

KARACRIX 入門実用ガイド

2章 通信制御ドライバS1について

(章別取扱説明書 v3.00)

本章は、以降の章を読み進めながら必要に応じて参照して頂いて構いません。
第3章以降に通信制御ドライバS1使用の解説がありますのでここで取り上げました。

株式会社 エスアイ創房

KaracrixBuilder

改定履歴

第 3.00 版 2009/11/01

おことわり

- (1) 本書内容の一部又は全部を、無断で他に転載することは禁止されています。
- (2) 本書内容は、将来予告無く変更する場合があります。

KARACRIX は株式会社エスアイ創房の登録商標です。

Microsoft, Windows, Excel は米国 Microsoft Corporation の登録商標です。

その他、本文中に記載されている社名および商品名は、一般に開発メーカーの登録商標です。

KARACRIX 入門実用ガイド 第 3.00 版 © S.I.Soubou Inc.

目次

2 章	通信制御ドライバ S 1 について.....	2-1
2.1	通信制御ドライバ S 1 の機能.....	2-1
2.2	通信制御ドライバ S 1 の位置付け	2-2
2.3	通信制御ドライバ S 1 の動作.....	2-3
2.4	通信制御ドライバ S 1 のインストールとコンパイル.....	2-5
2.5	制御パラメータの構成.....	2-9
2.6	制御パラメータ(通信情報部)の登録とリモート I / O 装置の関係.....	2-10
2.7	付録.....	2-17

2章 通信制御ドライバ S 1 について

KaracrixBuilder とリモート I/O 装置を組み合わせてシステム構築をおこなう際に、PC 上で実行されてリモート I/O 装置との通信部分を担当するプログラムが「通信制御ドライバ S1」です。本章では、この「通信制御ドライバ S1」の動作概要と使用法について解説します。

KaracrixBuilderV3 システムマニュアルでは通信制御ドライバ S1 のことを「プログラム S1」と説明しています。これは、KaracrixBuilder 側から見た通信制御ドライバ S1 はユーザプログラムの1つである為「プログラム」と説明しています。ただし、本ガイドで S1 の事をドライバと呼んでいる理由は、操作情報をコマンドに変換しリモート I/O 装置を駆動(ドライブ)する役割を担っている為「ドライバ」と呼んでいます。

本章で用いられる用語、「ポイント ID」「オブジェクト ID」「ポイントオブジェクト ID」「OBJID」は全て同義語です。

2.1 通信制御ドライバ S 1 の機能

基本機能

通信制御プログラムパラメータの定義に基づいてポイントオブジェクトとリモート I/O 装置を結合します。

1. ポイントオブジェクトメモリへの読み書きを行います。
2. コマンドキューの登録を読み込み処理します。
3. ポイントオブジェクト属性情報による警報の判断を行います。
4. リモート I/O 装置へのソケット通信処理を行います。

多機能処理機能

ポイントオブジェクト属性値の設定を併用することで以下に示す多機能な処理を行なっています。

1. リモート I/O 装置をポーリングアクセスします。
2. 通信装置のオンオフライン状態を監視します。
3. データ計測します。
4. リモート I/O 装置を操作します。
5. 接点警報を監視します。
6. アナログ上下限警報を監視します。
7. 操作履歴を生成します。
8. 警報履歴(通信エラー含む)を生成します。
9. 警報ブザーを鳴らします。
10. 警報メールを送信します。
11. 警報ロックを受け付けます。
12. リモート I/O 装置停復電時の積算カウント値を自動回復させます。

2.2 通信制御ドライバ S 1 の位置付け

KaracrixBuilder では、ユーザー (あなたの) プログラムからリモート I/O 装置の I/O 端子に接続された各種センサの値を取り込んだり、アクチュエータを操作するために、リモート I/O 装置と直接通信して制御する機能を担当する「通信制御ドライバ S1」を提供しています。

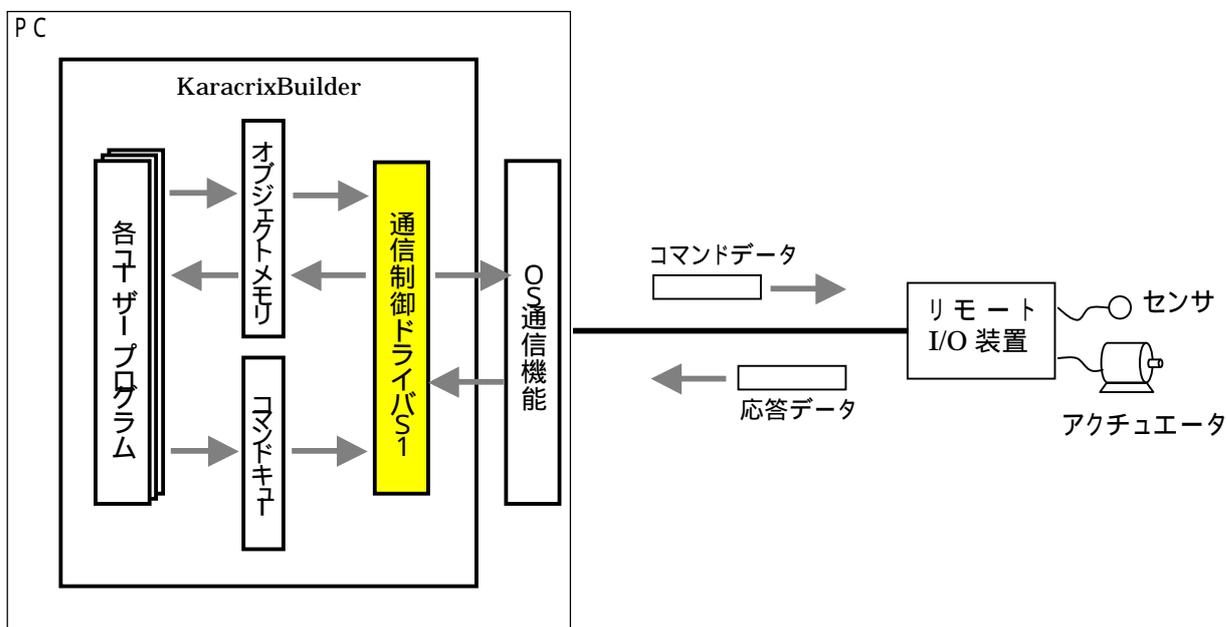


図 2.2.1 通信制御ドライバS1の位置付け

通信制御ドライバ S1(以下 S1)を使用する場合を考えて、各ユーザープログラムと S1 の間に情報受け渡し機構として「オブジェクトメモリ」と「操作コマンドキュー(送信用 FIFO メモリ)」を配置しています。

S1 は、センサなどが接続されているリモート I/O 装置の値を定期的に取り込んでオブジェクトメモリに書き込みます。また、コマンドキューを参照してアクチュエータへの操作が登録されていればこれをリモート I/O 装置に送信します。このサービスにより、S1 が稼働されている環境に於いて、ユーザープログラムがセンサの値を取得したい場合にはオブジェクトメモリの値を参照するだけで情報取得することができます。また、ユーザープログラムからアクチュエータなどの操作を行いたい場合には操作値をコマンドキューに登録すればあとは通信制御ドライバ S1 が送信してくれます。

オブジェクトメモリとは

状態などオブジェクトの各情報を格納する PC 内のメモリです。メモリは、KaracrixBuilder システムやユーザーが作成する自動制御プログラムによって読み書き共有使用されます。

コマンドキューとは

ユーザーの自動制御プログラムがリモート I/O 装置に繋がるアクチュエータを操作する場合、操作情報を格納する一時蓄積用のメモリです。蓄積された操作情報は通信制御ドライバ S1 が順番(FIFO)に取り出しリモート I/O 装置にコマンドを送出します。

2.3 通信制御ドライバ S1 の動作

通信制御ドライバ S1 の機能についてももう少し詳しく解説します。

通信制御ドライバ S1 は、S1 用の通信制御プログラムパラメータに登録されている複数のリモート I/O 装置の接続情報 (IP アドレス、ポート番号等) と I/O 端子をどのポイントオブジェクト ID (OBJID) にひも付けるかという定義情報を取得して動作します。

ポイントオブジェクト ID (OBJID) とは

オブジェクトの記号です。システムでオブジェクトを識別するために用います。英数半角 8 文字以内 (大小文字識別) の文字列で設定は一意 (唯一) でなければなりません。 (KaracrixBuilderV3 システムマニュアル 20 章「オブジェクト環境設定」参照)

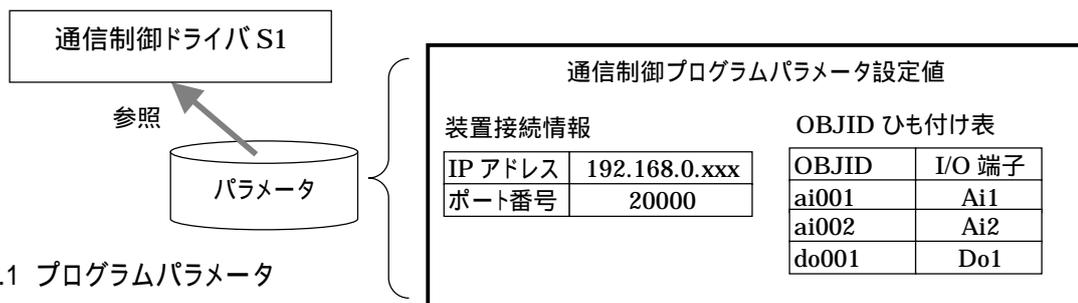


図 2.3.1 プログラムパラメータ

通信制御ドライバ S1 は、リモート I/O 装置と通信して I/O 端子に接続された各種センサの状態を取得して、その端子にひも付けられたポイントオブジェクト ID のオブジェクトメモリに値を書き込みます。

また、コマンドキューから操作値を取得してアクチュエータを操作するコマンドをリモート I/O 装置に送信し操作応答を得ます。

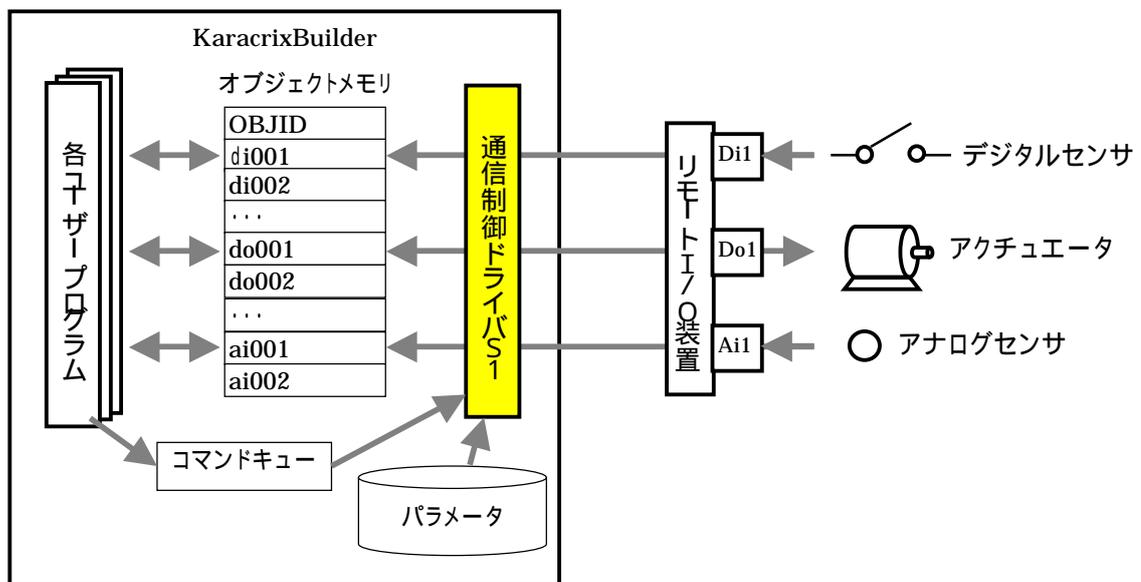


図 2.3.2 オブジェクトメモリを介して連携するユーザープログラムと通信制御ドライバ S1

2.3 通信制御ドライバS1の動作

ユーザープログラムからオブジェクトメモリの値を取得したり、コマンドキューに登録するには、KCX ライブラリ関数を使用します。また、通信制御ドライバS1 は OS のシステムコール(ソケット通信)を使用してリモート I/O 装置と通信します。

以下の図に、ユーザープログラムとリモート I/O 装置間の動作連携のイメージを示します。

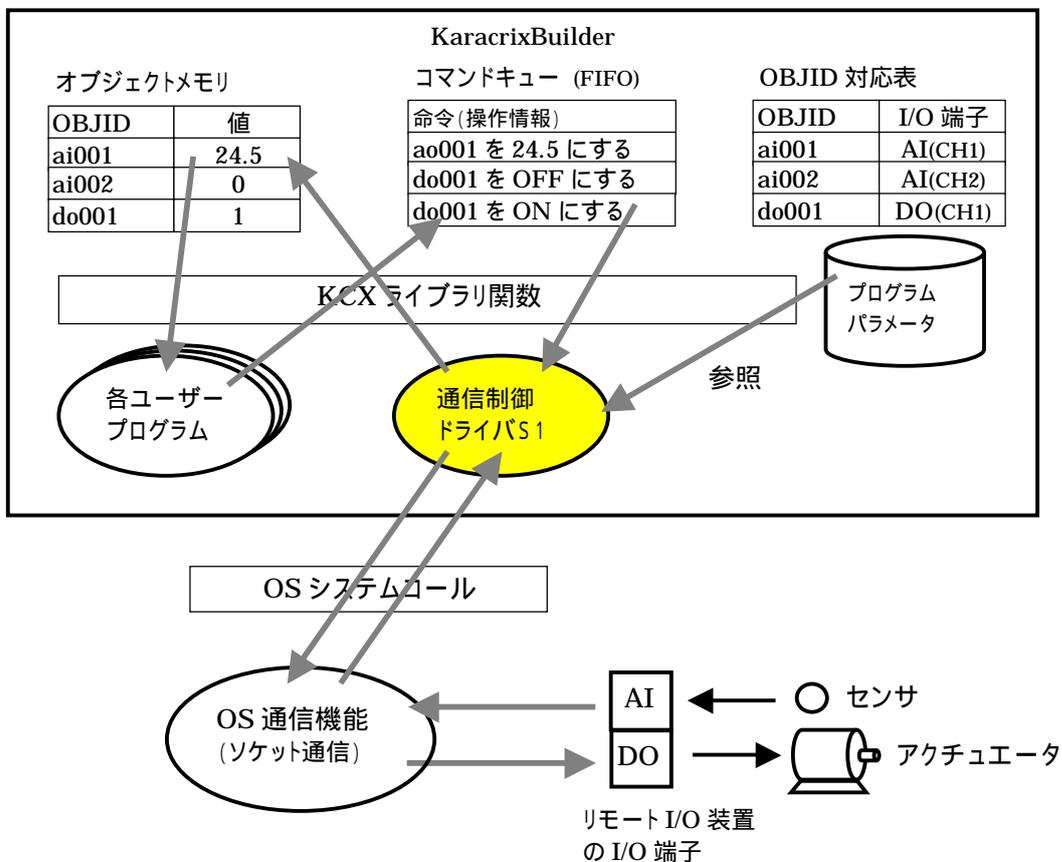


図 2.3.3 通信制御ドライバS1の動作イメージ

ドライバS1はプログラムパラメータを参照してオブジェクト ID とリモート I/O 端子の対応を取得する
 ユーザープログラムから操作したいオブジェクト ID とその操作値をコマンドキューに登録する
 ドライバS1はコマンドキューを調べて登録されているオブジェクト ID と操作値を取り出す
 ドライバS1は OS の提供するソケット通信機能(送信)を呼び出す
 OS の提供するソケット通信機能を使用してコマンドをリモート I/O 装置に送信する
 OS の提供するソケット通信機能を使用してリモート I/O 装置から応答(データ)を取得する
 ドライバS1は OS の提供するソケット通信機能(受信)を呼び出す
 ドライバS1はリモート I/O 装置からの応答(計測データ)をオブジェクトメモリに書き込む
 ユーザープログラムはオブジェクト ID を指定してドライバS1が書き込んだ計測データを取得する

2.4 通信制御ドライバ S1 のインストールとコンパイル

弊社ホームページのダウンロードのページから最新の通信制御ドライバ S1 をダウンロードすることができます。ダウンロードのページには、以下に示す画面の様に「通信 & 警報マネージャ多機能プログラム (通信制御ドライバ S1)」のタイトルで登録されていますので、使用する OS 環境に合ったプログラムリソースファイルをダウンロードして下さい。

ファイル名は、kcx_com_driver_S1_utf8.pgc または、kcx_com_driver_S1_euc.pgc になっています。

OS 環境の違いとは、デフォルトの文字コードが UTF8 (ユニコード) か EUC かの違いによるものです。最新の各種 OS の文字コードは、国際化のため UTF8 に移行しています。例えば、CentOS5.0 からは UTF8 がデフォルトの文字コードになっています。

ダウンロードしたプログラムリソースファイルを「制御プログラム登録」画面でインポートして下さい。*1

*1 ダウンロードしたリソースファイルのインポートの方法は、「KaracrixBuilderV3 システムマニュアル 18 章 リソースファイルの入出力」を参照して下さい。

以下に、CentOS5.0 の環境で作業をおこなう前提で、弊社ホームページのダウンロードのページ (2009 年 9 月現在のもの) からダウンロードする例で解説します。(最新のページ構成は変更される場合がございますので予めご了承下さい)

(1) リソースファイルのダウンロード

ダウンロードのページを表示します。CentOS5.0 は UTF8 の環境になっていますので、UTF8 用のファイル「kcx_com_driver_S1_utf8.pgc」をダウンロードして下さい。ダウンロードするファイルを保存する PC 内の場所は、/tmp (スラッシュ・テンプというディレクトリ(フォルダ)) です。場所にご注意ください。

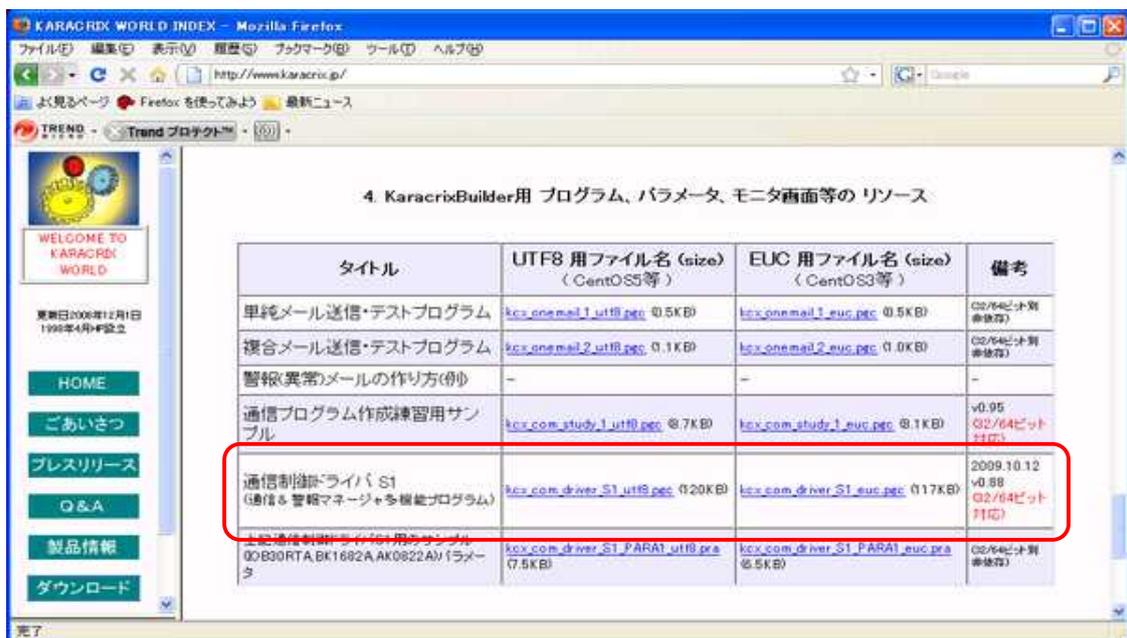


図 2.4.1 ダウンロードのホームページ

2.4 通信制御ドライバS1のインストールとコンパイル

(2) リソースファイルのインポート

「制御プログラム登録」画面を表示してプログラムが未登録の行で“プログラム名”欄を選択すると文字入力ダイアログが表示されますので、ここでプログラム名を登録して下さい。プログラム名は、「通信制御ドライバS1」で登録するものとします。



図 2.4.2 プログラム名の登録

ダウンロードしたリソースファイルが/tmp にある場合には、「外部データを取り込みますか?」と確認ダイアログが表示されますので“YES”を選択すると、続いて下図のようにリソースファイルの選択ダイアログが表示されますので、ここでインポートするファイル名が表示されている欄を選択して下さい。

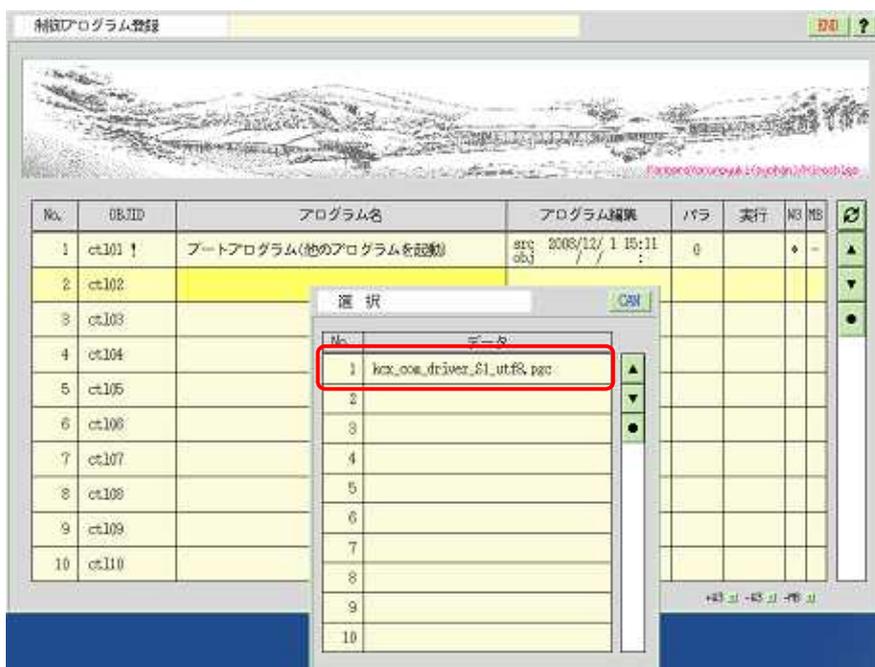


図 2.4.3 プログラムのインポート

このリソースファイルには、制御パラメータが添付されていますので、以下の画面の様に「プログラムからパラメータを分離取り込みますか?」とメッセージが表示されると思います。ここで、“YES”を選択してパラメータの分離取り込みを行なって下さい。制御パラメータについては、後ほど解説します。



図 2.4.4 パラメータのインポート

上記作業でインポートされるものは、「通信制御ドライバS1」プログラムと「制御パラメータ」の2種類です。インポートが成功した直後の画面を以下に示します。

“プログラム編集”欄に「src yyyy/mm/dd hh:ss」形式で登録された日時が表示されます。また、“パラ”欄には、設定できる制御パラメータの数が表示されています。



図 2.4.5 パラメータ欄の確認

2.4 通信制御ドライバS1のインストールとコンパイル

(3) 通信制御ドライバS1のコンパイル

インポートされた「通信制御ドライバS1」プログラムを実行できる形式に変換するために、コンパイル作業を行なう必要があります。

(OS 環境確認項目)

コンパイル作業を行なう前に、OS環境に開発パッケージがインストールされていることを確認して下さい。以下の画面のように「アプリケーション」「ソフトウェアの追加/削除」メニューでパッケージ管理ツールを起動して「X ソフトウェア開発」「レガシーなソフトウェアの開発」「開発ライブラリ」の3つの項目がインストールされていることを確認します。コンパイルを行なうためには上記の開発パッケージが必要です。もし、インストールされていないパッケージがある場合には、パッケージの追加をして下さい。

コンパイル作業が正常に行える環境が揃えば上記の限りではありません。



図 2.4.6 開発環境の確認

「制御プログラム登録」画面で、“プログラム編集”欄を選択して「プログラム簡易編集」画面でコンパイルを実行します(*2)。コンパイルが成功すると、以下の図に示すように“プログラム編集”欄のobjにコンパイル日時が表示されます。

*2 「プログラム簡易編集」画面でのコンパイル作業の解説は、「KaracrixBuilderV3 システムマニュアル 12章 制御プログラム」を参照して下さい。

No.	ObjID	プログラム名	プログラム編集	パラ	実行	MB	
1	ct101	ブートプログラム(他のプログラムを起動)	src 2008/12/1 18:11 obj 2008/12/1 18:11	0			▲
2	ct102	通信制御ドライバS1	src 2008/3/20 19:20 obj 2008/3/20 19:20	38	--	--	▼
3	ct103						●
4	ct104						
5	ct105						
6	ct106						
7	ct107						
8	ct108						
9	ct109						
10	ct110						

図 2.4.7 実行可能なコンパイル済みプログラム

2.5 制御パラメータの構成

前節の解説ではダウンロードしたプログラムリソースファイルをインストールした際に「制御パラメータ」の分離取り込みをしていますが、制御パラメータの構成について概要を解説します。

「メインメニュー」「制御パラメータ入力」「パラメータ編集」画面を表示して下さい。



図 2.5.1 パラメータ編集画面

パラメータとは、プログラム動作を指示する為に与えるデータです。

通信制御ドライバS1では、各パラメータを下記(図 2.5.2)に示すよう全体情報部と通信情報部に大きく2つに分けています。この中の、通信制御に係わらない全体情報部については「付録」で解説していますので、そちらを参照して下さい。本節では、通信制御に係わる箇所を以下で解説します。

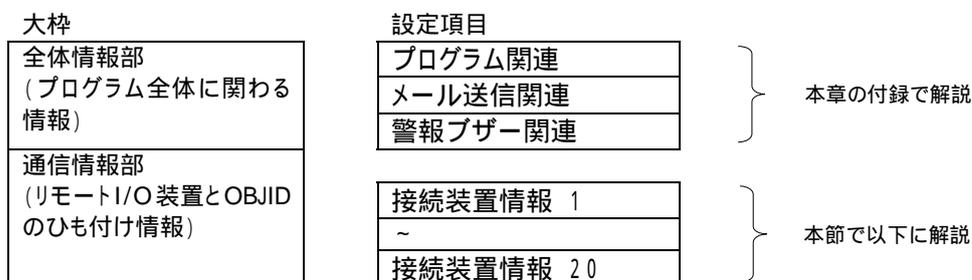


図 2.5.2 制御パラメータの構成

通信情報部は、リモート I/O 装置の入出力端子と OBJID とのひも付け情報を定義する領域になります。通信制御ドライバS1ではこの通信情報部を読み込んでリモート I/O 装置と自動接続を行います。

2.6 制御パラメータ(通信情報部)の登録とリモートI/O装置の関係

通信制御ドライバ S1 の制御パラメータは、リモート I/O 装置の入出力端子と KaracrixBuilder の OBJID をひも付ける変換テーブルとして参照されますので、下図の様な「パラメータ編集」画面等を使用して取り込んだ制御パラメータに値を設定することが重要な作業になります。

ここでは、KaracrixBuilder24A を例に、デフォルト登録されているポイントオブジェクトを弊社リモート I/O 装置を使用してひも付ける場合で解説します。

「メインメニュー」「制御パラメータ入力」「パラメータ編集」画面の通信情報部を表示して下さい。



図 2.6.1 パラメータ編集画面

リモート I/O 装置のひも付けを定義する項目は、以下のような設定項目として表示されます。

接続装置 ?
 装置 ? 割付ポイント ID
 + 割付ポイント ID
 + 割付ポイント ID

なお、ダウンロードしたリソースプログラムにおける接続可能なリモート I/O 装置の数は 20 台までとなっています。この数は、通信制御ドライバ S1 のプログラムの中で、台数(NETBOXs)を 20 に定義(#define)している記述のところによって決まっています。20 台以上の装置を接続する場合には、この定義値を大きくして(プログラムを再コンパイルしてから)使用して下さい。

(1) 設定データのフォーマット

リモート I/O 装置 1 台分の接続情報を定義するパラメータ設定項目に 4 行(固定)を割り当てています。設定項目の 1 行目には、装置の型認識 ID、IP アドレス、ポート番号、オプション(必要時)をスペース区切りで設定します。2、3、4 行目には、装置の入出力端子にひも付けるポイントオブジェクト ID を設定していきます。ポイント ID の記述は、3 行数以内で記述します。もし、全てのポイント ID を 1 行に記述することができる場合にはこれを 1 行に記述して残る 2 行は空白でも構いません。次に、ポイント ID 数は、使用する装置が保有する I/O の種類数分を全て埋め込んで記述する必要があります。記述する順番によりポイント ID と I/O の種類が対応するようにひも付けします。ひも付けるポイント ID がない(I/O 未使用)場合には、その I/O に対応する位置に未使用を意味する“-”(ハイフン)を記述するものとします。

1 行目のフォーマット [:空白(半角)]

装置型認識 ID	IP アドレス	ポート番号	オプション(必要時)
----------	---------	-------	------------

装置型認識 ID: TK、BK、AK、AK2、RT (弊社リモート I/O 装置の型識別 ID)

TK: KaracriBoard-TK0040A

BK: NetBox-BK1682A

AK: NetBox-AK0822A

AK2: NetBox-AK0620A

RT: KCXH-IOB30RTA

IP アドレス

装置の IP アドレスを xxx.yyy.zzz.www 形式で記述します。

このアドレスは、装置に設定されている IP アドレスと合わせておく必要があります。

装置のデフォルト IP アドレスは、192.168.0.200 となっています。

ポート番号

装置のコマンド待受けポート番号を記述します。

このポート番号は、装置に設定されているポート番号と合わせておく必要があります。

装置のデフォルトポート番号は、20000 となっています。

オプション

オプションは、装置に対し動作モードなどを設定する場合に用います。(現在未使用)

本行左端(装置型認識 ID の頭部)に#を記述すると装置 1 台分の設定情報(4 行分)は全てコメント行として扱われ、通信制御ドライバ S1 が読み込むときに無視されます。以下に設定例を示します。

(記述例: #TK...、#BK...、#AK...、#AK2...、#RT...)

2.6 制御パラメータ(通信情報部)の登録とリモートI/O装置の関係

2~4行目のフォーマット [:空白(半角)]

OBJID or - OBJID or - OBJID or -

装置 I/O 数分の並びに対応したポイントオブジェクト ID を記述します。

未使用の I/O には、“-”(ハイフン)を記述します。

通信制御ドライバ S1 でサポートしている弊社リモート I/O 装置が持っている I/O 端子の型別の記述の並びを以下に示します。この並び順と同じ並び順で、ポイントオブジェクト ID を割り付けます。

例えば TK0040A の I/O 端子は、Di:6ch、Pi:6ch、Do:4ch、Ai:4ch、Ao:3ch で計 23ch の入出力端子を持っています。この場合の設定例を以下に示します。

表 2.6.1 TK0040A 端子設定例

	I/O 端子の設定並び順
KaracriBoard-TK0040A の I/O 端子型仕様	Di1 Di2 Di3 Di4 Di5 Di6 Pi1 Pi2 Pi3 Pi4 Pi5 Pi6 Do1 Do2 Do3 Do4 Ai1 Ai2 Ai3 Ai4 Ao1 Ao2 Ao3 (Di:6ch、Pi:6ch、Do:4ch、Ai:4ch、Ao:3ch) 計 23ch 
設定例 1	di001 di002 di003 di004 di005 di006 pi001 pi002 pi003 pi004 pi005 pi006 do001 do002 do003 do004 ai001 ai002 ai003 ai004 ao001 ao002 ao003
設定例 2	di002 di004 di006 di005 di003 di001 pi004 pi005 pi006 pi003 pi002 pi001 do003 do004 do001 do002 ai002 ai001 ai004 ai003 ao002 ao003 ao001
設定例 3	- di001 - - di005 - pi001 - pi003 - - - do001 - do003 do004 - - - - - ao003

(注) Pi1 ~ Pi6 は Di1 ~ Di6 の端子を共用して使用します。

設定例 1 は、全ての I/O 端子にポイントオブジェクト ID を昇順に全てひも付けた例です。

設定例 2 は、1 と同様に全ての I/O 端子にひも付けていますが、ポイントオブジェクト ID の使用番号がランダムに設定されています。このような設定も可能であるとの例です。

設定例 3 は、使用しない I/O 端子がある場合の例です。“-”を記述した I/O 端子はひも付けされません。

Di と Pi について

Pi とは計量等を目的としたカウント値で、接点入力 Di の開閉数を示します。

TK0040A では、Di 端子1つで、接点入力の状態とカウント値の同時2つの状態を得ています。

上記 I/O 端子とひも付けるポイントオブジェクト ID の対向関係を以下の図にまとめました。

I/O 端子	設定例 1	設定例 2	設定例 3
Di1	di001	di002	-
Di2	di002	di004	di001
Di3	di003	di006	-
Di4	di004	di005	-
Di5	di005	di003	di005
Di6	di006	di001	-
Pi1 (Di1)	pi001	pi004	pi001
Pi2 (Di2)	pi002	pi005	-
Pi3 (Di3)	pi003	pi006	pi003
Pi4 (Di4)	pi004	pi003	-
Pi5 (Di5)	pi005	pi002	-
Pi6 (Di6)	pi006	pi001	-
Do1	do001	do003	do001
Do2	do002	do004	-
Do3	do003	do001	do003
Do4	do004	do002	do004
Ai1	ai001	ai002	-
Ai2	ai002	ai001	-
Ai3	ai003	ai004	-
Ai4	ai004	ai003	-
Ao1	ao001	ao002	-
Ao2	ao002	ao003	-
Ao3	ao003	ao001	ao003

図 2.6.2 端子設定対向関係

通信制御ドライバ S1 で対応している他の弊社製のリモート I/O 装置を使用する場合の設定並び順を以下に示しますので参考にして下さい。

表 2.6.2 リモート I/O 装置端子仕様

NetBox-BK1682A の I/O 端子型仕様	Di1 Di2 Di3 Di4 Di5 Di6 Di7 Di8 Di9 Di10 Di11 Di12 Di13 Di14 Di15 Di16 Pi1 Pi2 Pi3 Pi4 Pi5 Pi6 Pi7 Pi8 Pi9 Pi10 Pi11 Pi12 Pi13 Pi14 Pi15 Pi16 Do1 Do2 Do3 Do4 Do5 Do6 Do7 Do8 Do21 Do22 (Di:16ch、Pi:16ch、Do:8ch、Do2:2ch) 計 42ch
NetBox-AK0822A の I/O 端子型仕様	Ai1 Ai2 Ai3 Ai4 Ai5 Ai6 Ai7 Ai8 Ao1 Ao2 Do21 Do22 (Ai:8ch、Ao:2ch、Do2:2ch) 計 12ch
NetBox-AK0620A の I/O 端子型仕様	Di1 Di2 Pi1 Pi2 Do1 Do2 Ai1 Ai2 Ai3 Ai4 Ai5 Ai6 Ai7 Ai8 Ai9 Ai10 Ai11 Ai12 Ao1 Ao2 (Di:2ch、Pi:2ch、Do:2ch、Ai:12ch、Ao:2ch) 計 20ch
KCXH-IOB30RTA の I/O 端子型仕様	Ai1 Ai2 Ai3 Ai4 Ai5 Ai6 Ai7 Ai8 Di1 Di2 Di3 Di4 Do1 Do2 Do3 Do4 (Ai:8ch、Di:4ch、Do:4ch) 計 16ch

(2) リモートI/O装置を複数台接続する場合の設定方法

センサやアクチュエータが広範囲に分散して配置されている場合には、複数台のリモートI/O装置を使用する場合があります。この場合、必要な台数分のリモートI/O装置情報を制御パラメータとして設定します。使用するリモートI/O装置は、TK0040A、BK1682A、AK0822A、AK0620A、IOB30RTAが混在していても構いません。通信制御ドライバS1は、設定されているリモートI/O装置を順番に周回通信制御(ポーリング)していきます。

以下に、TK0040Aを2台使用した場合の設定例を示します。

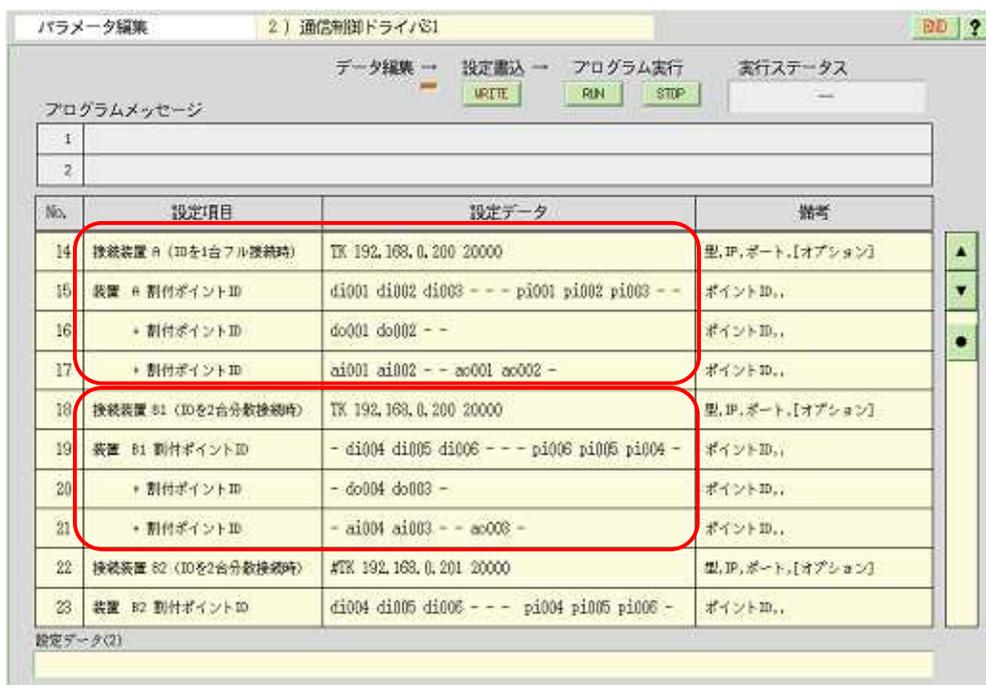


図 2.6.3 複数台設定例のパラメータ編集画面

TK0040Aを2台使用した設定例

表 2.6.3 TK0040A2台使用時の端子設定例

	I/O 端子の設定並び順
KaracriBoard-TK0040A の I/O 端子型仕様	Di1 Di2 Di3 Di4 Di5 Di6 Pi1 Pi2 Pi3 Pi4 Pi5 Pi6 Do1 Do2 Do3 Do4 Ai1 Ai2 Ai3 Ai4 Ao1 Ao2 Ao3 (Di:6ch, Pi:6ch, Do:4ch, Ai:4ch, Ao:3ch) 計 23ch
装置 1	di001 di002 di003 - - - pi001 pi002 pi003 - - - do001 do002 - - ai001 ai002 - - ao001 ao002 -
装置 2	- di004 di005 di006 - - - pi006 pi005 pi004 - - - - do004 do003 - - ai004 ai003 - - ao003 -

(3) 間違った設定例

KaracriBuilderで管理する1つのポイントオブジェクトIDを、装置に対して複数(1つの装置内に複数箇所、あるいは複数の装置に複数箇所)設定しないで下さい。

入力型のポイントの場合は、後優先読み込みで動作しますが、出力型の場合は出力先が不定となって正常な動作を保証できなくなります。以下の図では、do002とdo003を複数の装置に複数箇所設定してある間違った例を示しています。

	I/O 端子の設定並び順
KaracriBoard-TK0040A の I/O 端子型仕様	Di1 Di2 Di3 Di4 Di5 Di6 Pi1 Pi2 Pi3 Pi4 Pi5 Pi6 Do1 Do2 Do3 Do4 Ai1 Ai2 Ai3 Ai4 Ao1 Ao2 Ao3 (Di:6ch, Pi:6ch, Do:4ch, Ai:4ch, Ao:3ch) 計 23ch
装置 1	di001 di002 di003 - - - pi001 pi002 pi003 - - - do001 do002 do003 - ai001 ai002 - - ao001 ao002 -
装置 2	di004 di005 di006 - - - pi004 pi005 pi006 - - - - do002 do003 do004 ai003 ai004 - - ao003 - -



図 2.6.5 間違った設定例

付 録

制御パラメータ(全体情報部)の解説

制御パラメータの構成の中で全体情報部(リモート I/O 装置のひも付け情報を除くプログラム全体に関わるデータ)について、2.5 項「制御パラメータの構成」から引き継ぎ、解説します。

「メインメニュー」「制御パラメータ入力」「パラメータ編集」画面を表示して下さい。



図 2.7.1 パラメータ編集画面

プログラム関連情報

1. 起動時の接続試験モード

本パラメータは、通信制御ドライバS1が通信情報部(I/O 端子ひも付け)のパラメータを参照してリモート I/O 装置との通信接続試験を行なう時にプログラムメッセージを表示するかどうかを判断するために使用されます。リモート I/O 装置との通信情報パラメータ設定を記述する作業が正しく完了するまで、本パラメータを on に設定しておくことを推奨します。通信制御ドライバS1では、本起動時の接続試験モードが on の場合、設定された通信情報パラメータ内容を読み取り正誤の検査を行いながら動くようになっており、パラメータを読み取る過程をプログラムメッセージで確認することができます。また、設定したポイントの数やポイントオブジェクトの型が I/O 端子の型と合わない場合には、そのメッセージがプログラムメッセージにも出力されるようになっていきますので、設定した通信情報パラメータのミスも発見しやすくなります。モードを off にした場合、上記のプログラムメッセージは出力されません。パラメータ設定が正しいものとして、すぐにリモート I/O 装置との通信処理に移りますので、プログラムの起動時間を短縮することができます。なお、起動時間を気にする必要がない場合には、本モードを on にしておくことをお勧めします。

2. プログラムオプション

通信制御ドライバS1にオプションデータを与える時に用います。プログラムの付加機能を有効にしたり無効にしたり、またグローバルなパラメータを与えたりに使用します。現在は、未使用です。

メール送信関連情報

3. メール送信許可

ポイント(オブジェクト)毎のメール送信許可は、「メニュー」 「システム設定(システム環境設定メニュー)」 「ポイント登録」の警報(用途/選択)分類の“警報メール許可”で設定します。



図 2.7.2 警報メール許可設定

本パラメータによるメール送信許可は、その全てのポイントのメール送信許可の元栓に当たる許可を行います。

on 許可

off 不許可

本メール送信許可にオプションを与えることができます。

第2引数に時間を与えると、その時間内だけ on (送信許可) されます。

時間には、許可開始時間(Sh 時 Sm 分)と終了時間(Eh 時 Em 分)を与えます。

時間のフォーマットは、Sh:Sm-Eh:Em です。(スペースを挟まずに1文字列として記述します)

例 1) 7:30 ~ 21:40 の場合、「on 07:30-21:40」と記述します。(左記 はスペースを意味します)

例 2) 23:45 ~ 6:30 の場合、「on 23:45-06:30」と記述します。(左記 はスペースを意味します)

4. メールアドレス(1)～(5)

メール送信先の E メールアドレス(xxx@yyy.zz)を記述します。

そして、ここで(図 2.7.1)格納したメールアドレス(1)～(5)の番号(数値 1～5)を、下図 2.7.3 に示す各ポイントの汎用属性の「整数 属性 1～3」(メール送信先 1～3)の設定欄で、事象発生によりポイントが送信したいメールアドレスを必要数分指定します。(下図メール送信先 1～3 の設定欄の値が 0 の場合は無効を意味します)

No.	設定項目	説明	設定
1	メール送信先 1	メールアドレスが格納されている番地番号	0
2	メール送信先 2	メールアドレスが格納されている番地番号	3
3	メール送信先 3	メールアドレスが格納されている番地番号	4
4	整数 属性 4		0
5	整数 属性 5		0

通信制御ドライバ S1 を実行すると、「整数属性 1～3」のデフォルト設定項目名が「メール送信先 1～3」に変更されます。

図 2.7.3 ポイント属性設定画面(メール送信先の間接番号指定)

E メールアドレス(図 2.7.1)の先頭に#を付けておくと、メールアドレスを無効にする事ができます。

つまり、メールアドレスを記述していないのと同じ事となります。

メールアドレスを保留にする時などにご利用下さい。

例) abc@nippon.xx 有効

例) #abc@nippon.xx 無効(保留)

通信制御ドライバ S1 がポイントの警報を検出してメールを送信する時に E メールアドレスを取得するまでの手順を説明しておきます。

1. 制御パラメータに設定してあるメール送信許可(元栓)を調べます。on 以外の場合は処理を中止します。
2. 対象ポイント毎にポイント属性の警報メール許可を調べます。off の場合そのポイントの送信処理を中止します。
3. 警報が発生しているポイント属性の汎用の整数属性 1～3 をサーチします。
4. 整数属性 1～3 の設定値が、1～5 の範囲(0 は未登録を意味する)の場合、その値は制御パラメータ(図 2.7.1)内のメールアドレス格納番号 1～5 を示すものとして、制御パラメータのメールアドレス(1)～(5)をサーチします。
5. メールアドレス(1)～(5)に有効(空白以外、E メールアドレスの頭に無効を意味する # が付いていないもの)な E メールアドレスが設定されていた場合には、送先の E メールアドレス名が S1 によって取得され使用されます。
6. 1 つのポイントオブジェクトから同時送信できる宛先メールの数は、制御パラメータ内に格納された最大 5 つの E メールアドレスのうち、汎用の整数属性 1～3 で指定できる最大 3 つまでです。

5. メール装置名(共通)

メール本文の先頭行に装置名を入れる場合に使用します。メールがどの装置 (KaracrixBuilder) から送られてきたのかが受信側で分かります。ここで指定した文字列が以下の画面 (図 2.7.4) に示す様にメールを受信したときの本文に表示されます。

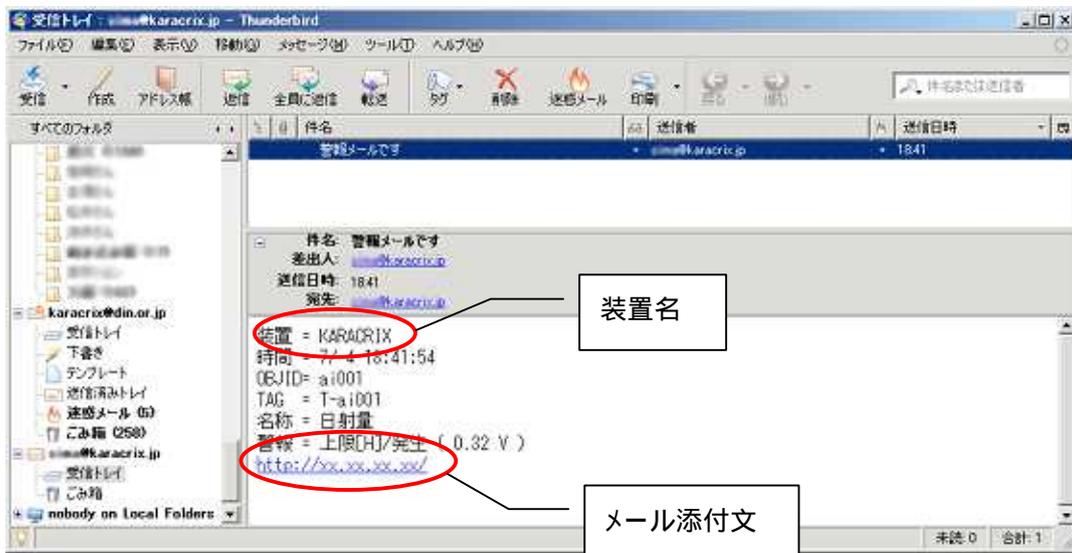


図 2.7.4 メールソフト表示例

6. メール添付文(共通)

メール本文の末行に一律コメントを入れる場合に使用します。

このコメントに例えば本システム(遠隔監視制御サーバ自身 / KaracrixBuilder)への URL を記述しておけば、メールが届いた時にこの URL をクリックし、ブラウザにリンクするなどして素早い異常確認の操作を行なうことも可能です。(メールソフト等の機能に依存します)

警報ブザー関連情報

7. 警報ブザーポイント ID

警報発生を知らせる例えばブザー装置等がある場合、これに接続するデジタル出力型のポイントオブジェクト ID を記述します。また、複数の装置を鳴動等させる場合、ポイント ID を複数記述することもできます。

No.	設定項目	設定データ	備考
11	警報ブザーポイントID	do001 do002	09型ポイントID
12	通信応答表示装置番号		装置番号 1~ (未指定時先頭機器)

図 2.7.5 警報ブザーポイントID設定

このポイント ID を記述した状態(有効)で、各ポイント属性の「警報制御コマンドコード」(警報ブザーポイント ID を使用する意味を持つ)に 1 を設定しておく、そのポイントに警報が発生する毎に、上記警報ブザーを ON させる事ができます。(下記の画面では、di001 ポイントに警報が発生した場合に本設定により警報ブザーのポイントオブジェクトが ON になります)



図 2.7.6 ポイント属性設定画面

警報ブザーを停止させるには、ブザーに繋がるデジタル出力型のポイントオブジェクトをOFFに操作するか、KaracrixBuilderのメインメニューにあるシステムボタンの「BzSTOP」を押すと停止できます。



図 2.7.7 メインメニュー画面

ポイント ID の先頭に#を付けると、設定が無効になります。

つまり、ポイント ID を記述していないのと同じ事となります。

例) #do001

例) do001 #do002 do001 は有効 do002 は無効(保留)

例) #do001 do002 #do003 do002 は有効 do001 と do003 は無効(保留)

8. 通信応答表示装置番号

リモート I/O 装置と通信中のコマンド応答文をモニターすることができます。

モニターできるのは、指定するリモート I/O 装置1台分のみで、その装置番号(1~20[最大定義値])を設定します。

省略時は、先頭に設定(接続装置情報)されたりリモート I/O 装置のコマンド応答文と見なします。

