

**SECS／PLC 通 信 接 続 プ ロ グ ラ ム**  
**( tdISPLC )**

**(Trust Design Simple SECS/PLC Communication Connection)**

**取 扱 説 明 書**

Version 15.120 : 2015.12.04  
Version 16.011 : 2016.01.15  
Version 16.012 : 2016.02.25  
Version 16.040 : 2016.04.05  
Version 18.032 : 2018.03.28

**合 同 会 社      ト ラ ス ト デ ザ イ ン**

**長野県 諏訪市 四賀赤沼 1870-1-B105**

Tel:0266-78-7970      E-mail:info@trust-design.co.jp  
Fax:0266-78-7971      URL:http://www.trust-design.co.jp

## 目

## 次

1. はじめに
2. P L C通信仕様 及び 設定ファイルの記述
3. 操作説明

## 1. はじめに

本プログラムは、PLCによる制御装置を、SECS プロトコルで、上位システムに接続します。

本プログラムは以下の特徴を持ちます。

- ・ PLC により制御する装置を SECSプロトコルを使用して上位システムに接続するための Windows PC にて動作するアプリケーションです。
- ・ PLC 内で発生したデータを、S6F11 等の SECSメッセージに変換して上位に通知したり、上位からの S2F49 による指示を PLC 内メモリに設定することにより指示したりすることができます。
- ・ 疑似 PLC 装置を指定することにより、Windows AP を共有メモリへのアクセスのみで、SECS に関する処理を行うことなく、上位ホストに簡単に SECS 接続することができます。
- ・ 1 台の PC 内で複数の本 A P を稼働させることができ、ホストと複数の PLC 装置との接続が可能です。
- ・ 設定、動作確認後、Windows に対して本 A P の自動起動設定を行えば、以降は本 A P が稼働する Windows PC の電源 ON にて、自動運転する事が可能です。
- ・ 現在、使用可能な PLC 装置は以下です。
  - ・ 三菱 ..... MELSEC MC プロトコル (3E、4E フレーム)
  - ・ オムロン ..... OMRON FINS コマンド
  - ・ キーエンス ..... KEYENCE TCP/IP
  - ・ 疑似 PLC 装置 ... 共有メモリを、疑似 PLC メモリとしてデータ交換

- ・ [ご注意 1] -----+  
| 本プログラムはPLC装置をSECSプロトコルで上位ホスト・システムに接続するための実装例です。 |  
| 実際の製造現場等で使用するPLC装置を上位システムにSECS接続するためには、本プログラムに |  
| よる実装項目以外にも、実装しなければいけない機能（例えば、可変長リスト対応、オープン・トラ |  
| ンザクション、データ型変換、ハートビート、エラー処理等々）があるかと思います。それらは各 |  
| ライン、各装置毎に固有の実装となると思われるので、本プログラムの機能だけでは十分な実装とは |  
| ならないと思われます。 |  
| あくまで、本プログラムは「実装例」です。実際の現場における本プログラム（及び、本プログラム |  
| を改訂して）の現場への適用に関してのご相談は、弊社（info@trust-design.co.jp）まで、ご連絡 |  
| ください。 |  
+-----+

- ・ [ご注意 2] -----+  
| 本プログラムをご使用になる場合、お客様にてPLC側の処理（疑似PLC装置の場合はWindowsプログラム |  
| ）を実装していただく必要があります。 |  
| 本仕様書 2. (1) をご参照の上、必要な情報の設定、取得、通信に使用するフラグのON/OFF等を行って |  
| てください。 |  
+-----+

- ・ SECS/HSMS による通信システムの開発には、弊社 SECS/HSMS 通信パッケージ（Trust Design Simple SECS Communication Library）（使用ライセンス無料）を公開しております。詳しくは、弊社ホームページをご覧ください。
- ・ SECS-1 及び HSMS-SS による通信処理（ホスト側、装置側）をシミュレートする検査アプリケーションとして、SECS/HSMS シミュレータ（暫定版）（Trust Design Simple SECS/HSMS Simulator (Preliminary Version)）（使用ライセンス無料）を公開しております。詳しくは、弊社ホームページをご覧ください。
- ・ SECS-1、HSMS-SS 相互の通信プロトコルを変換するアプリケーションとして、SECS/HSMS プロトコル変換プログラム（Trust Design Simple SECS/HSMS Protocol Converter）（使用ライセンス無料）を公開しております。詳しくは、弊社ホームページをご覧ください。
- ・ SECS(HSMS) による通信のモニターには、弊社 ネットワーク通信モニター（Trust Design Simple Network Communication Monitor）（使用ライセンス無料）を公開しております。詳しくは、弊社ホームページをご覧ください。

## 2. PLC 通信仕様 及び 設定ファイルの記述

### (1) PLC 通信仕様

#### (a) 使用する PLC デバイス・メモリ領域

(注) 各項の 2 行目に記載した (bss0:PLC)等は、各項で説明する各デバイスのアドレスを示すニモニックであり、(b)以降の説明で使用する。' : '以降の PLC もしくは PC は、そのデバイス値を更新する機器を示す。

1. PLC 稼動状態を PLC が設定するビットデバイス (1bit) ..... PLCSSDEV0 (ssb0:PLC)  
PLC が、上位ホストとの SECS 通信可能な状態の場合に 1 とし、通信不能状態では 0 とする。
2. PC 稼動状態を PC が設定するビットデバイス (1bit) ..... PLCSSDEV1 (ssb1:PC)  
PC が、上位ホストと SECS 接続状態である場合に 1 とし、非接続状態では 0 とする。
3. PLC からの SECS Message 送信時のハンドシェイクに使用するビットデバイス (2bit) ..... PLCHSDEV0 (hsb00:PLC、hsb01:PC)  
PLC が SECS Message 送信を行う場合に、まず以下の 5、7 に示す領域に SECS Message に関する情報を設定した後、PLC 設定 bit (hsb00) を 1 とする。それを PC が検出し、5、7 領域の情報を受け取り、SECS Message 送信が完了したら PC 設定 bit (hsb01) を 1 とする。  
PLC は、PC 設定 bit (hsb01) が 1 である事を確認したら、PLC 設定 bit (hsb00) を 0 とし、PC は、PLC 設定 bit (hsb00) が 0 である事を確認したら、PC 設定 bit (hsb01) を 0 とする。
4. PLC への SECS Message 送信時のハンドシェイクに使用するビットデバイス (2bit) ..... PLCHSDEV1 (hsb10:PLC、hsb11:PC)  
PC が上位ホストから SECS Message を受信したら、その SECSMessage に関わる情報を、以下の 6、7 に示す領域に格納した後、PC 設定 bit (hsb11) を 1 とする。それを PLC が検出し、6、7 領域の情報を受け取り処理を完了したら PLC 設定 bit (hsb10) を 1 とする。  
PC は、PLC 設定 bit (hsb10) が 1 である事を確認したら、PC 設定 bit (hsb11) を 0 とし、PLC は、PC 設定 bit が 0 (hsb11) である事を確認したら、PLC 設定 bit (hsb10) を 0 とする。
5. PLC からの SECS Message 送信時の SECS Message 情報を格納するワードデバイス (20W) .. PLCMSDEV0 指定アドレス (msd0:PLC) + 0 : SF-Code 及び W-Bit  

FE	8	7	0	
+-----+-----+-----+				
+-----+ +-----+				
+-----				F-Code (0 - 127)
+-----				S-Code (0 - 63)
+-----				W-Bit

  
+ 1 : DeviceID  

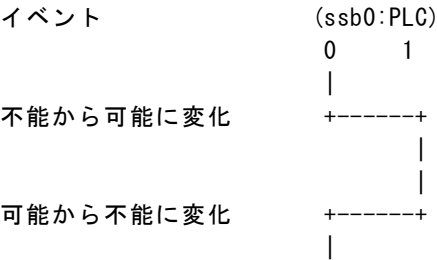
FE	0		
+-----+-----+-----+			
+-----+-----+			
+-----			DeiceID (0 - 0x7FFF)
+-----			Reserved (0)

  
+ 2 : SourceID (0 - 0xFFFF)  
+ 3 : TransacionID (0 - 0xFFFF)  
+ 4 - 19 : SECS メッセージ名称 (S1F2\_H、S6F11\_E00 等)  
SECS メッセージ定義ファイルに規定した名称
6. PLC への SECS Message 送信時の SECS Message 情報を格納するワードデバイス (20W) ... PLCMSDEV1 (msd1:PC)  
5. と同様の構成
7. SECS Message を構成する各項目値を格納するワードデバイス ..... 各 Message 毎に定義  
PLC から上位ホストへの SECS Message、あるいは上位ホストから PLC への SECS Message を構成する各メッセージ項目の値を格納するワードデバイスであり、SML 形式で指定するメッセージ定義ファイルに各メッセージ項目毎の値を格納する領域を定義する。

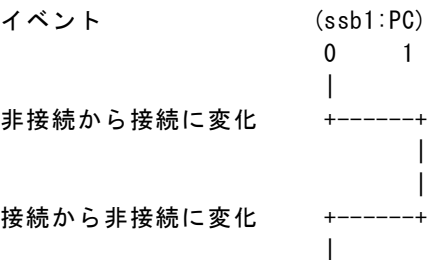
(b) タイミング・チャート

(注) 縦方向が時系列。(bss0:PLC)等は、(a)に示す PLC デバイス種別。

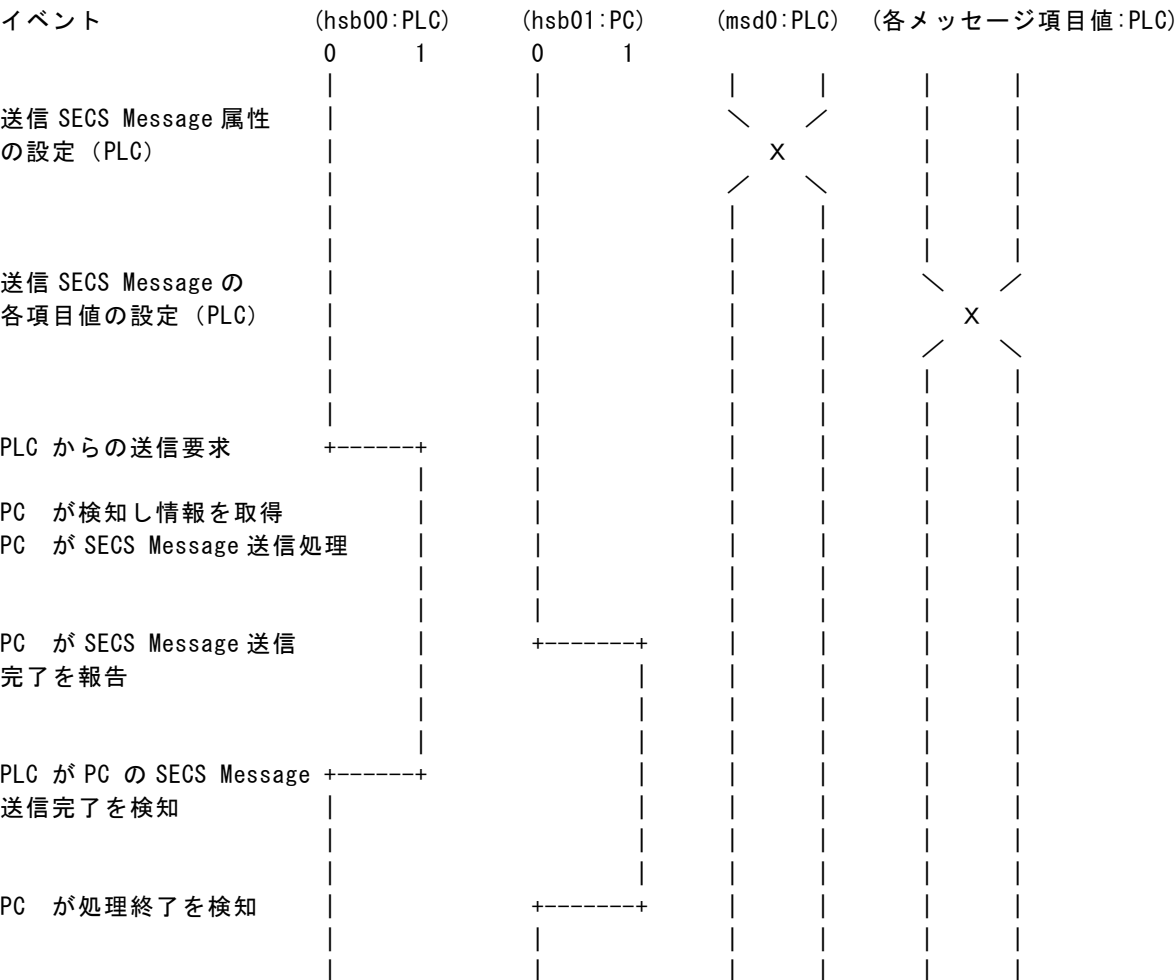
1. PLC SECS 通信処理状態



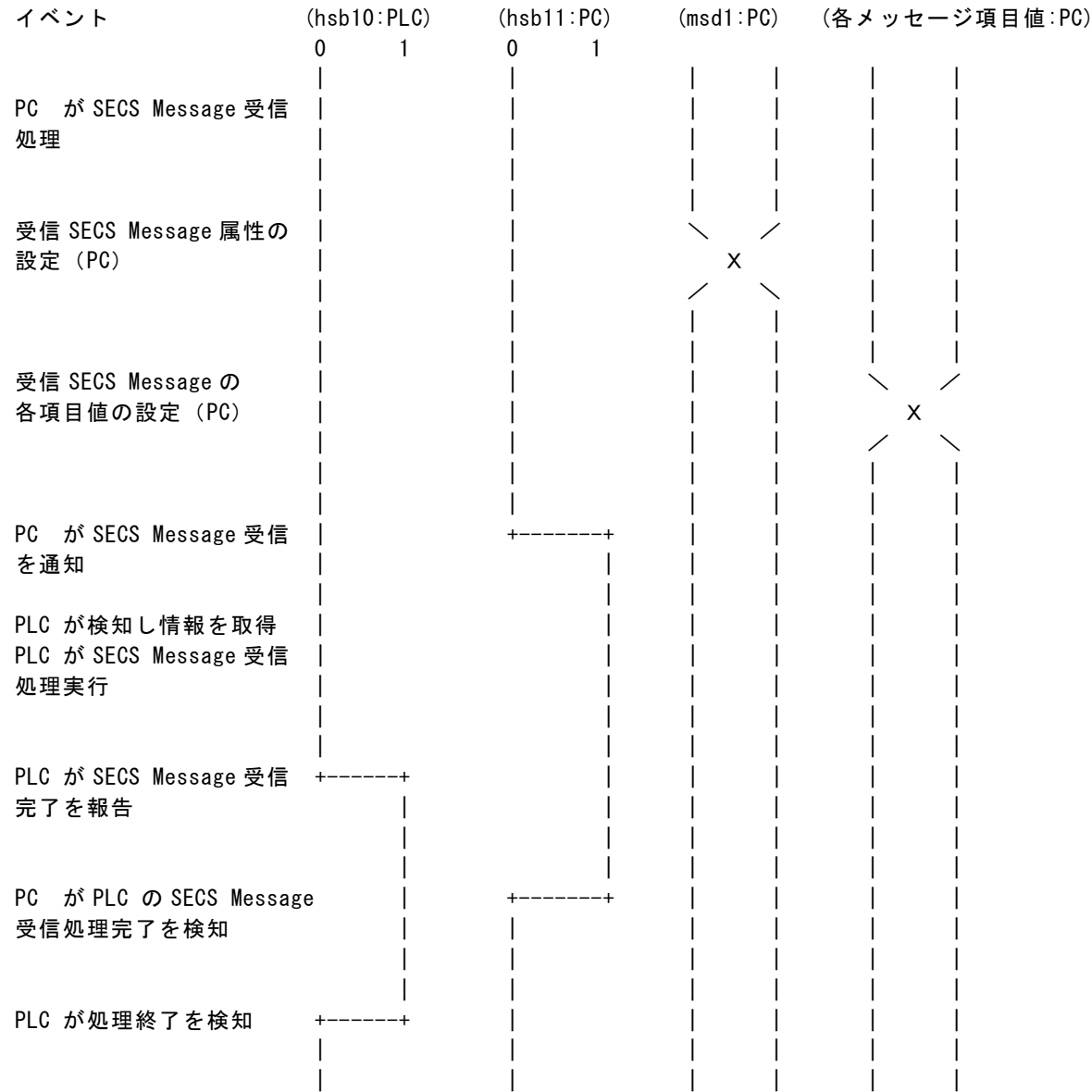
2. PC SECS 接続状態



3. PLC から PC へ SECS Message 送信処理



4. PC から PLC へ SECS Message 受信通知



(注 1) 本プログラムは、SECS Open Transaction には対応しません。SECS Message の送受信は、必ず、1つの1次メッセージに対する2次メッセージ応答を組みとしたトランザクションが完了してから、次のトランザクションを発生させる必要があります。

(c) 疑似 PLC 装置に関して

(i) 疑似 PLC 装置としての共有メモリ割り付け

Windows PC 内の共有メモリを、疑似 PLC 装置のメモリとして使用する。

- ・ 共有メモリのオブジェクト名称は、tdISPLC.ini [PLC] PLCHOST で指定する名称とする。
- ・ 共有メモリのサイズは、tdISPLC.ini [PLC] PLCPORT で指定するバイト・サイズとする。

・ 共有メモリの構成

```
typedef struct{
    unsigned char  ssb [ 2];          // Status    bit 値 (第0ビットのみ使用)
    unsigned char  dmy0 [ 2];
    unsigned char  hsb [ 2][ 2];      // Handshake bit 値 (第0ビットのみ使用)
    unsigned short msd [ 2][32];      // SECS Basic data
    unsigned short ssd [ 1];          // SECS Item  data 格納領域
} DevPseudo;
```

(注 1) 内容は、(2) (a) (ii) を参照すること。

(注 2) SECS Item data 格納領域のサイズは、tdISPLC.ini [PLC] PLCPORT で指定するバイト・サイズから 136Bytes (=2+2+4+2\*64) を引いた値となる。従って、PLCPORT には、SECS Item のデータ値を格納する領域サイズに 136Bytes 以上を加えた値を指定すること。  
また、ssd[0]は、格納領域として使用 (指定) できないことに留意すること。(2) (b) (注 3) 参照)

- ・ 即ち、C 言語の場合、以下のようなコードにて共有メモリを割り付ける。

```
HANDLE      sd;
DevPseudo  *md;
// key : tdISPLC.ini [PLC] PLCHOST で指定する名称
// sz   : tdISPLC.ini [PLC] PLCPORT で指定するバイト・サイズ
sd=CreateFileMapping(INVALID_HANDLE_VALUE, 0, PAGE_READWRITE, 0, sz, key);
md=(DevPseudo*)MapViewOfFile(sd, FILE_MAP_ALL_ACCESS, 0, 0, sz);
```

(ii) 共有メモリの構成

共有メモリの全体構成は、上記 (i) 参照

以下に、各項目値の意味を示す。

ssb0:PLC 等の表記は、(a)に示す PLC デバイス種別を示す。

```
ssb[ 0]      : ssb0:PLC
ssb[ 1]      : ssb1:PC
```

```
dmy0         : 未使用
```

```
hsb[ 0][ 0]  : hsb00:PLC
hsb[ 0][ 1]  : hsb01:PC
hsb[ 1][ 0]  : hsb10:PLC
hsb[ 1][ 1]  : hsb11:PC
```

```
msd[ 0]      : msd0:PLC  (msd[ 0][20]～[ 0][31] は未使用)
msd[ 1]      : msd1:PC   (msd[ 1][20]～[ 1][31] は未使用)
```

ssd[ 0]～ : SECS Message 定義ファイルで規定する位置に、SECS Message Item データ値を格納する。  
(2) (b) 参照。 0x0000 が ssd[0] を示す。

## (2) 設定ファイルの記述

本プログラムを起動する前に、必ず以下の2つのファイルを正しく設定し、用意する必要があります。

- ・ SECS/PLC 通信接続パラメータ設定ファイル ..... (tdISPLC.ini)
- ・ SECS Message 構成記述ファイル (SML 形式) ..... (tdISPLC.ini で指定)

### (a) SECS/PLC 通信接続パラメータ設定ファイル

- ・ 本ファイルは、通常 tdISPLC.ini という名称で、本プログラム (tdISPLC.exe) と同じフォルダに配置する必要があります。起動パラメータを指定することにより、任意のファイルを使用することができます。
- ・ tdISPLC.ini の設定方法は、弊社「SECS/HSMS 通信パッケージ (Trust Design Simple SECS Communication Library) (TDS)」に付属する「プログラマーズ・マニュアル (TDS.pdf)」の該当部 (2.1(1)) をご参照ください。SECS/HSMS 通信パッケージ (Free) は、弊社ホーム・ページ (<http://www.trust-design.co.jp/>) より、ダウンロードできます。
- ・ SECS 接続側の設定を [DEFAULT] 及び [SECS] セクションに記述し、PLC 接続側の設定を [PLC] セクションに記述します。  
同梱の tdISPLC.ini をご参照の上、同ファイルを修正する等してご使用ください。

### (i) [SECS] セクション

[DEFAULT] 及び [SECS] セクションの記述に関しては、特に以下の項目にご注意ください。  
(詳細は、上述の TDS.pdf 2.1(1) をご参照ください。)

- ・ SECSMODE ..... SECS 通信パラメータ
  - bit#0,1 .... 通信形式 (SECS-1 or HSMS-SS)
    - 4 ..... 装置 or ホスト
    - 5 ..... SECS 時 Master or Slave
    - 6 ..... HSMS 時 Active or Passive
- ・ DEVMODE ..... デバイス制御モード
  - bit#0 ..... デバイス I D チェックの有無
    - 1 ..... 受信待ち状態でない2次メッセージに対する処理
    - 4 ..... 1次メッセージ送信トランザクション管理
    - 8-12 ... S9Fx、Reject 自動送信の有無
    - 15 ..... T6T0 発生時の処理

PLC からホストへの1次メッセージ送信時のトランザクション管理を PLC 自身が行う場合は、(DEVMODE&0x10)!=0 として、tdISPLC ではデータ送信に関するトランザクション管理を行わない設定とする必要があります。この場合、(DEVMODE&0x0100)==0 としても、T3T0 発生時の S9F9 送信を tdISPLC は行いません。(そもそも、tdISPLC では T3T0 は発生しません。)従って、この場合の S9F9 送信も PLC から送信指示する必要がある事に留意する必要があります。

PLC からホストへの1次メッセージ送信時のトランザクション管理を PLC 自身では行わず、tdISPLC で行う場合は、(DEVMODE&0x10)==0 とします。この場合、tdISPLC で発生した T3T0 に関する情報は PLC には伝わらないので、PLC では独自に何らかの T0 処理を行わなければならない可能性があります。

<< 次ページに続く >>



<< 前ページから続く >>

- DEVID ..... 接続デバイス I D
- XDEV ..... 接続デバイス I D の最大数
- XMSGSIZE ..... 最大 SECS メッセージ・バイト長  
ある程度余裕を持った数値を指定してください。
  
- SDEVICE ..... SECS-1 接続時に使用する COM ポート名称 (“COM1” 等)
- HOST ..... HSMS-SS 接続時で Active 接続する場合の接続先ホスト名称 もしくは IP アドレス。
- PORT ..... HSMS-SS 接続に使用する TCP/IP ポート番号
- LINKINT ..... HSMS-SS 接続時のリンクテスト実行間隔  
リンクテストを実行する場合は、実行間隔の秒数を指定してください。
  
- TRCDIR ..... 通信トレース・ファイル格納フォルダ  
相対パスを指定する場合、tdlSPLC.ini ファイルの存在するフォルダが  
基点になります。通信トレース・ファイルのファイル名称に関しては  
TDS.pdf 2.1 (3) をご参照ください。
- TRCTTYPE ..... 通信トレースへの通信メッセージ出力形式 (TDS.pdf 2.1 (2) (c) 注記参照)
- TRCTOUT ..... 通信トレース出力モード
- TRCTLEVEL ..... 通信トレース出力レベル  
SECS-1 接続時、通信制御コードも出力する場合は、6 以上の値を指定  
してください。HSMS-SS 接続時、リンクテストに関するトレースも出力  
する場合は、9 or 10 以上の値を指定してください。

以下は、PLC との I/F において、各 SECS Message を構成する各項目値に関する情報を記述し、また通信トレース出力にも使用する、メッセージ定義ファイルに関する記述です。本 A P では、この定義が必ず必要です。

- MDMSSG ..... SECS 通信及び PLC 通信を行う SECS メッセージ構造を定義したファイル(.sml)  
を指定します。このファイルは、通信トレース出力にも使用します。
- MDMXITEM ..... データ項目総数の最大個数
- MDMXMSSG ..... 定義するメッセージの最大個数
- MDMXMITEM ..... データ項目総数の最大個数+メッセージ展開時の最大項目数
- MDMXPOOL ..... メッセージ定義 設定データ格納領域サイズ  
これらの項目は、ある程度余裕を持った数値を指定してください。

(ii) [PLC] セクション

PLC との接続条件、PLC 通信を行うための情報格納位置（各 SECS メッセージ項目の格納位置を除く）等を定義します。（各 SECS メッセージ項目の PLC デバイス上の格納位置は、MDMSSG に示すメッセージ定義ファイルに定義します。）

# 以下は PLC 通信トレース・ファイルに関する設定です。  
 # 設定内容は、TRCXDATA を除き SECS 通信トレースと同様であり、本セクションに指定しない場合  
 # は [DEFAULT] での記述が有効となります。  
 # 詳細は、上述の TDS. pdf の 2.1(1)(c) を参照してください。  
 # PLC 通信ログ・ファイルは、TRCDIR に指定するディレクトリに “tp0000t.trc” といったファイル  
 # 名称で作成します。  
 # TRCXDATA は、PLC 通信ログとして出力する PLC メモリの最大個数。無指定の場合 及び 0 を指定  
 # した場合は、全てを出力します。（ただし、中間行で、その全てが 0 の場合は出力しません。）

```
TRCDIR    = "."           // PLC 通信トレース・ファイル格納位置
TRCTTYPE  = 0x0008       // PLC 通信トレース出力形式
TRCTOUT   = 0x0542       // PLC 通信トレース出力モード
TRCTLEVEL = 3            // PLC 通信トレース出力レベル
                        // 3 以上を指定すると、取得値、設定値を出力します。
                        // 5 以上を指定すると、Handshake bit の ON/OFF を出力します。
TRCTATTR  = 0x810f       // PLC 通信トレース出力属性
TRCTSIZE  = 5000000      // PLC 通信トレース・ファイルのバイト・サイズ
TRCXDATA  = 0            // PLC 通信トレース出力最大メモリ個数
```

# 以降は PLC 接続条件等の設定です。  
 # 本アプリケーション固有の設定であり、必ず [PLC] セクションに指定してください。

```
PLCINTER  = "5000, 100"  // PLC 通信インターバル      (ms)
                        // “接続要求間隔, メモリ監視間隔”
PLCTO     = "5000, 1000" // PLC 通信タイムアウト時間 (ms)
                        // “接続要求時, メモリ監視時”

PLCMODE   = "0x0000"     // PLC メモリのバイト・オーダー変換指示等
                        //      8 7      3210
                        // +---+---+---+---+ (0:変換しない 1:変換する)
                        //      | |      |||+-- BIN, LOG, I1, U1
                        //      | |      ||+--- I2, I4, I8, U2, U4, U8
                        //      | |      |+--- F4, F8
                        //      | |      +--- A, J, K
                        //      | |
                        //      | +----- Message Name
                        //      +----- BIN, LOG, I1, U1 の扱い
                        //      0: PLC 1W に 2Byte 格納
                        //      1: PLC 1W に 1Byte 格納
                        // bit#0~7: 各 SECS Message 項目及び Message Name (wms0+4, wms1+4
                        //      に格納) 毎に、PLC Word Device 格納時の Byte Order の
                        //      変換を指示します。
                        // bit#8 : 1Byte 項目を PLC Word Device にどのように格納するかを
                        //      示します。
```

<< 次ページに続く >>

<< 前ページから続く >>

# 以下の項目は、使用する PLC 装置毎の設定となります。

```
PLCTYPE   = "15"           // 接続 PLC タイプ
                                // = 0: MELSEC (3E or 4E)
                                //   2: OMRON  (FINS)
                                //   4: KEYENCE (TCP/IP)
                                //  15: 疑似 PLC (PC 内共有メモリを使用)

PLCOPT0    = "0"           // PLC 接続オプション
                                // PLCTYPE= 0 (MELSEC) の場合
                                // = 0: 3E   =1: 4E
                                // PLCTYPE= 2 (OMRON) の場合
                                // = My node address
                                // PLCTYPE= 4 (KEYENCE) の場合
                                // = 0: KV-LE21V
                                // PLCTYPE=15 (疑似 PLC) の場合
                                // = 0: 固定

#PLCHOST    = "192.168.1.1" // PLC 接続ホスト IP アドレス
PLCHOST     = "tdISPLC00"   // PLCTYPE=15 の場合、共有メモリの識別名称

#PLCPORT    = "9600"        // PLC 接続 TCP ポート番号
PLCPORT     = "0x4200"      // PLCTYPE=15 の場合、共有メモリのバイト・サイズ
```

<< 次ページに続く >>

<< 前ページから続く>>

```
# 以下は、PLC との通信に使用するメモリ割付に関する記述です。
#
# PLC と SECS メッセージ交換を行う以下基本項目を設定するメモリ領域を指定します。
# 各 SECS メッセージに含む各項目値の PLC メモリ上の格納位置は、メッセージ定義ファイル
# (SML 形式での指定) にて指定します。
#
# PLCSSDEV0 : PLC 稼動状態を PLC が設定するビット・デバイス領域を示します ..... (ssb0)
# PLCSSDEV1 : SECS 稼動状態を PC が設定するビット・デバイス領域を示します ..... (ssb1)
#           "デバイス・コード, アドレス" で指定します。
#
# PLCHSDEV0 : PLC からの SECS Message 送信時のハンドシェイク用ビット・デバイス .. (hsb00, 01)
# PLCHSDEV1 : SECS からの SECS Message 送信時のハンドシェイク用ビット・デバイス .. (hsb10, 11)
#           "デバイス・コード, PLC 要求アドレス, PC 応答アドレス" で指定します。
#
# PLCMSDEV0 : PLC からの送信時の SECS Message 情報を格納するワード・デバイス .... (msd0)
# PLCMSDEV1 : SECS からの送信時の SECS Message 情報を格納するワード・デバイス .... (msd1)
#           "デバイス・コード, アドレス" で指定します。
#
# (注) SECS Message 情報は、20W を使用し、以下の構成とします。
#
#   msd0|msd1 + 0      : SF-Code 及び W-Bit
#                       FE      8 7      0
#                       +-----+-----+-----+
#                       |+-----+ +-----+
#                       | |           +----- F-Code (0 - 127)
#                       | +----- S-Code (0 - 63)
#                       +----- W-Bit
#
#   + 1      : DeviceID (0 - 0x7FFF)
#             FE      0
#             +-----+-----+-----+
#             |+-----+-----+
#             |           +----- DeviceID
#             +----- Reserved (0)
#             (注) PLC からの送信時、PLC からの DeviceID 指定値が =0 の場合は
#                   PC にて、本ファイルの DEVID に指定した値を付与します。
#
#   + 2      : SourceID (0 - 0xFFFF)
#   + 3      : TransactionID (0 - 0xFFFF)
#   + 4 - 19 : メッセージ名称 (S1F2_H、S6F11_E00 等)
#             メッセージ定義ファイルで規定した名称
```

<< 次ページに続く>>

<< 前ページから続く >>

```
# PLCTYPE=0 (MELSEC) の場合 -----
# ビット・デバイス・コード : 0x90 : 内部リレー
#                               0x92 : ラッチ・リレー
#                               0x94 : エッジ・リレー
#                               0x9c : 入力リレー
#                               0x9d : 出力リレー
#                               0xa0 : リンク・リレー
# ワード・デバイス・コード : 0xa8 : データ・レジスタ
#                               0xb4 : リンク・レジスタ
PLCSSDEV0 = "0xa0, 0x1002"           // PLC Status
PLCSSDEV1 = "0xa0, 0x1012"           // SECS Status
PLCHSDEV0 = "0xa0, 0x1000, 0x1010"    // PLC --> PC Handshake
PLCHSDEV1 = "0xa0, 0x1001, 0x1011"    // PLC <-- PC Handshake
PLCMSDEV0 = "0xa8, 0x1008"           // PLC --> PC Data
PLCMSDEV1 = "0xa8, 0x2008"           // PLC <-- PC Data
```

```
# PLCTYPE=2 (OMRON) の場合 -----
# ビット・デバイス・コード : 0x02 : データ・メモリ
#                               0x20 : 拡張データ・メモリ バンク #0
#                               :           :
#                               0x2c : 拡張データ・メモリ バンク #C
#                               0x30 : チャンネル I/O
#                               0x31 : 内部補助リレー
#                               0x32 : 保持リレー
#                               0x33 : 特殊補助リレー
# ワード・デバイス・コード : 0x82 : データ・メモリ
#                               0xa0 : 拡張データ・メモリ バンク #0
#                               :           :
#                               0xac : 拡張データ・メモリ バンク #C
# (注) ビット・デバイスの指定は "ワード・アドレス:ビット・アドレス" 形式で指定します。
PLCSSDEV0 = "0x02, 0x1000:2"         // PLC Status
PLCSSDEV1 = "0x02, 0x1001:2"         // SECS Status
PLCHSDEV0 = "0x02, 0x1000:0, 0x1001:0" // PLC --> PC Handshake
PLCHSDEV1 = "0x02, 0x1000:1, 0x1001:1" // PLC <-- PC Handshake
PLCMSDEV0 = "0x82, 0x1008"           // PLC --> PC Data
PLCMSDEV1 = "0x82, 0x2008"           // PLC <-- PC Data
```

<< 次ページに続く >>

<< 前ページから続く >>

```
# PLCTYPE=4 (KEYENCE) の場合 -----
# ビット・デバイス・コード : R      : リレー
#                               B      : リンク・リレー
#                               MR     : 内部補助リレー
#                               LR     : ラッチ・リレー
#                               CR     : コントロール・リレー
#                               VB     : ワーク・リレー
# ワード・デバイス・コード : DM     : データ・メモリ
#                               EM     : 拡張データ・メモリ
#                               CM     : コントロール・メモリ
#                               WM     : ワーク・メモリ
PLCSSDEV0 = "B, 0x1002"           // PLC  Status
PLCSSDEV1 = "B, 0x1012"           // SECS Status
PLCHSDEV0 = "B, 0x1000, 0x1010"   // PLC --> PC Handshake
PLCHSDEV1 = "B, 0x1001, 0x1011"   // PLC <-- PC Handshake
PLCMSDEV0 = "DM, 0x1008"          // PLC --> PC Data
PLCMSDEV1 = "DM, 0x2008"          // PLC <-- PC Data
```

```
# PLCTYPE=15 (Pseudo PLC) の場合 -----
# 以下に固定 (変更不可)
PLCSSDEV0 = "0x00, 0x0000"        // PLC  Status
PLCSSDEV1 = "0x00, 0x0001"        // SECS Status
PLCHSDEV0 = "0x01, 0x0000, 0x0001" // PLC --> PC Handshake
PLCHSDEV1 = "0x01, 0x0002, 0x0003" // PLC <-- PC Handshake
PLCMSDEV0 = "0x02, 0x0000"        // PLC --> PC Data
PLCMSDEV1 = "0x02, 0x0020"        // PLC <-- PC Data
```

(b) SECS Message 構成記述ファイル

各 SECS メッセージに含む各項目値の PLC メモリ上の格納位置を、(a) SECS/PLC 通信接続パラメータ設定ファイル (tdlSPLC.ini) の MDMSSG に示すメッセージ定義ファイル (SML 形式での指定) に指定します。

PLC が使用する または 設定する必要がある SECS メッセージ項目の定義行に、"/"で指定する行コメントとして、(行コメントの先頭)に、以下の項目を定義することにより、その項目値を (1) (b) 3.、4. の「各項目値の設定」で格納、保持する PLC メモリ領域を指定します。

“デバイス・コード, アドレス 1, アドレス 2, 個数”

- ・ デバイス・コード : (a) の PLCMSDEV0、PLCMSDEV1 の説明に示す、各 PLC 種別毎のワード・デバイス・コード
- ・ アドレス 1 : PLC → SECS (PC) 時に使用するアドレス
- ・ アドレス 2 : SECS (PC) → PLC 時に使用するアドレス
- ・ 個数 : 使用するワード・デバイス個数 (即ち、ワード数)

(注 1) SECS メッセージ項目定義のあと 1 つ以上のホワイト・スペースを空けて “/” を記述してコメントであることを示し、その後に上記 4 項目を ‘,’ で区切って、空白を空けずに指定してください。  
また、上記 4 項目の全体を、必ず「2 重引用符」(“) で括ってください。  
SECS メッセージ定義行には、PLC アドレス指定以外のコメントを記述しないでください。

(注 2) PLC の受信時において、PLC が必要としない項目、PLC からの送信時に PLC が設定する必要が無い項目に関しては、本指定を行う必要はありません。本指定を行わない場合は、PLC の受信時においては、その項目を無視し、PLC からの送信時には、その項目値は、メッセージ定義ファイルで指定した項目デフォルト値となります。

(注 3) アドレス値として =0 は使用できません。  
=0 を指定した場合、その項目に関する処理は行いません。

(注 4) PLCTYPE=15 (Pseudo PLC) の場合、デバイス・コードは 0x03 固定です。他の値を使用することはできません。

(注 5) リスト項目個数を不定にすることはできません。リスト項目数 (L[n]の n) は、必ず確定数を指定してください。

(例 1) PLCTYPE=0 (MELSEC) の場合

```

S1F2_E      E
<L[2]      // LIST 項目数は、確定でなければなりません。
           // コメントは、メッセージ項目指定行以外の行に
           // 記述してください。
<A[6]MDLN "EQUIP1">      // "0xa8, 0x1020, 0x0000, 3"
<A[6]SOFTREV "01.000">   // "0xa8, 0x1023, 0x0000, 3"
>

S2F49_H      W
<L[3]
<A[6]RCMD "LOAD">      // "0xa8, 0x0000, 0x21a0, 3"
<B[1]STID 0x01>      // "0xa8, 0x0000, 0x21a3, 1"
<A[16]LOTID "">      // "0xa8, 0x0000, 0x21a4, 8"
>

```

### 3. 操作説明

#### (1) 起動

インストールしたフォルダにある `tdISPLC.exe`（もしくは `tdISPLC.exe` へのショートカット）をダブル・クリック等により起動します。

（参考）起動オプションとして、以下を指定可能です。

<code>+i ini_file</code>	: 設定ファイルの名称 省略時 : <code>tdISPLC.ini</code>
<code>+S style_file</code>	: 動作スタイル（位置、サイズ、指定パラメータ）を保存するファイルの名称 省略時 : <code>tdISPLCWin.ini</code>
<code>+s SECS_Section</code>	: SECS 設定のセクション名称 省略時 : <code>SECS</code>
<code>+p PLC_Section</code>	: PLC 設定のセクション名称 省略時 : <code>PLC</code>

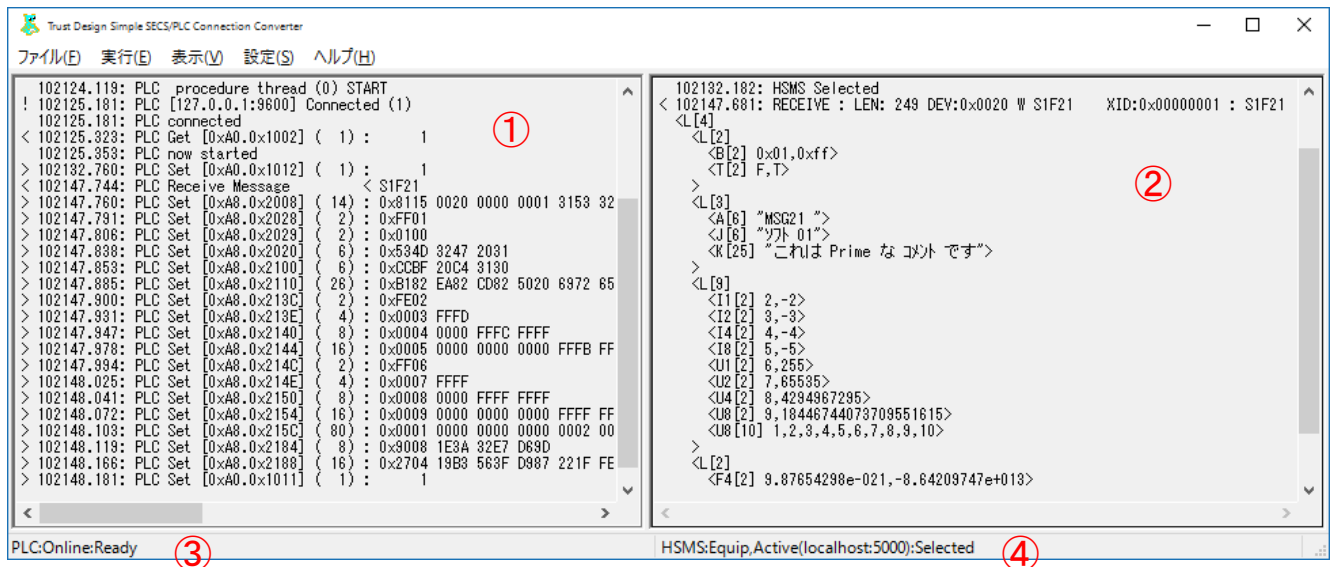
[注意] -----+

	本プログラムは、ライセンス管理用として、UDP/IP の以下のポートを使用します。また、UDP/Multicast	
	アドレスとして、以下のクラスDアドレスを使用します。ご使用になるコンピュータのファイアウォール等	
	により、これらをブロックしないよう設定してください。	
	・ 36277/udp	
	・ 239.254.200.75	

-----+



## (2) 画面操作説明



- ① : PLC 接続側通信トレース表示 . . . . . PLC との通信内容のトレース表示を行います。
- ② : SECS 接続側通信トレース表示 . . . . . SECS 通信メッセージのトレース表示を行います。
- ③ : PLC 接続側状態 . . . . . PLC I/F の現在接続状態を表示します。
- ④ : SECS 接続側状態 . . . . . SECS I/F の現在接続状態を表示します。

(a) メニュー

(a-1) [ファイル]

- ・ アプリケーションの終了 ..... tdISPLC を終了します。

(a-2) [実行]

- ・ 開始 ..... SECS/PLC 通信接続処理を開始します。
- ・ 終了 ..... SECS/PLC 通信接続処理を終了します。

(注 1) 相手側との接続が確立していない等の場合、[終了] 処理に時間がかかる場合があります。  
そのような場合に、即座にプログラムを終了した場合は、タイトルバー右端の [X] により、終了してください。

(a-3) [表示]

- ・ 通信トレース表示クリア ..... 通信トレースウインド (①、②) をクリアします。
- ・ 通信トレース最終行表示 ..... 通信トレースウインド (①、②) のスクロール・バーを、最終行が表示されている状態にします。  
(注意) 高速で自動スクロールしている状態等で、通常のスクロールバーの操作では、なかなか最終行が表示された状態にならない場合に使用します。  
最終行が表示されている状態の時には、自動スクロールします。
- ・ SECS List 表示 ..... SECS 通信トレースウインド (②) の送受信 SECS メッセージを、指定の List 形式で表示します。
- ・ SECS Hexa 表示 ..... SECS 通信トレースウインド (②) の送受信 SECS メッセージを、16 進数形式で表示します。
- ・ PLC 詳細表示 ..... PLC 通信トレースウインド (①) に、通信 SECS メッセージ情報を 16 進数形式及び ASCII 文字として表示します。
- ・ PLC H S 表示 ..... PLC 通信トレースウインド (①) に、PLC との通信 Handshake bit の I/O 内容を表示します。
- ・ PLC 冗長表示 ..... PLC 通信トレースウインド (①) への通信データ表示を、PLC 詳細表示、PLC H S 表示が ON の場合、それぞれの表示内容に関して、アイドル状態での通信も含めて、全ての通信に関して表示します。  
この指定をしていない場合は、ビット・デバイスの I/O に関して、状態変化があった場合にのみ表示します。

(注 1) 各表示状態の指示は、SECS 通信、PLC 通信ともに、通信トレース・ファイルへの出力には影響しません。通信トレース・ファイルへの出力は、設定ファイルの TRCTTYPE、TRCTLEVEL 等により決定します。

(a-4) [設定]

- ・ 自動開始 ..... プログラム起動後、自動的に処理を開始する設定とします。  
本設定は、次のプログラム起動時に有効となります。
- ・ 位置サイズ保存 ..... プログラム終了時に、本プログラムの終了時点での位置、サイズ等を保存し、次回起動時にウインドの状態を同じ状態に復元します。

(a-5) [ヘルプ]

- ・ バージョン情報 ..... 本プログラムのバージョン情報を表示します。

(b) PLC 通信トレース表示

本プログラムで発生する全体に関わるイベント 及び PLC 通信処理内容のトレース表示を行います。

(注 1) [表示] で、[PLC 詳細表示]、[PLC H S 表示] をチェックしてない状態では、送受信 SECS メッセージ名称のみを表示します。チェックしている状態では、(a-3)に示す PLC 通信内容の表示を行います。

(注 2) トレース・ファイルへの出力は、[表示] での指定に関わらず、TRCLEVEL 値が 3 以上である場合に、PLC 通信データ内容の 16 進数表示を行います。5 以上である場合は、PLC とのハンドシェイクを含め、ビット・デバイスの I/O も表示します。ただし、アイドル状態での通信内容の出力は行いません。2. (2) (b) (ii) TRCLEVEL をご参照ください。

(c) SECS 通信トレース表示

送受信した SECS 通信メッセージのトレース表示を行います。

[表示] で、[SECS List 表示]、[SECS Hexa 表示] 共にチェックをしない状態では、送受信ヘッダの情報のみを表示します。

(注 1) SECS 通信メッセージのリスト形式表示の形態は、tdISPLC.ini ファイルの [SECS] セクションにおける以下のパラメータで決まります。

- ・ TRCTTYPE ..... 通信トレースへの通信メッセージ出力形式
  - bit#2 ..... 項目データ表示形式
    - =0: 各項目を 1 行のみで表示し、1 行に納まらない場合は、後部を省略する。
    - 1: 各項目を複数行で表示し、数値項目は 1 行に 20 データ、文字列項目は 100Bytes 分を表示する。
  - bit#4,5,6 .. リスト出力形式 (通常 =2 としてご使用ください。)
    - =0: TDS オリジナル形式
    - 2: SML 形式
  - bit#7 ..... データ項目名表示 (メッセージ定義を使用しない場合は =0 としてください。)
  - bit#8,9 ..... メッセージ定義ファイルの形式
    - (本プログラムでは、必ず =0 として SML 形式のメッセージ定義ファイルをご使用ください。)
    - =0: SML 形式

(注 2) SECS 通信メッセージの 16 進数表示の形態は、tdISPLC.ini ファイルの [SECS] セクションにおける以下のパラメータで決まります。

- ・ TRCTTYPE ..... 通信トレースへの通信メッセージ出力形式
  - bit#3 ..... 16 進数表示形式
    - =0: 1 行に 16Bytes 表示する。
    - 1: 1 行に 20Bytes 表示する。

(注 3) SECS-1 接続時の通信制御コードの送受信は、通信トレース表示ウインドには表示しません。通信トレース・ファイルには (tdISPLC.ini の TRCTLEVEL>=6 の場合) 出力します。

(注 4) HSMS-SS 接続時の LinkTest メッセージは、通信トレース表示ウインドには表示しません。通信トレース・ファイルには (tdISPLC.ini の TRCTLEVEL>=9 の場合) 出力します。

(注 5) 通信トレースのファイルへの出力は、tdISPLC.ini の TRCDIR、TRCTTYPE、TRCTOUT、TRCTLEVEL、TRCTSIZE に依存します。詳細は、前述の TDS.pdf 2.1 (1) をご参照ください。