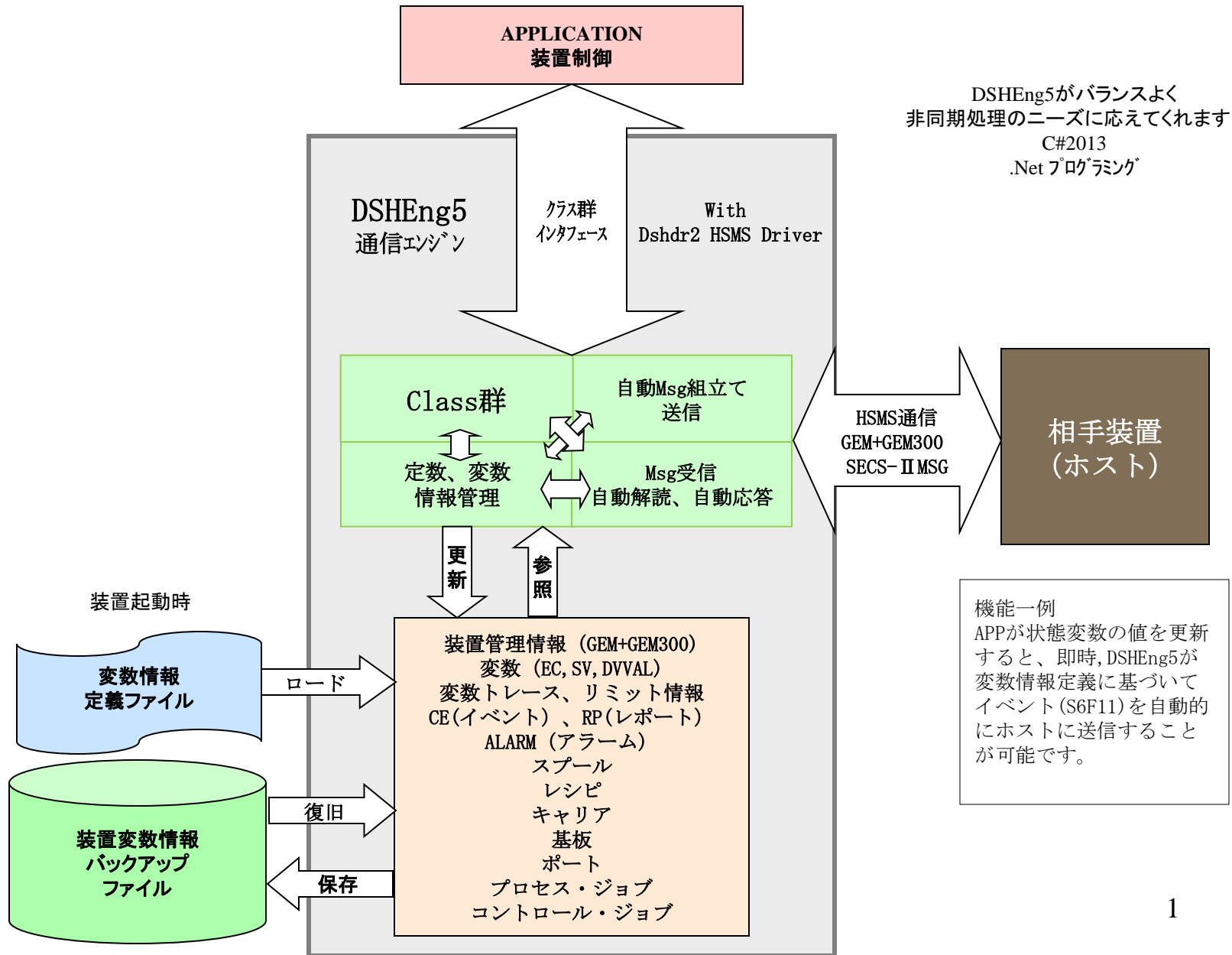


DSHEng5通信エンジン 装置 デモプログラムの紹介

(2019年8月)

きめ細くGEM機能の確認ができます。

文書番号 DSHEng5-19-30500-00



装置デモ・WPシミュレーションの画面例

formLog

```

***** DSHeng5 GEM Communication Engine Demo Program Start *****
----- 製品情報 -----
DSHeng5 Communication Engine : Trial type
S/N : 1524124494-1

----- File Time Stamp -----
Current Directory内
DSHeng5.dll      : 2019/08/21 14:05:16
dsHdr2.dll       : 2019/07/28 10:01:57
DSHeng5XSN.dll   : 2018/10/29 15:08:10
Eng5AppCsDemo.exe : 2019/08/21 15:00:22

* set_reserved_variable()
* set_reserved_variable() end ei=0
----- Equipment Started Successfully -----

----- Data Format -----
VID format = U4 44
ECID format = U4 44
SVID format = U4 44
DVID format = U4 44
ALID format = U4 44
RPID format = U4 44
CEID format = U4 44
DATAID format = U4 44
** rsp_sent_wait_time = 0 ms
----- Engine Started successfully -----

! enable callback() end_status = 0
! enable callback() end_status = 0
s6f11 ceid=15800 CE_ReadyToLoad
s6f11 ceid=15792 CE_LoadTransferBlocked
s6f11 ceid=15793 CE_LoadMaterialArrived
s6f11 ceid=15794 CE_LoadWaitingForHost
    
```

Log On/Off On 37 ファイル・リセット

DSHeng5-32Bit デモプログラム

通信接続状態: HSMS selected, GEM通信確立

エンジン制御: 情報復元 あり, 装置/体ス 装置

メッセージ送信モード: フロットモード(wait), S6F11表示指定 詳細(V), MD FMT U4

通信確立制御: 通信Enable, Enable取消し, 通信Disable

SxFy Ack設定, 製品情報, バックアップ情報 ALL, クラス情報参照, GC実行, BKUP確認

管理情報操作 / 関連メッセージ送信: EC装置定数, SV状態変数, DVデータ変数, V変数情報(全), SVトレース情報, 変数リミット情報, CEイベント情報, RPLポート情報, ALアラーム情報, PP情報(S7F3), FPP情報, RCPLレシ*情報, CJ情報, PRJ情報, CARキャリア情報, Substrate情報, Free Trid, S10F1, 3, 5, スプール情報, 1次Msg送信, S2F41送信, S2F49送信, S3F17送信, ポート制御

WP処理シナリオ操作: Load busy WP開始, 終了予約, Process busy WP停止, モニター画面, Unload busy 状況表示

HSMSDQモニター起動

WP Operation Monitoring

画面左下の WP処理開始 ボタンのクリックで始めます。

処理状態: ID照会完了, FOUP オープン待ち

処理開始, SMap照会, FOUP開, CarID照会, ロート終了, 搬出開始

半導体製造装置

PORT開始, PORT開始

CarrierID: CARID_01, Usage, RCPID, Location, Pr.JobID, C.JobID, エントロル: ONLINE, LOCAL, ONLINE, OFFLINE, REMOTE, LOCAL, アクセスモード: AUTO, 処理済み回数: 0, AUTO, MANUAL, サイクル時間: 0, アラーム, 端末メッセージ, Alarm, Term Msgリセット

```

#-----
# DSHDR2 Communication Definition File
# FileName : ¥DshEng5¥cnf¥comm_eq.def
#-----
START DSH
  MAX_MSG_SIZE      = x100040
  MAX_TRANSACTION   =1024
  LOG_FILE          = ¥dsheng5¥log¥DSHDR2. LOG
  LOG_MODE          = DAILY
  LOG_LINE          = 10000
  LOG_LIFE          = 6
  TIME_FORMAT       = "YYYY/MM/DD HH:NN:SS. CC"
  MON_PORT          = 9999
END
#-----
# PORT Definition
#-----
START PORT
  PORT              = 1
  PROTOCOL          = HSMS
  PORT_MODE         = ACTIVE
  TCP_PORT          = 6001
  IP                = 192.168.1.11
  T3                = 45
  T5                = 10
  T6                = 5
  T7                = 10
  T8                = 5
  LINKTEST          = 60
  S9F1              = 0
END
#-----
START PORT
  PORT              = 9
  PROTOCOL          = HSMS
  PORT_MODE         = PASSIVE
  TCP_PORT          = 5009
  T3                = 45
  T5                = 10
  T6                = 5
  T7                = 10
  T8                = 5
  LINKTEST          = 0
END

```

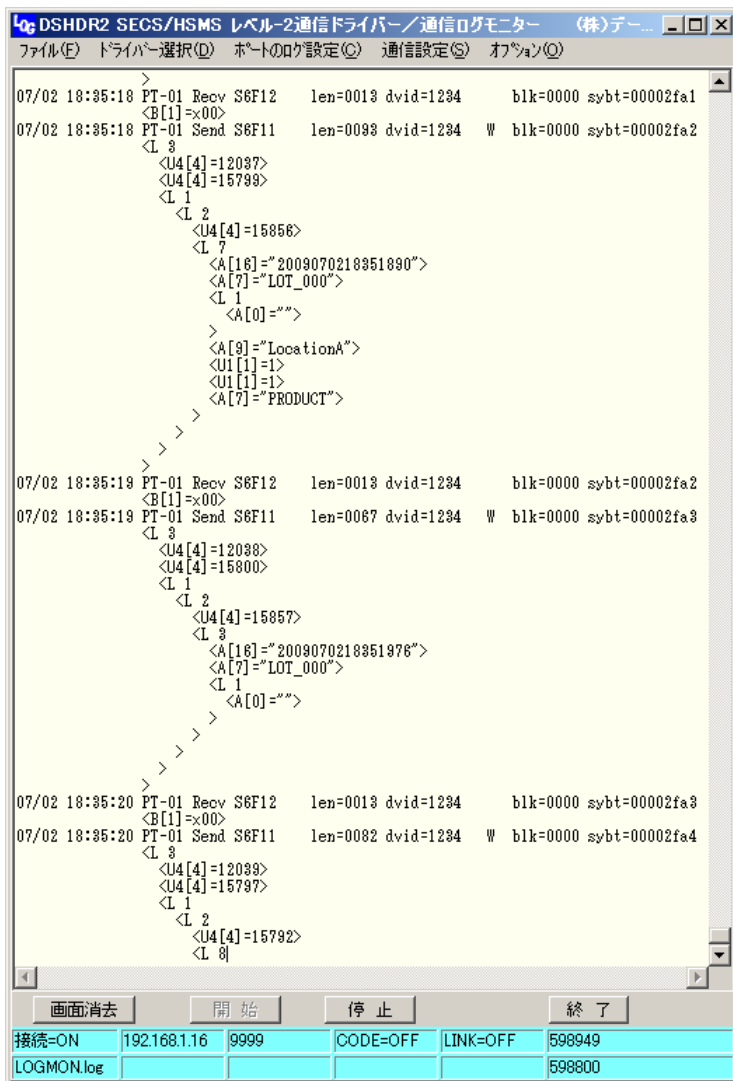
```

#-----
START PORT
  PORT              = 9
  PROTOCOL          = HSMS
  PORT_MODE         = PASSIVE
  TCP_PORT          = 5009
  T3                = 45
  T5                = 10
  T6                = 5
  T7                = 10
  T8                = 5
  LINKTEST          = 0
END
#-----
# DEVICE Definition
#-----
START DEVICE
  DEVICE            = 1
  DVID              = x1234
#   DVID            = x1236
  PORT              = 1
END
#-----
START DEVICE
  DEVICE            = 9
  DVID              = x5999
  PORT              = 9
END

```

これは装置側のものですが、
ホスト側のものは、comm_h.def
として準備されています。

装置デモ・ログモニター画面例



装置 - ホスト間のHSMS通信トランザクションをリアルタイムでモニタリングします。

装置、ホスト以外のPCからもモニタリング可能です。

(DSHDR2 HSMS通信ドライバーと接続しています。)

```
//----- EC - Equipment Constant -----
def_ec EC_MdlIn{
    ecid:    1        // =0x00000001
    format:  A[6..6]
    nominal: "DSH100"
}

def_ec EC_SoftRev{
    ecid:    2        // =0x00000002
    format:  A[6..6]
    nominal: "REV001"
}

def_ec EC_InitCommState{
    ecid:    4        // =0x00000004
    format:  U2[1]
    min:     "0"
    max:     "1"
    nominal: "0"
}

def_ec EC_InitControlState{
    ecid:    5        // =0x00000005
    format:  U2[1]
    min:     "0"
    max:     "1"
    nominal: "0"
}
```

```
//----- SV - Equipment Status Variable
def_sv SV_Clock{
    svid:    8192     // =0x00002000
    format:  A[16..16]
    nominal: ""
}

def_sv SV_CommunicationState{
    svid:    8193     // =0x00002001
    format:  U1[1]
    nominal: "0"
}

def_sv SV_ControlState{
    svid:    8194     // =0x00002002
    format:  U2[1]
    units:   "state"
    min:     "0"
    max:     "2"
    nominal: "0"
    event:   "CE_ControlState"
}

def_sv SV_CJExecName1{
    svid:           // =0x00002003
    format:  A[16..16]
    units:   "CJOB"
    nominal: ""
}
```

```
//----- DV - Data Variable -----
def_dv DV_StartTime{
    dvid:    8300     // =0x0000206c
    format:  A[19..19]
    nominal: ""
}

def_dv DV_EndTime{
    dvid:    8301     // =0x0000206d
    format:  A[19..19]
    nominal: ""
}

def_dv DV_Temp1{
    dvid:    8302     // =0x0000206e
    format:  A[8..8]
    units:   "Degree"
    nominal: ""
}
```

詳細はEQINFO.TXTファイルを
参照してください。

次頁にRP,CEを紹介します。

RP(Report), CE(CollectionEvent) の定義

```
//----- RP - Collection Event -----  
-----  
def_report RP_Communicating{  
    rptid:    10          // =0x0000000a  
    vname:    "SV_Clock"  // vid=8192  
}  
  
def_report RP_ControlState{  
    rptid:    100         // =0x00000064  
    vname:    "SV_Clock"  // vid=8192  
    vname:    "SV_ControlState" // vid=8194  
}  
  
def_report RP_PortAccessMode{  
    rptid:    1200        // =0x000004b0  
    vname:    "EC_PortAccessMode" // vid=256  
}  
  
def_report RP_Port1AccessMode{  
    rptid:    1201        // =0x000004b1  
    vname:    "EC_Port1AccessMode" // vid=257  
}  
  
def_report RP_Port2AccessMode{  
    rptid:    1202        // =0x000004b2  
    vname:    "EC_Port2AccessMode" // vid=258  
}
```

```
//----- CE - Collection Event -----  
-----  
-----def_ce CE_Communicating{  
    ceid:     2           // =0x00000002  
    enabled:  1  
    rptname:  "RP_Communicating" // rptid=10  
}  
  
def_ce CE_SpoolDeactivated{  
    ceid:     999         // =0x000003e7  
    enabled:  1  
}  
  
def_ce CE_ControlState{  
    ceid:     100         // =0x00000064  
    enabled:  1  
    rptname:  "RP_ControlState" // rptid=100  
}  
  
def_ce CE_AlarmOn{  
    ceid:     200         // =0x000000c8  
    enabled:  1  
}  
  
def_ce CE_AlarmOff{  
    ceid:     201         // =0x000000c9  
    enabled:  1  
}
```

一部分を紹介します。

DSHEng5は

これらの情報を内部に取り込み、RP、CE、変数間のリンク関係を生成し、管理します。

変数の値の設定・参照のためのサービスをAPI関数によって提供します。

CEイベントはclassSendS6F11クラスのSendS6F11メソッドを使って、CEIDを指定するだけでS6F11を送信してくれます。

次頁にAlarmを紹介します。

AL(Alarm) 情報の定義

```
//----- AL - Alarm-----
def_alarm AL_AlarmTempOver{
    alid:      1          // =0x00000001
    enabled:   1
    alcd:      "2"
    altx:      "Chamber Temperature Over"
    ce_on:     "CE_AlarmOn" // ceid=200
    ce_off:    "CE_AlarmOff" // ceid=201
}

def_alarm AL_AlarmPressure_1_Low{
    alid:      101        // =0x00000065
    enabled:   1
    alcd:      "2"
    altx:      "Chamber Pressure-1 Over"
    ce_on:     "CE_AlarmOn" // ceid=200
    ce_off:    "CE_AlarmOff" // ceid=201
}

def_alarm AL_AlarmPressure_2_Low{
    alid:      102        // =0x00000066
    enabled:   1
    alcd:      "2"
    altx:      "Chamber Pressure-2 Over"
    ce_on:     "CE_AlarmOn" // ceid=200
    ce_off:    "CE_AlarmOff" // ceid=201
}
```

DSHEng5は左で定義されるアラーム情報に従って

class_SendS5F1クラスのSendS5F1メソッドを使ってS5F1を送信します。

ALIDと発生/復帰の指定だけでS5F1を送信可能にします。

一部分を紹介します。

他に以下の情報を変数定義ファイル内に定義できます。(テスト用)

SPOOL
TRACE
RECIPE

EQINFO. txtファイルは

DshGemSet5.exe 編集プログラムを使って画面で編集できます。

詳細はEQINFO.TXTファイルを参照してください。

デモプログラムの起動は次頁です。

デモプログラムの起動とメイン画面



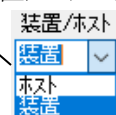
デモプログラムの開始

C : ¥DSHEng5¥bin¥Eng5AppCsDemo.exe

を起動します。

左の画面が表示されます。

装置/ホスト 通信サイト'を選択します。



通信を開始するために エンジン開始ボタンをクリックしてください。

(通信Enableの処理も含まれます。)

次頁のように装置通信が可能になります。

通信確立後のメイン画面

formLog

**** DSHEng5 GEM Communication Engine Demo Program Start ****

----- 製品情報 -----

DSHEng5 Communication Engine : Trial type
S/N : 1524124494-1

----- File Time Stamp -----
Current Directory内
DSHEng5.dll : 2019/08/21 14:05:16
dsbhdr2.dll : 2017/07/28 10:01:57
DSHEng5XSM.dll : 2018/10/29 15:08:10
Eng5AppCsDemo.exe : 2019/08/21 15:00:22

通信接続状態 エンジン制御

HSMS selected
 GEM通信確立

情報復元 あり

装置/ホスト 装置

メッセージ送信モード
ブロックモード(wait)

BKUP復元設定 エンジン開始

S6F11表示指定 詳細(V)

LOG表示 U4

エンジン停止

通信確立制御

通信Enable Enable取消し 通信Disable

SxFy Ack設定 製品情報 バックアップ情報 ALL

クラス情報参照 GC実行 BKUP確認

管理情報操作 / 関連メッセージ送信

EC装置定数 SV状態変数 DVデータ変数

V変数情報(全) SVトレース情報 変数リミット情報

CEイベント情報 RPLレポート情報 ALアラーム情報

PP情報(S7F3) FPP情報 RCPLレポート情報

CJ情報 PRJ情報

CARキャリア情報 Substrate情報 Free Trid

S10F1, 3, 5 スプール情報 1次Msg送信

S2F41送信 S2F49送信 S3F17送信 ホット制御

WP処理シナリオ操作

Load busy WP開始 終了予約

Process busy WP停止 モニター画面

Unload busy 状況表示

HSMSのモニター起動

消去 Log On/Off On 37 ファイルリセット

装置/ホストは自エンジンがどちら側の通信を行うか選択します。

エンジン開始のクリック後、相手ホストとの間で通信確立ができれば、通信状態が“確立済み”になります。(S1F13,14のやり取りの成功)

その後、以下の操作ができます。

各ボタンをクリックすると、それに対応する機能の操作画面が表示されます。

EC, SV, DVVALの設定・参照 など
TRACE, LIMIT, CE, REPORT, ALARM,
PP, RECIPE, CARRIER, SUBSTRATE,
CJ, PRJ などの情報の生成、設定、参照などの操作画面です。

Wafer処理開始/終了操作ボタン

装置定数情報操作画面

formEC

ECID選択

値入力

IDリスト表示 ID全情報取得

変数値設定 変数値取得

デフォルト値取得 変数名取得

最小値取得 データフォーマット取得

最大値取得 配列サイズ取得

値の妥当性チェック 物理単位名取得

配列データ取得 配列データ設定

配列サイズ変更 配列サイズ=0

リストサイズ* MD数

Listサイズ変更 Listサイズ=0

List vid1個設定 List全vid設定

情報一覧表

EC S2F13, S2F15, S2F29送信

ECIDリストReset IDリスト

ECIDリストAdd

EC要求

S2F13送信

EC送信

S2F15送信

一覧要求

S2F29送信

連続送信停止 連続送信開始

閉じる

formEC

ECID選択

値入力

IDリスト表示 ID全情報取得

変数値設定 変数値取得

デフォルト値取得 変数名取得

最小値取得 データフォーマット取得

最大値取得 配列サイズ取得

値の妥当性チェック 物理単位名取得

配列データ取得 配列データ設定

配列サイズ変更 配列サイズ=0

リストサイズ* MD数

Listサイズ変更 Listサイズ=0

List vid1個設定 List全vid設定

情報一覧表

EC S2F13, S2F15, S2F29送信

ECIDリストReset IDリスト

ECIDリストAdd

EC要求

S2F13送信

EC送信

S2F15送信

一覧要求

S2F29送信

連続送信停止 連続送信開始

閉じる

ECID選択用コンボボックスにシステムに登録されているEC変数のID)が表示されます。

- ID単位の操作
ECIDを選択し、操作したいボタンをクリックします。
- 変数値設定
変数値を設定する場合は、値入力欄に値を設定した後、このボタンをクリックします。
- 変数値取得
変数情報取得用のボタンクリックの場合は、取得値はメイン画面のログ部に表示されます。

SV、DVVALの操作画面については、それぞれの変数の操作になるだけで、基本的に本画面の操作と同じ操作になりますので、説明を省略します。

DSHEng5はclass_ECを介して変数IDをキーにして全変数の情報のアクセスが可能です。

値の設定、変更、情報（単位、データ範囲など）の参照が可能です。

表示がEnabled=falseになっているボタンはメインフォームでホストサイトが選択されるとEnabled=trueに変更され、操作できるようになります。

SVトレース(Trace)設定画面

Trace情報
TRID選択

DSPER(x 10ms) DSPER_TIME
TOTSMP REPGSZ
SV-COUNT

SVID選択

| index |
|--------------------------------|
| <input type="text" value="0"/> |

トレース条件設定S2F23

本画面はトレース機能を実行するための画面です。通信サイトがホストが選択された場合のもので、

S2F23メッセージに設定する項目データを入力設定します。

DSPER, TOTSMP, REPGSZ, SV数、トレースするSVIDを設定します。

その後、S2F23送信 ボタンをクリックしてトレース情報を装置側に送信します。

トレース結果 S6F1 メッセージを受信したら、それをログ画面に表示します。

変数リミット(Limit)情報設定画面

formLIMIT

VID選択 DV_Temp1_bin

設定

設定ID数 1

| LimitID | Lowerdb | Upperdb |
|---------|---------|---------|
| 10 | 10 | 90 |
| 11 | 20 | 80 |
| | | |

リミット情報設定 リミット情報取得 リミット情報削除

Enable LimitEvent Disable LimitEvent

Limit情報アクセスVIDリスト

VIDリスト リセット VIDリスト 追加

Limit定義 S2F45送信

一覧要求 S2F47送信

IDリスト DV_Temp1_bin

閉じる

本画面は変数リミット監視機能を実行するための画面です。通信サイトが**ホスト**に選択された場合のものです。

S2F45メッセージに設定する項目データを入力設定します。

その後、**S2F45送信** ボタンをクリックして変数リミット情報を装置側に送信します。

収集イベント(CE)情報操作画面

装置側の画面

formCE

CEID選択 更新

IDリスト表示 情報一覧表 CE名取得

イベント通知

イベント通知(S6F11) 周期ms 実施回数

連続送信開始 ID順番 回数 0:無制限

有効/無効設定 ON(1) 有効/無効取得

S2F37送信 個別

イベントレポート有効/無効

リンクRP取得 順位(0,1,2) リンクRP数取得

全リンクRP取得 CEコンテンツ取得

S6F15送信 イベントレポート要求

Link Event Report

S2F35で送信したいCEID数を設定し実行します。

CEIDリスト初期化 CEID数

画面左上のCEIDを選択してください。そして、CEID設定順位(0,1...)とRPID数を設定し実行します。

CEIDリンク初期化 リスト内CEID順位(0~) リンクするRPID数

以下、CEIDにリンクするRPIDを順に設定してください。

RPID選択

RPID追加登録

S2F35送信

S2F35を正常に送信できた場合、内部の情報を更新します。

S2F35全CEID送信

閉じる

左側が装置側、右側がホスト側の操作画面です。

ホスト側の画面

formCE

CEID選択 更新

IDリスト表示 情報一覧表 CE名取得

イベント通知

イベント通知(S6F11) 周期ms 実施回数

連続送信開始 回数 0:無制限

有効/無効設定 ON(1) 有効/無効取得

S2F37送信 個別

イベントレポート有効/無効

リンクRP取得 順位(0,1,2) リンクRP数取得

全リンクRP取得 CEコンテンツ取得

S6F15送信 イベントレポート要求

Link Event Report

S2F35で送信したいCEID数を設定し実行します。

CEIDリスト初期化 CEID数

画面左上のCEIDを選択してください。そして、CEID設定順位(0,1...)とRPID数を設定し実行します。

CEIDリンク初期化 リスト内CEID順位(0~) リンクするRPID数

以下、CEIDにリンクするRPIDを順に設定してください。

RPID選択

RPID追加登録

S2F35送信

S2F35を正常に送信できた場合、内部の情報を更新します。

S2F35全CEID送信

閉じる

レポート(RP)情報操作画面

装置側の画面

RPID選択 RP Communicating 更新

IDリスト表示 情報一覧表 RPコンテンツ取得

レポート名取得

リンクVID取得 順位(0,1,2.) 個別レポート要求

全リンクVID取得 0 S6F19送信

Define Report S2F33

S2F33で送信したいRPID数を設定し実行します。

設定するRPID数 1

新規RPIDリスト初期化 新規RPID 9990

既存RPIDリスト初期化

画面左上のRPIDを選択してください。そして、RPID設定順位(0,1,2.)とVD数を設定し実行します。

RPIDリンク初期化 リスト内RPID順位(0~0)

リンクするVID数 1

以下、RPID(リンクするVIDを順に設定してください。

VID追加登録 VID選択 EC_Mdln

S2F33送信

S2F33を正常に送信できた場合、内部のリンク情報を更新します。

S2F33全RPID削除

S2F33全RPID送信

閉じる

左側が装置側、右側がホスト側の操作画面です。

ホスト側の画面

RPID選択 RP Communicating 更新

IDリスト表示 情報一覧表 RPコンテンツ取得

レポート名取得

リンクVID取得 順位(0,1,2.) 個別レポート要求

全リンクVID取得 0 S6F19送信

Define Report S2F33

S2F33で送信したいRPID数を設定し実行します。

設定するRPID数 1

新規RPIDリスト初期化 新規RPID 9990

既存RPIDリスト初期化

画面左上のRPIDを選択してください。そして、RPID設定順位(0,1,2.)とVD数を設定し実行します。

RPIDリンク初期化 リスト内RPID順位(0~0)

リンクするVID数 1

以下、RPID(リンクするVIDを順に設定してください。

VID追加登録 VID選択 EC_Mdln

S2F33送信

S2F33を正常に送信できた場合、内部のリンク情報を更新します。

S2F33全RPID削除

S2F33全RPID送信

閉じる

アラーム(AL)情報操作画面

装置側の画面

ALID選択 AL_AlarmTempOver

ALCD 2

ALTXT "Chamber Temperature Over"

IDリスト表示 アラーム名取得

ALCD取得 ALTXT取得

情報一覧表

アラーム通知

アラーム通知(S5F1) 復旧/発生 発生(1)

連続送信開始

有効/無効設定 有効(1)

S5F3 有効/無効

有効/無効取得

対象ALID(個別/全部)

ON時CEID取得 個別ALID

OFF時CEID取得 S5F3送信

S5F5 Alarm List取得

要求ALIDリセット

要求ALID追加

S5F5送信

閉じる

左側が装置側、右側がホスト側の操作画面です。

ホスト側の画面

ALID選択 AL_AlarmTempOver

ALCD 2

ALTXT "Chamber Temperature Over"

IDリスト表示 アラーム名取得

ALCD取得 ALTXT取得

情報一覧表

アラーム通知

アラーム通知(S5F1) 復旧/発生 発生(1)

連続送信開始

有効/無効設定 有効(1)

S5F3 有効/無効

有効/無効取得

対象ALID(個別/全部)

ON時CEID取得 個別ALID

OFF時CEID取得 S5F3送信

S5F5 Alarm List取得

要求ALIDリセット

要求ALID追加

S5F5送信

閉じる

レシピ (RCP) 情報操作画面

The screenshot shows a software window titled 'formRCP' with a teal header. The interface is organized into several sections:

- RCPID選択**: A dropdown menu showing 'RCPID001' and an '更新' (Update) button.
- 情報アクセス**: A group of buttons including '新規ID割り当て' (highlighted with a blue border), 'IDリスト表示', 'RCP情報設定', 'RCP情報取得', 'RCPID削除', 'RCP State設定', a 'state' dropdown menu with 'SETUP' selected, and 'RCP State取得'.
- レシピパラメータ**: Input fields for 'param1' (rcpara1), 'paraval1' (54321), 'param2' (rcpara2), 'paraval2' (12345), and 'rcpbody' (rcpbody-A).
- メッセージ送信**: A section with three rows of controls:
 - Row 1: 'name action' (S15F9送信), 'rmnscmd' (Create), 'status request' (S15F9送信).
 - Row 2: 'rename request' (S15F5送信), 'new name' (RCP0012NEW), 'create' (S15F13送信).
 - Row 3: 'space request' (S15F7送信), 'rcpseccode' (1), 'retrieve request' (S15F17送信).
- Footer**: A dropdown menu with '1000', a 'S15F13連続送信' button, an empty input field, and a '閉じる' (Close) button.

装置、ホスト側は同じ画面を使用します。

コントロールジョブ(CJ)情報操作画面

ES formCJ

CjID選択 State

新規ID登録 IDリスト表示 State設定 State取得

CJ情報設定 CJ情報取得 S14F9送信 S14F11送信

CJID削除 S14F9連続送信

設定情報

objspec objtype

CarrierInputSpec CurrentPrJob

d collection plan

MtrlOutByStatus

Carid MtrlStatus

MtrlOutSpec Src

Src Carid Dst Carid

pause ceid

ProcessCtrlSpec

PrjobId

ControRule Name OutputRule Mstatus

value value

ProcessOrderMgmt JobStartMethod

PrJob State List

Prjid Prj State

CJコマンド S16F27

CJコマンド

para有無 cpname cpval

AccessCountクリア AccessCount取得 チェック 閉じる

左側が装置側、
右側がホスト側の
操作画面です。

ES formCJ

CjID選択 State

新規ID登録 IDリスト表示 State設定 State取得

CJ情報設定 CJ情報取得 S14F9送信 S14F11送信

CJID削除 S14F9連続送信

設定情報

objspec objtype

CarrierInputSpec CurrentPrJob

d collection plan

MtrlOutByStatus

Carid MtrlStatus

MtrlOutSpec Src

Src Carid Dst Carid

pause ceid

ProcessCtrlSpec

PrjobId

ControRule Name OutputRule Mstatus

value value

ProcessOrderMgmt JobStartMethod

PrJob State List

Prjid Prj State

CJコマンド S16F27

CJコマンド

para有無 cpname cpval

AccessCountクリア AccessCount取得 チェック 閉じる

プロセスジョブ (PRJ) 情報操作画面

装置側の画面

PrjID選択 PJ-0001

IDリスト表示
新規ID割り当て
PRJ情報設定
PRJ情報取得
PRJID削除
PRJ State設定
PRJ State取得

設定情報
mf CARRIER (13)
prrecipemethod 1
サイトID RCP-0001
prprocessstart 0
ceid CE_Communicating
State QUEUED

MF=13 Carrier
キャリア数 1
キャリアID CARID_01

MF=14 Material
MID数 25
MID MID-0

関連メッセージ送信
PRJ生成 S16F11送信
PRJ生成(Multi) S16F15送信
PRJ Deque S16F17送信
GetAllJob S16F19送信
Get Space S16F21送信

PRJコマンド S16F5
コマンド ABORT
パラメータ数 0個
ParaCmd para-c
ParaVal pc1122
S16F5送信

連続送信停止 連続送信開始 閉じる

左側が装置側、右側がホスト側の操作画面です。

ホスト側の画面

PrjID選択 PJ-0001

IDリスト表示
新規ID割り当て
PRJ情報設定
PRJ情報取得
PRJID削除
PRJ State設定
PRJ State取得

設定情報
mf CARRIER (13)
prrecipemethod 1
サイトID RCPID001
prprocessstart 0
ceid CE_Communicating
State QUEUED

MF=13 Carrier
キャリア数 1
キャリアID CARID_01

MF=14 Material
MID数 25
MID MID-0

関連メッセージ送信
PRJ生成 S16F11送信
PRJ生成(Multi) S16F15送信
PRJ Deque S16F17送信
GetAllJob S16F19送信
Get Space S16F21送信

PRJコマンド S16F5
コマンド ABORT
パラメータ数 0個
ParaCmd para-c
ParaVal pc1122
S16F5送信

連続送信停止 連続送信開始 閉じる

キャリア (Carrier) 情報操作画面

ES formCAR

CarID選択

設定情報

IdStatus MapStatus

AccStatus

Usage Location

Mid SubstId

SlotCount

新規ID登録 IDリスト表示 slot id

| | | | | |
|-------------|-------------|----|----|----|
| キャリア情報設定 | キャリア情報取得 | 10 | 20 | 30 |
| キャリアID削除 | | 11 | 21 | 31 |
| | | 12 | 22 | 32 |
| IdStatus設定 | IdStatus取得 | 13 | 23 | 33 |
| | | 14 | 24 | 34 |
| MapStatus設定 | MapStatus取得 | 15 | 25 | |
| | | 16 | 26 | |
| Location設定 | Location取得 | 17 | 27 | |
| | | 18 | 28 | |
| Usage設定 | Usage取得 | 19 | 29 | |
| | | | | |
| AccStatus設定 | AccStatus取得 | | | |
| | | | | |
| SlotMap設定 | SlotMap取得 | | | |

slot pos(1,2...25)

装置、ホスト側は同じ画面を使用します。
キャリア情報の設定・参照を行います。

基板(Substrate)情報操作画面

The screenshot shows the 'formSUBST' application window. At the top, there is a 'SubstID選択' dropdown menu with 'SUBSTID001' selected. Below this is a '設定情報' section with a grid of input fields and dropdown menus:

| | | | |
|-------------|----------|-----------|--------------|
| AcquiredID | SUBST100 | State | AtSource |
| LotID | LOT-111 | IDStatus | NotConfirmed |
| SubstLocID | LOC-2 | Material | State-0 |
| Source | LOC-0 | ProcState | NeedsProces |
| Destination | LOC-3 | LocState | Unoccupied |
| BatchLocID | BLOC-01 | Type | Wafer |
| PosInBatch | POS-001 | Usage | Product |

Below the settings are several columns of buttons for operations:

- Left Column:** 新規ID登録, Subst情報設定, Subst情報取得, SubstID削除, AcquiredId設定, AcquiredId取得, SubstLotId設定, SubstLotId取得, SubstLocId設定, SubstLocId取得, SourceLoc設定, SourceLoc取得, DestLoc設定, DestLoc取得, BatchLoc設定, BatchLoc取得.
- Second Column:** IDリスト表示, PosInBat設定, PosInBat取得, State設定, State取得, IdStatus設定, IdStatus取得, MaterialStat設定, MaterialStat取得, ProcStat設定, ProcStat取得, LocStat設定, LocStat取得, Type設定, Type取得, Usage設定, Usage取得.
- Right Column (Loc History):** locID (LOCID-1), TimeIn (2019082622182186), TimeOut (2019082622182186), TimeIn更新, TimeOut更新, LocHistリセット, LocHist追加, LocHist取得, HistCount取得.

At the bottom right, there is a '閉じる' button.

装置、ホスト側は同じ画面を使用します。

基板情報の設定、参照を行います。

応答メッセージ Ack 設定画面

受信した1次メッセージに対して応答するAckの値を設定します。

| | | | |
|--------------|----|----------------|------|
| S1F15 oflack | 通常 | S10F3 ackc10 | 通常 |
| S1F17 onlack | 3 | S10F5 ackc10 | 通常 |
| S2F23 rsack | 通常 | S14F9 objack | 通常 |
| S2F41 hcack | 通常 | S14F11 objack | 通常 |
| S2F43 rsack | 通常 | S14F19 svcack | 通常 |
| S2F45 lvack | 通常 | S14F21 dataack | 通常 |
| S2F49 hcack | 通常 | S15F3 rmack | 通常 |
| S3F17 caack | 通常 | S15F5 rmack | 通常 |
| S3F23 caack | 通常 | S15F13 rmack | 通常 |
| S3F25 caack | 通常 | S16F5 acka | True |
| S3F27 caack | 通常 | S16F11 acka | True |
| S3F35 rpmack | 通常 | S16F15 acka | True |
| S7F1 ackc7 | 通常 | S16F17 acka | True |
| S7F3 ackc7 | 通常 | S16F27 acka | True |
| S7F23 ackc7 | 通常 | | |

デフォルトに戻す 設定 Quit

本画面は、装置側が受信する1次メッセージに対して応答メッセージを相手に返しますが、そのときに返すAckアイテムの値を故意にエラーを示す値を返すようにするための画面です。

以下のように操作します。

1. Ackの値をエラーにしたいメッセージのコンボボックスの選択を変えて行います。
2. その後、設定ボタンをクリックします。これでOKです。

例えば、S1F17に対し、onlack = 3 を返すようにするためには、左の画面のようにS1F17のonlackを3に選択し、設定ボタンをクリックします。

ログモニターの通信ログを下に示します。

WPシミュレーション画面 - ホスト側 の開始 / 終了

画面左下の WP処理開始 ボタンのクリックで始めます。

処理状態 **処理中, 処理終了待ち**

処理開始 半導体製造装置 処理終了

SMap照合

FOUP開 CARID_01 FOUP閉

CarID照合

ポート終了 搬出開始

ポート-1 ポート-2

ポート開始 ポート開始

CarrierID CARID_01 Usage

RCPID Location

PrJobID CJobID

コントロール ONLINE, LOCAL

ONLINE OFFLINE REMOTE LOCAL

アクセスモード AUTO 処理済み回数 0

AUTO MANUAL サイクル時間 0

アラーム

端末メッセージ

Alarm, Term Msgリセット

WP処理は、ホスト側のPCWP処理を
開始してください。

WP処理シナリオ操作

- Load busy **WP開始** 終了予約
- Process busy **WP停止** モニター画面
- Unload busy 状況表示

Top画面の下の **WP開始** ボタンの
クリックで開始します。

開始後、装置側からのCEIDが
CE_LoadTransferBlockedのS6F11の
受信によってWP処理が開始します。

WPシミュレーション画面 – 装置側 の開始 / 終了

ES WP Operation Monitoring

画面左下の WP処理開始 ボタンのクリックで始めます。

処理状態 **処理中, 処理終了待ち**

処理開始 SMap照合 FOUF開 CarID照合 ロート終了

半導体製造装置

CARID_01

FOUF開

搬出開始

ポート-1 ポート-2

ロート開始

CarrierID CARID_01 Usage

RCPID Location

PrJobID CJobID

コントロール ONLINE, LOCAL

ONLINE OFFLINE REMOTE LOCAL

アクセスモード AUTO 処理済み回数 0

AUTO MANUAL サイクル時間 0

アラーム

端末メッセージ

Alarm, Term Msgリセット

WP処理は、ホスト側のPCでWPの処理を開始しておいてください。

WP処理シナリオ操作

- Load busy **WP開始** 終了予約
- Process busy **WP停止** モニター画面
- Unload busy 状況表示

Top画面の下の **WP開始** ボタンのクリックで開始します。

開始後、装置側からのCEIDが CE_LoadTransferBlockedのS6F11の受信によってWP処理が開始します。

WP処理シナリオ操作

- Load busy **WP開始** **終了予約**
- Process busy **WP停止** モニター画面
- Unload busy 状況表示

1 サイクル終了しあたら、次の CE_LoadTransferBlockedのS6F11の受信で処理が開始します。

HSMSロケモニター起動