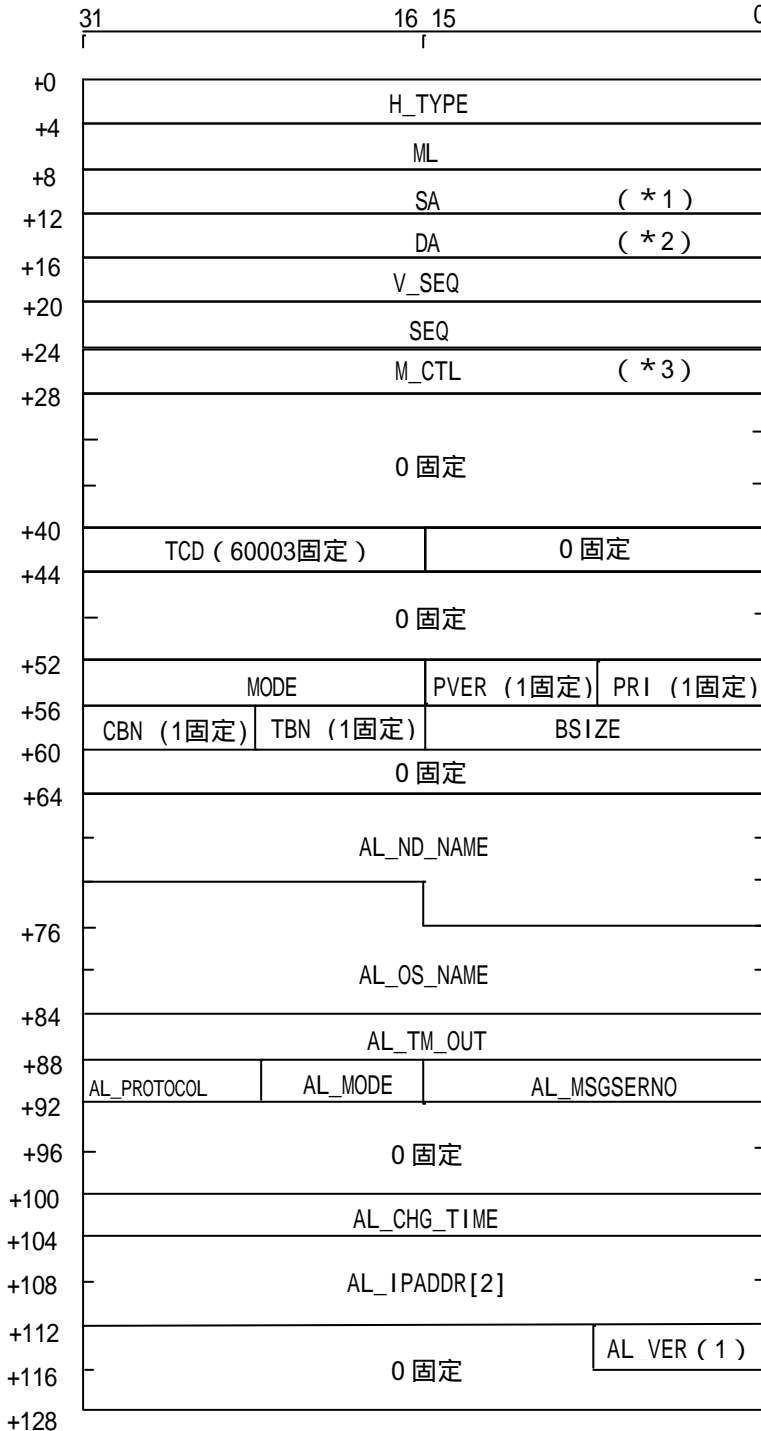


DMN : ドメイン番号 (現状 0 固定)
 DFN : データフィールド番号
 MGN : マルチキャストグループ番号

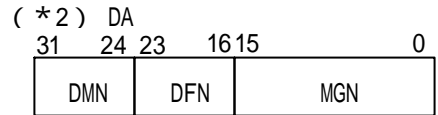
(*3) M_CTL : 0x80000000固定 (マルチキャスト通信)

記号名称	サイズ (B)	内 容
H_TYPE	4	ヘッダタイプ。ASCIIにて"NUXM"を設定。
ML	4	NeXUSヘッダを含めたメッセージの長さ。
SA	4	メッセージ送信元アドレス。(*1)参照。
DA	4	メッセージ送信先アドレス。(*2)参照。
V_SEQ	4	通番バージョン番号。通番が初期化された時刻を設定します。
SEQ	4	メッセージ送信通番。値は0x00000001 ~ 0x7FFFFFFF。
M_CTL	4	メッセージ送信制御情報。(*3)参照。
TCD	2	トランザクションコード。
MODE	2	メッセージモード。0 : オンラインモード、1 : テストモード
PVER	1	NeXUSプロトコルバージョン (1 固定)
PRI	1	メッセージの優先レベル (0 固定)
CBN	1	カレントブロック番号
TBN	1	トータルブロック数
BSIZE	2	分割パケット内のユーザデータサイズ (フラグメントブロックのバイト数)

付録 E 生存信号ヘッダフォーマット



DMN : ドメイン番号 (現状 0 固定)
 DFN : データフィールド番号
 LNN : 送信元ノード番号



DMN : ドメイン番号 (現状 0 固定)
 DFN : データフィールド番号
 MGN : マルチキャストグループ番号

(*3) M_CTL : 0x80000000固定
 (マルチキャスト通信)

記号名称	サイズ (B)	内 容
H_TYPE	4	ヘッダタイプ。ASCIIにて"NUXM"を設定。
ML	4	NeXUSヘッダを含めたメッセージの長さ。
SA	4	メッセージ送信元アドレス。(* 1) 参照。
DA	4	メッセージ送信先アドレス。(* 2) 参照。
V_SEQ	4	通番バージョン番号。通番が初期化された時刻を設定します。
SEQ	4	メッセージ送信通番。値は0x00000001 ~ 0x7FFFFFFF。
M_CTL	4	メッセージ送信制御情報 (0x80000000 固定)
TCD	2	トランザクションコード (6 0 0 0 3 固定)
MODE	2	メッセージモード。0 : オンラインモード、1 : テストモード
PVER	1	NeXUS プロトコルバージョン (1 固定)
PRI	1	メッセージの優先レベル (1 固定)
CBN	1	カレントブロック番号 (1 固定)
TBN	1	トータルブロック数 (1 固定)
BSIZE	2	分割パケット内のユーザデータサイズ。
AL_ND_NAME	10	ノード名称 (最後にNULLが設定されたASCII文字列。最大9文字)。
AL_OS_NAME	10	ベンダ機器名称 (HI_S10)。
AL_TM_OUT	4	生存信号タイムアウト監視時間 (秒)。
AL_MSGSERNO	2	生存報告メッセージ通番。
AL_MODE	1	生存報告モード。(* 4) 参照。
AL_PROTOCOL	1	プロトコル種別 (2 固定)。
AL_CHG_TIME	4	ノード状態変化発生時間。(* 5) 参照。
AL_IPADDR	8	IPアドレス (AL_IPADDR[0]はLAN1)。
AL_VER	1	生存信号メッセージバージョン番号 (1 固定)。

(* 4) AL_MODE = 1 : 通常モード (生状態)
= 2 : SHUTDOWN 予告モード

(* 5) AL_CHG_TIME : 立上げ時のグリニッジ時間にて、年月日時分秒を設定。
停止時 0 を設定。

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

機電事業部 産業情報制御システム部

電話(03)3258-1111(大代表)

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、当社営業担当または当社所員に、お渡しくださいますようお願い申し上げます。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	_____
ご意見欄	_____ _____ _____ _____ _____ _____ _____ _____ _____ _____ _____ _____ _____ _____ _____ _____ _____ _____ _____ _____ _____