



DSHEng4 装置通信エンジン (GEM+GEM300)
ソフトウェア・パッケージ

APP インタフェース
ライブラリ関数説明書
(C, C++, .Net-Vb,C#)

VOL- 8 / 1 5

3 . 1 4 キャリア情報アクセスサービス関数

2 0 0 9年6月

株式会社データマップ

文書番号 DSHEng4-09-30328-00



[取り扱い注意]

- ・ この資料ならびにソフトウェアの一部または全部を無断で使用、複製することはできません。
- ・ 本説明書に記述されている内容は予告なしで変更される可能性があります。
- ・ Windows は米国 Microsoft Corporation の登録商標です。
- ・ ユーザーが本ソフトウェアの使用によって生じた遺失履歴、(株)データマップの予見の有無を問わず発生した特別損害、付随的損害、間接損害およびその他の拡大損害に対して責任を負いません。

【改訂履歴】

番号	改訂日付	項目	概略
1.	2009.6	初版	

目次

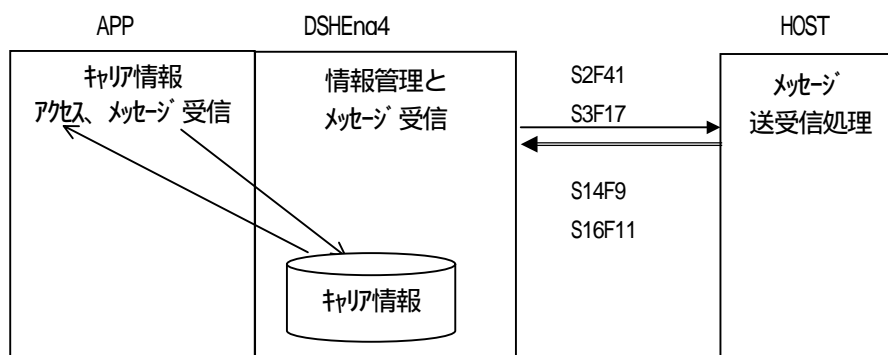
3.14 CAR キャリア情報アクセスサービス関数.....	1
3.14.1 使用する情報格納構造体.....	3
3.14.2 CAR キャリア情報アクセス関数.....	5
3.14.2.1 EngAllocCarInfo() - キャリアの登録.....	5
3.14.2.2 EngSetCarInfo() - キャリア情報の設定.....	6
. EngSetCarInfoX() - インデクスでのキャリア情報の設定.....	6
3.14.2.3 EngGetCarInfo() - キャリア情報の取得.....	7
. EngGetCarInfoX() - インデクス指定でのキャリア情報の取得.....	7
3.14.2.4 EngDelCarInfo() - キャリアの削除.....	9
. EngDelCarInfoX() - インデクスでのキャリアの削除.....	9
3.14.2.5 EngSetCarState() - キャリア情報状態の設定.....	10
. EngSetCarStateX() - インデクスでのキャリア情報状態の設定.....	10
3.14.2.6 EngGetCarState() - キャリア情報状態の取得.....	12
. EngGetCarStateX() - インデクスでのキャリア情報状態の取得.....	12
3.14.2.7 EngSetCarIdStatus() - キャリアID読取り状態の設定.....	14
. EngSetCarIdStatusX() - インデクスでのキャリアID読取り状態の設定.....	14
3.14.2.8 EngGetCarIdStatus() - キャリアID読取り状態の取得.....	16
. EngGetCarIdStatusX() - インデクスでのキャリアID読取り状態の取得.....	16
3.14.2.9 EngSetCarMapStatus() - スロットマップ状態の設定.....	18
. EngSetCarMapStatusX() - インデクスでのスロットマップ状態の設定.....	18
3.14.2.10 EngGetCarMapStatus() - スロットマップ状態の取得.....	20
. EngGetCarMapStatusX() - インデクスでのスロットマップ状態の取得.....	20
3.14.2.11 EngSetCarAccessStatus() - アクセス状態の設定.....	22
. EngSetCarAccessStatusX() - インデクスでのアクセス状態の設定.....	22
3.14.2.12 EngGetCarAccessStatus() - アクセス状態の取得.....	24
. EngGetCarAccessStatusX() - インデクスでのアクセス状態の取得.....	24
3.14.2.13 EngSetCarLocation() - ロケーションの設定.....	26
. EngSetCarLocationX() - インデクスでのロケーションの設定.....	26
3.14.2.14 EngGetCarLocation() - ロケーションの取得.....	27
. EngGetCarLocationX() - インデクスでのロケーションの取得.....	27
3.14.2.15 EngSetCarUsage() - Usageの設定.....	28
. EngSetCarUsageX() - インデクスでのUsageの設定.....	28
3.14.2.16 EngGetCarUsage() - Usageの取得.....	29
. EngGetCarUsageX() - インデクスでのUsageの取得.....	29
3.14.2.17 EngSetCarLotid() - Lotidの設定.....	30
. EngSetCarLotidX() - インデクスでのLotidの設定.....	30
3.14.2.18 EngGetCarLotid() - Lotidの取得.....	32
. EngGetCarLotidX() - インデクスでのLotidの取得.....	32
3.14.2.19 EngSetCarSubstid() - Substidの設定.....	34
. EngSetCarSubstidX() - インデクスでのSubstidの設定.....	34
3.14.2.20 EngGetCarSubstid() - Substidの取得.....	36
. EngGetCarSubstidX() - インデクスでのSubstidの取得.....	36
3.14.2.21 EngSetCarSlotmap() - Slotmapの設定.....	38
. EngSetCarSlotmapX() - インデクスでのSlotmapの設定.....	38
3.14.2.22 EngGetCarSlotmap() - Slotmapの取得.....	40
. EngGetCarSlotmapX() - インデクスでのSlotmapの取得.....	40

3 . 14 . 2 . 23	EngSetCarSlotCount() – スロット数の設定	42
.	EngSetCarSlotCountX() – インデクスでのスロット数の設定.....	42
3 . 14 . 2 . 24	EngGetCarSlotCount() – スロット数の取得.....	43
.	EngGetCarSlotCountX() – インデクスでのスロット数の取得.....	43
3 . 14 . 2 . 25	EngGetCarList() – 全登録キャリアID取得関数.....	44
3 . 14 . 2 . 26	EngGetCarId() – インデクスからCARID (キャリアID) の取得.....	45
3 . 14 . 2 . 27	EngGetCarIdIndex() – CARID (キャリアID) からインデクスの取得	46
3 . 14 . 3	CAR キャリア関連ライブラリ関数.....	47
3 . 14 . 3 . 1	DshFreeTCAR_INFO() - キャリア情報構造体メモリの開放.....	47
3 . 14 . 3 . 2	DshCopyTCAR_INFO() - キャリア情報構造体メモリのコピー	48
3 . 14 . 3 . 3	DshInitCarInfo() – TCAR_INFOの初期設定.....	49
3 . 14 . 3 . 4	DshInitCarSlotInfo() – キャリアスロット情報の初期設定	51
3 . 14 . 3 . 5	DshPutCarSlotInfo() – キャリア情報にスロット情報を設定.....	53
3 . 14 . 3 . 6	DshPutCarSlotInfoCopy() – キャリア情報にスロット情報をコピーして設定	54
3 . 14 . 3 . 7	DshInitCarContent() – キャリアコンテンツマップ情報の初期設定.....	55
3 . 14 . 3 . 8	DshPutCarContent() – キャリアコンテンツマップ情報の設定	56

(VOL - 9 に続く)

3.14 CAR キャリア情報アクセスサービス関数

ここで述べるキャリア情報は、DSHEng4 が管理します。従って、APP はこれらの情報をアクセスと関連メッセージを送信するために以下の DSHEng4 API 関数を使用します。



(1) 情報アクセスと送信 API 関数

キャリア情報のアクセスとホストへのメッセージ送信に関連するサービスのための API 関数名は一覧表のとおりです。

	API 関数名	機能
1	EngAllocCarInfo()	キャリア領域を割当て登録します。
2	EngSetCarInfo()	キャリア情報を設定・変更します。
3	EngGetCarInfo()	CARID 指定でキャリア情報を取得します。
4	EngGetCarInfoX()	キャリアインデックス指定でキャリア情報を取得します。
5	EngDelCarInfo()	CARID 指定でキャリア情報を削除します。
6	EngDelCarInfoX()	キャリアインデックス指定でキャリア情報を削除します。
7	EngGetCarId()	指定したキャリアインデックスの CARID を取得します。
8	EngGetCarIdIndex()	指定した CARID の情報インデックスを取得します。
9	EngSetCarIdStatus()	CARID 指定で CARID の状態を設定します。
10	EngSetCarIdStatusX()	キャリアインデックス指定で CARID の状態を設定します。
11	EngGetCarIdStatus()	CARID 指定で CARID の状態を取得します。
12	EngGetCarIdStatusX()	キャリアインデックス指定で CARID の状態を取得します。
13	EngSetCarMapStatus()	CARID 指定でキャリアマップの状態を設定します。
14	EngSetCarMapStatusX()	キャリアインデックス指定でキャリアマップの状態を設定します。
15	EngGetCarMapStatus()	CARID 指定でキャリアマップの状態を取得します。
16	EngGetCarMapStatusX()	キャリアインデックス指定でキャリアマップの状態を取得します。
17	EngSetCarLocation()	CARID 指定でキャリアロケーションを設定します。
18	EngSetCarLocationX()	キャリアインデックス指定でキャリアロケーションを設定します。
19	EngGetCarLocation()	CARID 指定でキャリアロケーションを取得します。
20	EngGetCarLocationX()	キャリアインデックス指定でキャリアロケーションを取得します。
21	EngGetCarList()	キャリア ID の一覧リストを取得します。

CARID インデクスは、DSHEng4 が管理する各 CARID 情報領域の番号です。このインデクスの値は、EngAllocCarInfo()関数実行時に DSHEng4 によって割当てられ、APP に渡されます。また、キャリアの取得時に、情報格納構造体のメンバー、index に設定されます。

(2) ライブラリ関数

他に APP が使用できるキャリア情報処理用 API 関数として、以下の関数があります。

	API 関数名	機能
1	DshFreeTCAR_INFO()	キャリア情報が格納されている TCAR_INFO 構造体の内部で使用されているメモリを開放するための関数です。
2	DshCopyTCAR_INFO()	TCAR_INFO のキャリア情報を別の構造体にコピーします。
3	DshInitCarInfo()	キャリア情報 TCAR_INFO 構造体の初期設定をします。
4	DshInitCarSlotInfo()	スロット情報 Tslot_INFO 構造体の初期設定をします。
5	DshPutCarSlotInfo()	TCAR_INFO 構造体内にスロット情報を 1 個追加設定します。
6	DshPutCarSlotInfoCopy()	キャリア情報にスロット情報をコピーして設定します。

(3) ユーザ作成ライブラリ関数

	ライブラリ関数名	機能
	なし	

3.14.1 使用する情報格納構造体

キャリア情報を操作する関数は、キャリア情報の格納のために TCAR_INFO 構造体を使用します。キャリア情報とそのアクションなどに関連する構造体は下記のとおりです。

(1) TCAR_INFO Carrier Information

```
typedef struct{
    int      index;           // 登録インデックス値
    int      state;          // 状態値
    int      capacity;       // キャリアの容量
    char     *usage;         // キャリアの用途
    char     *carid;         // キャリア ID
    int      map_status;     // マップ読み込み状態
    int      id_status;      // ID読み込み状態
    int      acc_status;     // アクセス状態
    char     *location;      // 搬送位置
    int      slot_count;     // スロット数
    TSLLOT_INFO **slot_list; // スロット情報のリスト
} TCAR_INFO;
```

(2) TSLLOT_INFO Carrier Slot Information

```
typedef struct{
    int      status;         // スロット状態
    int      slotid;        // スロット ID(U1)
    char     *mid;          // マテリアル ID (ロット ID)
} TSLLOT_INFO;
```

(3) TCACT_INFO - Carrier Action Information - S3F17

```
typedef struct{
    TDATAID  dataid;
    char     *caction;       // car action cmd
    int      action_index;   // caction の index
    char     *carspec;       // carrier spec ( carid )
    int      ptn;            // port no.
    int      cp_count;       // parameter count
    TCACT_PARA **cp_list;    // paramete list
}TCACT_INFO;
```

(4) TCACT_PARA - Carrier Action Parameter Information

```
typedef struct{
    char     *cattrid;        // cattrid
    int      attr_index;     // cattrid のインデックス
    void     *cattrdata;     // cattrdata
}TCACT_PARA;
```

(5) TCACT_ERR_INFO – Carrier Action Response Information – S3F18

```
typedef struct{
    int      caack;           // CAACK
    int      err_count;      // 含まれているエラー情報の数
    TERR_INFO **err_list;    // エラー情報リスト
} TCACT_ERR_INFO;
```

(6) TERR_INFO – Object 応答エラー情報

```
typedef struct{
    int      errcode;        // エラーコード (U1)
    char     *errtext;      // エラーテキスト
} TERR_INFO;
```

(3.21 参照)

3.14.2 CAR キャリア情報アクセス関数

3.14.2.1 EngAllocCarInfo() - キャリアの登録

(1) 呼出書式

[C, C++]

```
API int APIX EngAllocCarInfo(
    char *carid,           // キャリア ID 格納領域のポインタ
    int *index            // 得られた情報領域インデックス格納用ポインタ
);
```

[.NET VB]

```
Function EngAllocCarInfo (
    ByVal carid As String,
    ByRef index As Int32) As Int32
```

[.NET C#]

```
int EngAllocCarInfo(
    byte[] carid,
    ref int index );
```

(2) 引数

carid

登録したいキャリア ID が格納されているポインタです。

index

登録された情報領域のインデックス値が格納される領域のポインタです。

(3) 戻り値

戻り値	意味
0	正常に登録できた。
1	指定された CARID は既に登録されていた。
(-1)	登録できなかった。

(4) 説明

キャリアを新規にシステムに登録するための関数です。

登録は、引数 carid で与えられるキャリア ID をシステムに登録します。

正常に登録できた場合は、index で指定される領域に登録された情報領域のインデックスが設定返却されます。もし、carid に指定されたキャリアが既に登録済みであった場合には関数の戻り値 =1 を返却します。index には既に登録されている情報領域のインデックスが設定されます。

得られたインデックスを使って、情報の設定、取得、削除などのアクセスを行うことができます。

3.14.2.2 EngSetCarInfo() - キャリア情報の設定 EngSetCarInfoX() インデクスでのキャリア情報の設定

(1) 呼出書式

[C, C++]

```
API int APIX EngSetCarInfo(
    TCAR_INFO *pinfo           // キャリア情報格納構造体のポインタ
);

API int APIX EngSetCarInfoX(
    int index,                // EngAllLocCarInfo() で得られたキャリア情報のインデクス
    TCAR_INFO *pinfo         // キャリア情報格納構造体のポインタ
);
```

[.NET VB]

```
Function EngSetCarInfo (
    ByRef cinfo As dsh_info.TCAR_INFO) As Int32

Function EngSetCarInfoX (
    ByVal index As Int32,
    ByRef cinfo As dsh_info.TCAR_INFO) As Int32
```

[.NET C#]

```
int EngSetCarInfo(
    ref TCAR_INFO cinfo );

int EngSetCarInfoX(
    int index,
    ref TCAR_INFO cinfo );
```

(2) 引数

pinfo

設定したいキャリア情報が格納されている格納構造体領域のポインタです。

(3) 戻り値

戻り値	意味
0	正常に設定できた。
(-1)	設定できなかった。

(4) 説明

本関数は、pinfo に格納されているキャリア情報の設定・変更に使います。

引数 pinfo 内の carid メンバーに指定されるキャリア ID の情報として設定されます。

pinfo 内には CARID の他、スロット情報などの情報が含まれます。

指定した CARID が既に登録済みである場合には、pinfo 内の情報はすべて書き換えられます。

pinfo 内の CARID が未登録であった場合は、登録手続きをしてから CAR 情報を設定します。

(EngAllLocCarInfo() 関数で行われる登録と同じ登録が行われます。)

3.14.2.3 EngGetCarInfo() - キャリア情報の取得 EngGetCarInfoX() - インデクス指定でのキャリア情報の取得

(1) 呼出書式

[C, C++]

```
API int APIX EngGetCarInfo(
    char *carid, // CARID が格納されている領域のポインタ
    TCAR_INFO *pinfo // キャリア情報を格納する構造体ポインタを格納するポインタ
);
```

```
API int APIX EngGetCarInfoX(
    int index, // キャリア情報のインデクス
    TCAR_INFO *pinfo // キャリア情報を格納するポインタ
);
```

[.NET VB]

```
Function EngGetCarInfo (
    ByVal carid As String,
    ByRef cinfo As dsh_info.TCAR_INFO) As Int32
```

```
Function EngGetCarInfoX (
    ByVal index As Int32,
    ByRef cinfo As dsh_info.TCAR_INFO) As I
```

[.NET C#]

```
int EngGetCarInfo(
    byte[] carid,
    ref TCAR_INFO cinfo );
```

```
int EngGetCarInfoX(
    int index,
    ref TCAR_INFO cinfo );
```

(2) 引数

carid

キャリア ID が格納されている領域のポインタです。

pinfo

取得したキャリア情報を格納する構造体領域のポインタです。

index

キャリア ID 情報のインデクスです。登録時に EngAllLocCarInfo()関数によって与えられます。インデクスは、CARID から EngGetCarIdIndex()関数で取得することができます。

(3) 戻り値

戻り値	意味
0	正常に取得できた。
(-1)	取得できなかった。(CARID が未登録であった。)

(4) 説明

carid または index に指定されているキャリアの情報を pinfo 構造体領域に格納します。

TCAR_INFO 構造体の中に情報を格納するために必要なメモリは、DSHEng4 が準備確保します。即ち、構造体のメンバーの中でポインタになっている情報の実体即ち、carid、スロット情報などのためのメモリは DSHEng4 が準備します。

これらのメモリは、使用后、ユーザが DSHEng4 の API 関数を使って次のように開放してください。

```
TCAR_INFO *pinfo;

if ( EngGetCarInfo(carid, pinfo ) == 0 ){
    pinfo の処理
    処理終了後
    DshFreeTCAR_INFO( pinfo );           // pinfo 内に使用されているメモリの開放
}
```

3.14.2.4 EngDelCarInfo() - キャリアの削除 EngDelCarInfoX() インデクスでのキャリアの削除

(1) 呼出書式

[C, C++]

```
API int APIX EngDelCarInfo(
    char *carid; // CARID が格納されている領域のポインタ
);
```

```
API int APIX EngDelCarInfoX(
    int index, // インデクス(0,1,2,...)
);
```

[.NET VB]

```
Function EngDelCarInfo (
    ByVal carid As String) As Int32
```

```
Function EngDelCarInfoX (
    ByVal index As Int32) As Int32
```

[.NET C#]

```
int EngDelCarInfo(
    byte[] carid );
```

```
int EngDelCarInfoX(
    int index );
```

(2) 引数

carid

キャリア ID が格納されている領域のポインタです。

index

削除したいキャリア情報のインデクスです。

(3) 戻り値

戻り値	意味
0	正常に削除できた。
(-1)	CARID が未登録であった。

(4) 説明

carid または index に指定されているキャリア ID の情報をシステムの登録から削除します。

3.14.2.5 EngSetCarState() キャリア情報状態の設定 EngSetCarStateX() インデクスでのキャリア情報状態の設定

(1) 呼出書式

[C, C++]

```
API int APIX EngSetCarState(
    char    *carid,           // CARID が格納されている領域のポインタ
    int     state            // 設定したい状態値
);
```

```
API int APIX EngSetCarStateX(
    int     index,          // キャリア情報のインデクス
    int     state            // 設定したい状態値
);
```

[.NET VB]

```
Function EngSetCarState (
    ByVal carid As String,
    ByVal state As Int32) As Int32
```

```
Function EngSetCarStateX (
    ByVal index As Int32,
    ByVal state As Int32) As Int32
```

[.NET C#]

```
int EngSetCarState(
    byte[] carid,
    int state);
```

```
int EngSetCarStateX(
    int index,
    int state);
```

(2) 引数

carid

キャリア ID が格納されている領域のポインタです。

state

設定したい CARID 情報の状態値です。

値=(-1)は使用不可を意味する値になります。

index

キャリア情報のインデクスです。登録時に EngAllCarInfo()関数によって与えられます。
インデクスは、CARID から EngGetCarIdIndex()関数で取得することができます。

(3) 戻り値

戻り値	意味
0	正常に設定できた。
(-1)	CARID が未登録であった。

(4) 説明

キャリア情報の状態値を設定します。

状態値の意味合いは値=(-1)以外についてはユーザが定義して使用します。

3.14.2.6 EngGetCarState() キャリア情報状態の取得 EngGetCarStateX() インデクスでのキャリア情報状態の取得

(1) 呼出書式

[C, C++]

```
API int APIX EngGetCarState(
    char    *carid,           // CARID が格納されている領域のポインタ
    int     *state           // 取得した状態値格納用領域ポインタ
);
```

```
API int APIX EngGetCarStateX(
    int     index,           // キャリアのインデクス
    int     *state           // 取得した状態値格納用領域ポインタ
);
```

[.NET VB]

```
Function EngGetCarState (
    ByVal carid As String,
    ByRef state As Int32) As Int32
```

```
Function EngGetCarStateX (
    ByVal index As Int32,
    ByRef state As Int32) As Int32
```

[.NET C#]

```
int EngGetCarState(
    byte[] carid,
    ref int state);
```

```
int EngGetCarStateX(
    int index,
    ref int state);
```

(2) 引数

carid

キャリア ID が格納されている領域のポインタです。

state

取得した CARID 情報の状態値を格納する領域のポインタです。

index

キャリア情報のインデクスです。登録時に EngAllCarInfo()関数によって与えられます。
インデクスは、CARID から EngGetCarIdIndex()関数で取得することができます。

(3) 戻り値

戻り値	意味
0	正常に取得できた。
(-1)	CARID が未登録であった。

(4) 説明

キャリア情報の状態値を取得します。
状態値の意味合いは値=(-1)以外についてはユーザが定義して使用します。

3.14.2.7 EngSetCarIdStatus() キャリア ID 読取り状態の設定 EngSetCarIdStatusX() インデクスでのキャリア ID 読取り状態の設定

(1) 呼出書式

[C, C++]

```
API int APIX EngSetCarIdStatus(
    char    *carid,           // CARID が格納されている領域のポインタ
    int     status           // 設定したい状態値
);
```

```
API int APIX EngSetCarIdStatusX(
    int     index,           // キャリア情報のインデクス
    int     status           // 設定したい状態値
);
```

[.NET VB]

```
Function EngSetCarIdStatus (
    ByVal carid As String,
    ByVal state As Int32) As Int32
```

```
Function EngSetCarIdStatusX (
    ByVal index As Int32,
    ByVal state As Int32) As Int32
```

[.NET C#]

```
int EngSetCarIdStatus(
    byte[] carid,
    int state );
```

```
int EngSetCarIdStatusX(
    int index,
    int state );
```

(2) 引数

carid

キャリア ID が格納されている領域のポインタです。

status

設定したい CARID 読取りの状態値です。

index

キャリア情報のインデクスです。登録時に EngAllCarInfo()関数によって与えられます。
インデクスは、CARID から EngGetCarIdIndex()関数で取得することができます。

(3) 戻り値

戻り値	意味
0	正常に設定できた。
(-1)	CARID が未登録であった。

(4) 説明

キャリア情報の ID 読み取り状態値を設定します。
デフォルトの値は次のとおりです。

キャリアID状態のデフォルト値

状態値記号	値
ST_CarIdNotRead	0
ST_CarIdWaitingForHost	1
ST_CarIdVerificationOK	2
ST_CarIdVerificationFail	3
ST_CarIdNotExist	4

3.14.2.8 EngGetCarIdStatus() キャリア ID 読取り状態の取得 EngGetCarIdStatusX() インデクスでのキャリア ID 読取り状態の取得

(1) 呼出書式

[C, C++]

```
API int APIX EngGetCarIdStatus(
    char    *carid,           // CARID が格納されている領域のポインタ
    int     *status          // 取得した状態値格納用領域ポインタ
);
```

```
API int APIX EngGetCarIdStatusX(
    int     index,           // キャリアのインデクス
    int     *status          // 取得した状態値格納用領域ポインタ
);
```

[.NET VB]

```
Function EngGetCarIdStatus (
    ByVal carid As String,
    ByRef state As Int32) As Int32
```

```
Function EngGetCarIdStatusX (
    ByVal index As Int32,
    ByRef state As Int32) As Int32
```

[.NET C#]

```
int EngGetCarIdStatus(
    byte[] carid,
    ref int state );
```

```
int EngGetCarIdStatusX(
    int index,
    ref int state );
```

(2) 引数

carid

キャリア ID が格納されている領域のポインタです。

status

取得した CARID の読取り状態値を格納する領域のポインタです。

index

キャリア情報のインデクスです。登録時に EngAllCarInfo()関数によって与えられます。
インデクスは、CARID から EngGetCarIdIndex()関数で取得することができます。

(3) 戻り値

戻り値	意味
0	正常に取得できた。
(-1)	CARID が未登録であった。

(4) 説明

キャリア情報の ID 読み取り状態値を取得します。
デフォルトの値は次のとおりです。

キャリアID状態のデフォルト値

状態値記号	値
ST_CarIdNotRead	0
ST_CarIdWaitingForHost	1
ST_CarIdVerificationOK	2
ST_CarIdVerificationFail	3
ST_CarIdNotExist	4

3.14.2.9 EngSetCarMapStatus() スロットマップ状態の設定 EngSetCarMapStatusX() インデクスでのスロットマップ状態の設定

(1) 呼出書式

[C, C++]

```
API int APIX EngSetCarMapStatus(
    char    *carid,           // CARID が格納されている領域のポインタ
    int     status           // 設定したい状態値
);
```

```
API int APIX EngSetCarMapStatusX(
    int     index,           // キャリア情報のインデクス
    int     status           // 設定したい状態値
);
```

[.NET VB]

```
Function EngSetCarMapStatus (
    ByVal carid As String,
    ByVal map_status As Int32) As Int32
```

```
Function EngSetCarMapStatusX (
    ByVal index As Int32,
    ByVal map_status As Int32) As Int32
```

[.NET C#]

```
int EngSetCarMapStatus(
    byte[] carid,
    int map_status );
```

```
int EngSetCarMapStatusX(
    int index,
    int map_status );
```

(2) 引数

carid

キャリア ID が格納されている領域のポインタです。

status

設定したい CARID のスロットマップ状態値です。

index

キャリア情報のインデクスです。登録時に EngAllCarInfo()関数によって与えられます。
インデクスは、CARID から EngGetCarIdIndex()関数で取得することができます。

(3) 戻り値

戻り値	意味
0	正常に設定できた。
(-1)	CARID が未登録であった。

(4) 説明

キャリア情報のスロットマップ状態値を設定します。

スロットマップ状態のデフォルト値

状態値記号	値
ST_SlotMapNotRead	0
ST_SlotMapWaitingForHost	1
ST_SlotMapVerificationOK	2
ST_SlotMapVerificationFail	3
ST_SlotNotExist	4

3.14.2.10 EngGetCarMapStatus() スロットマップ状態の取得
 EngGetCarMapStatusX() インデクスでのスロットマップ状態の取得

(1) 呼出書式

[C, C++]

```
API int APIX EngGetCarMapStatus(
    char    *carid,           // CARID が格納されている領域のポインタ
    int     *status          // 取得した状態値格納用領域ポインタ
);
```

```
API int APIX EngGetCarMapStatusX(
    int     index,           // キャリアのインデクス
    int     *status          // 取得した状態値格納用領域ポインタ
);
```

[.NET VB]

```
Function EngGetCarMapStatus (
    ByVal carid As String,
    ByRef map_status As Int32) As Int32
```

```
Function EngGetCarMapStatusX (
    ByVal index As Int32,
    ByRef map_status As Int32) As Int32
```

[.NET C#]

```
int EngGetCarMapStatus(
    byte[] carid,
    ref int map_status );
```

```
int EngGetCarMapStatusX(
    int index,
    ref int map_status );
```

(2) 引数

carid

キャリア ID が格納されている領域のポインタです。

status

取得した CARID のスロットマップ状態値を格納する領域のポインタです。

index

キャリア情報のインデクスです。登録時に EngAllCarInfo()関数によって与えられます。
 インデクスは、CARID から EngGetCarIdIndex()関数で取得することができます。

(3) 戻り値

戻り値	意味
0	正常に取得できた。
(-1)	CARID が未登録であった。

(4) 説明

キャリア情報のスロットマップ状態値を取得します。

スロットマップ状態のデフォルト値

状態値記号	値
ST_SlotMapNotRead	0
ST_SlotMapWaitingForHost	1
ST_SlotMapVerificationOK	2
ST_SlotMapVerificationFail	3
ST_SlotNotExist	4

3.14.2.11 EngSetCarAccessStatus() アクセス状態の設定 EngSetCarAccessStatusX() インデクスでのアクセス状態の設定

(1) 呼出書式

[C, C++]

```
API int APIX EngSetCarAccessStatus(
    char    *carid,           // CARID が格納されている領域のポインタ
    int     status           // 設定したい状態値
);
```

```
API int APIX EngSetCarAccessStatusX(
    int     index,           // キャリア情報のインデクス
    int     status           // 設定したい状態値
);
```

[.NET VB]

```
Function EngSetCarAccessStatus (
    ByVal carid As String,
    ByVal acc_status As Int32) As Int32
```

```
Function EngSetCarAccessStatusX (
    ByVal index As Int32,
    ByVal acc_status As Int32) As Int32
```

[.NET C#]

```
int EngSetCarAccessStatus(
    byte[] carid,
    int acc_status );
```

```
int EngSetCarAccessStatusX(
    int index,
    int acc_status );
```

(2) 引数

status

設定したい CARID のアクセス状態値です。

index

キャリア情報のインデクスです。登録時に EngAllCarInfo()関数によって与えられます。インデクスは、CARID から EngGetCarIdIndex()関数で取得することができます。

(3) 戻り値

戻り値	意味
0	正常に設定できた。
(-1)	CARID が未登録であった。

(4) 説明

キャリア情報のアクセス状態値を設定します。

アクセス状態のデフォルト値

状態値記号	値
CAR_NOT_ACCESSED	0
CAR_IN_ACCESS	1
CAR_COMPLETE	2
CAR_STOPPED	3

3.14.2.12 EngGetCarAccessStatus() アクセス状態の取得 EngGetCarAccessStatusX() インデクスでのアクセス状態の取得

(1) 呼出書式

[C, C++]

```
API int APIX EngGetCarAccessStatus(
    char    *carid,           // CARID が格納されている領域のポインタ
    int     *status          // 取得した状態値格納用領域ポインタ
);
```

```
API int APIX EngGetCarAccessStatusX(
    int     index,           // キャリアのインデクス
    int     *status          // 取得した状態値格納用領域ポインタ
);
```

[.NET VB]

```
Function EngGetCarAccessStatus (
    ByVal carid As String,
    ByRef acc_status As Int32) As Int32
```

```
Function EngGetCarAccessStatusX (
    ByVal index As Int32,
    ByRef acc_status As Int32) As Int32
```

[.NET C#]

```
int EngGetCarAccessStatus(
    byte[] carid,
    ref int acc_status );
```

```
int EngGetCarAccessStatusX(
    int index,
    ref int acc_status );
```

(2) 引数

carid

キャリア ID が格納されている領域のポインタです。

status

取得した CARID のアクセス状態値を格納する領域のポインタです。

index

キャリア情報のインデクスです。登録時に EngAllCarInfo()関数によって与えられます。
 インデクスは、CARID から EngGetCarIdIndex()関数で取得することができます。

(3) 戻り値

戻り値	意味
0	正常に取得できた。
(-1)	CARID が未登録であった。

(4) 説明

キャリアのアクセス状態値を取得します。

アクセス状態のデフォルト値

状態値記号	値
CAR_NOT_ACCESSED	0
CAR_IN_ACCESS	1
CAR_COMPLETE	2
CAR_STOPPED	3

3.14.2.13 EngSetCarLocation() ロケーションの設定 EngSetCarLocationX() インデクスでのロケーションの設定

(1) 呼出書式

[C, C++]

```
API int  APIX EngSetCarLocation(
    char    *carid,           // CARID が格納されている領域のポインタ
    char    *location        // 設定したいロケーション格納ポインタ
);
API int  APIX EngSetCarLocationX(
    int     index,           // キャリア情報のインデクス
    char    *location        // 設定したいロケーション格納ポインタ
);
```

[.NET VB]

```
Function EngSetCarLocation (
    ByVal carid As String,
    ByVal location As String) As Int32
Function EngSetCarLocationX (
    ByVal index As Int32,
    ByVal location As String) As Int32
```

[.NET C#]

```
int EngSetCarLocation(
    byte[] carid,
    byte[] location );
int EngSetCarLocationX(
    int index,
    byte[] location );
```

(2) 引数

carid

キャリア ID が格納されている領域のポインタです。

location

設定したい CARID のロケーション値 (文字列) です。

index

キャリア情報のインデクスです。登録時に EngAllocCarInfo()関数によって与えられます。インデクスは、CARID から EngGetCarIdIndex()関数で取得することができます。

(3) 戻り値

戻り値	意味
0	正常に設定できた。
(-1)	CARID が未登録であった。

(4) 説明

キャリア情報のロケーション(文字列)を設定します。

3.14.2.14 EngGetCarLocation() ロケーションの取得 EngGetCarLocationX() インデクスでのロケーションの取得

(1) 呼出書式

[C, C++]

```
API int  APIX EngGetCarLocation(
    char    *carid,           // CARID が格納されている領域のポインタ
    char    *location        // 取得したロケーション格納ポインタ
);
API int  APIX EngGetCarLocationX(
    char    index,          // キャリアのインデクス
    char    *location      // 取得したロケーション格納ポインタ
);
```

[.NET VB]

```
Function EngGetCarLocation (
    ByVal carid As String,
    ByVal location As String) As Int32
Function EngGetCarLocationX (
    ByVal index As Int32,
    ByVal location As String) As Int32
```

[.NET C#]

```
int EngGetCarLocation(
    byte[] carid,
    byte[] location );
int EngGetCarLocationX(
    int index,
    byte[] location );
```

(2) 引数

carid

キャリア ID が格納されている領域のポインタです。

location

取得した CARID のロケーション値を格納する領域のポインタです。

index

キャリア情報のインデクスです。登録時に EngAllocCarInfo()関数によって与えられます。インデクスは、CARID から EngGetCarIdIndex()関数で取得することができます。

(3) 戻り値

戻り値	意味
0	正常に取得できた。
(-1)	CARID が未登録であった。

(4) 説明

キャリア情報のロケーション(文字列)を取得します。

3.14.2.15 EngSetCarUsage() Usage の設定 EngSetCarUsageX() インデクスでの Usage の設定

(1) 呼出書式

[C, C++]

```
API int  APIX EngSetCarUsage(
    char    *carid,           // CARID が格納されている領域のポインタ
    char    *usage           // 設定したい Usage 格納ポインタ
);
API int  APIX EngSetCarUsageX(
    int     index,          // キャリア情報のインデクス
    char    *usage         // 設定したい Usage 格納ポインタ
);
```

[.NET VB]

```
Function EngSetCarUsage (
    ByVal carid As String,
    ByVal usage As String) As Int32
Function EngSetCarUsageX (
    ByVal index As Int32,
    ByVal usage As String) As Int32
```

[.NET C#]

```
int EngSetCarUsage(
    byte[] carid,
    byte[] usage );
int EngSetCarUsageX(
    int index,
    byte[] usage );
```

(2) 引数

carid

キャリア ID が格納されている領域のポインタです。

usage

設定したい CARID の usage 値 (文字列) です。

index

キャリア情報のインデクスです。登録時に EngAllocCarInfo()関数によって与えられます。インデクスは、CARID から EngGetCarIdIndex()関数で取得することができます。

(3) 戻り値

戻り値	意味
0	正常に設定できた。
(-1)	CARID が未登録であった。

(4) 説明

キャリア情報の Usage(文字列)を設定します。

3.14.2.16 EngGetCarUsage() Usage の取得 EngGetCarUsageX() インデクスでの Usage の取得

(1) 呼出書式

[C, C++]

```
API int  APIX EngGetCarUsage(
    char    *carid,           // CARID が格納されている領域のポインタ
    char    *Usage           // 取得した Usage 格納ポインタ
);
API int  APIX EngGetCarUsageX(
    char    index,          // キャリアのインデクス
    char    *Usage         // 取得した Usage 格納ポインタ
);
```

[.NET VB]

```
Function EngGetCarUsage (
    ByVal carid As String,
    ByVal usage As String) As Int32
Function EngGetCarUsageX (
    ByVal index As Int32,
    ByVal usage As String) As Int32
```

[.NET C#]

```
int EngGetCarUsage(
    byte[] carid,
    byte[] usage );
int EngGetCarUsageX(
    int index,
    byte[] usage );
```

(2) 引数

carid

キャリア ID が格納されている領域のポインタです。

Usage

取得した CARID の Usage 値を格納する領域のポインタです。

index

キャリア情報のインデクスです。登録時に EngAllocCarInfo()関数によって与えられます。インデクスは、CARID から EngGetCarIdIndex()関数で取得することができます。

(3) 戻り値

戻り値	意味
0	正常に取得できた。
(-1)	CARID が未登録であった。

(4) 説明

キャリア情報の Usage(文字列)を取得します。

3.14.2.17 EngSetCarLotid() Lotid の設定

EngSetCarLotidX() インデクスでの Lotid の設定

(1) 呼出書式

[C, C++]

```
API int  APIX EngSetCarLotid(
    char    *carid,           // CARID が格納されている領域のポインタ
    int     order,          // スロット順位(0,1,2...)
    char    *lotid           // 設定したい lotid 格納ポインタ
);
```

```
API int  APIX EngSetCarLotidX(
    int     index,          // キャリア情報のインデクス
    int     order,          // スロット順位(0,1,2...)
    char    *lotid         // 設定したい lotid 格納ポインタ
);
```

[.NET VB]

```
Function EngSetCarLotid (
    ByVal carid As String,
    ByVal order As Int32,
    ByVal lotid As String) As Int32
```

```
Function EngSetCarLotidX (
    ByVal index As Int32,
    ByVal order As Int32,
    ByVal lotid As String) As Int32
```

[.NET C#]

```
int EngSetCarLotid(
    byte[] carid,
    int order,
    byte[] lotid );
```

```
int EngSetCarLotidX(
    int index,
    int order,
    byte[] lotid );
```

(2) 引数

carid

キャリア ID が格納されている領域のポインタです。

order

アクセス対象キャリアスロットの順位です。(0,1,2...)

lotid

設定したいスロットの lotid 値 (文字列) です。

index

キャリア情報のインデクスです。登録時に EngAllCarInfo()関数によって与えられます。

インデクスは、CARID から EngGetCarIdIndex()関数で取得することができます。

(3) 戻り値

戻り値	意味
0	正常に設定できた。
(-1)	CARID が未登録であった。

(4) 説明

指定キャリア ID の order 番目のスロットの lotid(文字列)を設定します。

3.14.2.18 EngGetCarLotid() Lotidの取得 EngGetCarLotidX() インデクスでのLotidの取得

(1) 呼出書式

[C, C++]

```
API int  APIX EngGetCarLotid(
    char    *carid,           // CARID が格納されている領域のポインタ
    int     order,           // スロット順位(0,1,2...)
    char    *lotid           // 取得した lotid 格納ポインタ
);
```

```
API int  APIX EngGetCarLotidX(
    int     index,           // キャリアのインデクス
    int     order,           // スロット順位(0,1,2...)
    char    *lotid           // 取得した lotid 格納ポインタ
);
```

[.NET VB]

```
Function EngGetCarLotid (
    ByVal carid As String,
    ByVal order As Int32,
    ByVal lotid As String) As Int32
```

```
Function EngGetCarLotidX (
    ByVal index As Int32,
    ByVal order As Int32,
    ByVal lotid As String) As Int32
```

[.NET C#]

```
int EngGetCarLotid(
    byte[] carid,
    int order,
    byte[] lotid );
```

```
int EngGetCarLotidX(
    int index,
    int order,
    byte[] lotid );
```

(2) 引数

carid

キャリア ID が格納されている領域のポインタです。

order

アクセス対象キャリアスロットの順位です。(0,1,2...)

lotid

取得したスロットの lotid 値を格納する領域のポインタです。

index

キャリア情報のインデクスです。登録時に EngAllCarInfo()関数によって与えられます。

インデクスは、CARID から EngGetCarIdIndex()関数で取得することができます。

(3) 戻り値

戻り値	意味
0	正常に取得できた。
(-1)	CARID が未登録であった。

(4) 説明

指定されたキャリア Id の order 番目の lotid(文字列)を取得します。

3.14.2.19 EngSetCarSubstid() Substid の設定 EngSetCarSubstidX() インデクスでの Substid の設定

(1) 呼出書式

[C, C++]

```
API int  APIX EngSetCarSubstid(
    char    *carid,           // CARID が格納されている領域のポインタ
    int     order,           // スロット順位(0,1,2...)
    char    *substid         // 設定したいsubstid 格納ポインタ
);
```

```
API int  APIX EngSetCarSubstidX(
    int     index,           // キャリア情報のインデクス
    int     order,           // スロット順位(0,1,2...)
    char    *substid         // 設定したいsubstid 格納ポインタ
);
```

[.NET VB]

```
Function EngSetCarSubstid (
    ByVal carid As String,
    ByVal order As Int32,
    ByVal substid As String) As Int32
```

```
Function EngSetCarSubstidX (
    ByVal index As Int32,
    ByVal order As Int32,
    ByVal substid As String) As Int32
```

[.NET C#]

```
int EngSetCarSubstid(
    byte[] carid,
    int order,
    byte[] substid );
```

```
int EngSetCarSubstidX(
    int index,
    int order,
    byte[] substid );
```

(2) 引数

carid

キャリア ID が格納されている領域のポインタです。

order

アクセス対象キャリアスロットの順位です。(0,1,2...)

substid

設定したいスロットの substid 値 (文字列) です。

index

キャリア情報のインデクスです。登録時に EngAllCarInfo()関数によって与えられます。

インデクスは、CARID から EngGetCarIdIndex()関数で取得することができます。

(3) 戻り値

戻り値	意味
0	正常に設定できた。
(-1)	CARID が未登録であった。

(4) 説明

指定キャリア ID の order 番目のスロットの subst id(文字列)を設定します。

3.14.2.20 EngGetCarSubstid() Substid の取得 EngGetCarSubstidX() インデクスでの Substid の取得

(1) 呼出書式

[C, C++]

```
API int  APIX EngGetCarSubstid(
    char    *carid,           // CARID が格納されている領域のポインタ
    int     order,           // スロット順位(0,1,2...)
    char    *substid         // 取得した substid 格納ポインタ
);
```

```
API int  APIX EngGetCarSubstidX(
    char    index,           // キャリアのインデクス
    int     order,           // スロット順位(0,1,2...)
    char    *substid         // 取得した substid 格納ポインタ
);
```

[.NET VB]

```
Function EngGetCarSubstid (
    ByVal carid As String,
    ByVal order As Int32,
    ByVal substid As String) As Int32
```

```
Function EngGetCarSubstidX (
    ByVal index As Int32,
    ByVal order As Int32,
    ByVal substid As String) As Int32
```

[.NET C#]

```
int EngGetCarSubstid(
    byte[] carid,
    int order,
    byte[] substid );
```

```
int EngGetCarSubstidX(
    int index,
    int order,
    byte[] substid );
```

(2) 引数

carid

キャリア ID が格納されている領域のポインタです。

order

アクセス対象キャリアスロットの順位です。(0,1,2...)

substid

取得したスロットの substid 値を格納する領域のポインタです。

index

キャリア情報のインデクスです。登録時に EngAllCarInfo()関数によって与えられます。

インデクスは、CARID から EngGetCarIdIndex()関数で取得することができます。

(3) 戻り値

戻り値	意味
0	正常に取得できた。
(-1)	CARID が未登録であった。

(4) 説明

指定されたキャリア Id の order 番目のスロットの基板 ID substid(文字列)を取得します。

3.14.2.21 EngSetCarSlotmap() Slotmap の設定 EngSetCarSlotmapX() インデクスでの Slotmap の設定

(1) 呼出書式

[C, C++]

```
API int  APIX EngSetCarSlotmap(
    char    *carid,           // CARID が格納されている領域のポインタ
    int     order,           // スロット順位(0,1,2...)
    int     slotmap          // 設定したいslotmap 値
);
```

```
API int  APIX EngSetCarSlotmapX(
    int     index,           // キャリア情報のインデクス
    int     order,           // スロット順位(0,1,2...)
    int     slotmap          // 設定したいslotmap 値
);
```

[.NET VB]

```
Function EngSetCarSlotmap (
    ByVal carid As String,
    ByVal order As Int32,
    ByVal slotmap As Int32) As Int32
```

```
Function EngSetCarSlotmapX (
    ByVal index As Int32,
    ByVal order As Int32,
    ByVal slotmap As Int32) As Int32
```

[.NET C#]

```
int EngSetCarSlotmap(
    byte[] carid,
    int order,
    int slotmap );
```

```
int EngSetCarSlotmapX(
    int index,
    int order,
    int slotmap );
```

(2) 引数

carid

キャリア ID が格納されている領域のポインタです。

order

アクセス対象キャリアスロットの順位です。(0,1,2...)

slotmap

設定したいスロットの slotmap 値です。

index

キャリア情報のインデクスです。登録時に EngAllCarInfo()関数によって与えられます。

インデクスは、CARID から EngGetCarIdIndex()関数で取得することができます。

(3) 戻り値

戻り値	意味
0	正常に設定できた。
(-1)	CARID が未登録であった。

(4) 説明

指定キャリア ID の order 番目のスロットの slotmap を設定します。

3.14.2.22 EngGetCarSlotmap() Slotmap の取得 EngGetCarSlotmapX() インデクスでの Slotmap の取得

(1) 呼出書式

[C, C++]

```
API int  APIX EngGetCarSlotmap(
    char    *carid,           // CARID が格納されている領域のポインタ
    int     order,           // スロット順位(0,1,2...)
    int     *slotmap         // 取得した slotmap 値格納ポインタ
);
```

```
API int  APIX EngGetCarSlotmapX(
    char    index,           // キャリアのインデクス
    int     order,           // スロット順位(0,1,2...)
    int     *slotmap         // 取得した slotmap 値格納ポインタ
);
```

[.NET VB]

```
Function EngGetCarSlotmap (
    ByVal carid As String,
    ByVal order As Int32,
    ByRef slotmap As Int32) As Int32
```

```
Function EngGetCarSlotmapX (
    ByVal index As Int32,
    ByVal order As Int32,
    ByRef slotmap As Int32) As Int32
```

[.NET C#]

```
int EngGetCarSlotmap(
    byte[] carid,
    int order,
    ref int slotmap );
```

```
int EngGetCarSlotmapX(
    int index,
    int order,
    ref int slotmap );
```

(2) 引数

carid

キャリア ID が格納されている領域のポインタです。

order

アクセス対象キャリアスロットの順位です。(0,1,2...)

slotmap

取得したスロットの slotmap 値を格納する領域のポインタです。

index

キャリア情報のインデクスです。登録時に EngAllCarInfo()関数によって与えられます。

インデクスは、CARID から EngGetCarIdIndex()関数で取得することができます。

(3) 戻り値

戻り値	意味
0	正常に取得できた。
(-1)	CARID が未登録であった。

(4) 説明

指定されたキャリア ID の order 番目のスロットの slotmap 値を取得します。

3.14.2.23 EngSetCarSlotCount() スロット数の設定 EngSetCarSlotCountX() インデクスでのスロット数の設定

(1) 呼出書式

[C, C++]

```
API int APIX EngSetCarSlotCount(
    char    *carid,           // CARID が格納されている領域のポインタ
    int     count            // 設定したいスロット数
);
API int APIX EngSetCarSlotCountX(
    int     index,          // キャリア情報のインデクス
    int     count            // 設定したいスロット数
);
```

[.NET VB]

```
Function EngSetCarSlotCount (
    ByVal carid As String,
    ByVal count As Int32) As Int32
Function EngSetCarSlotCountX (
    ByVal index As Int32,
    ByVal count As Int32) As Int32
```

[.NET C#]

```
int EngSetCarSlotCount(
    byte[] carid,
    int count);
int EngSetCarSlotCountX(
    int index,
    int count);
```

(2) 引数

carid

キャリア ID が格納されている領域のポインタです。

count

設定したい CARID のスロット数です。

index

キャリア情報のインデクスです。登録時に EngAllocCarInfo()関数によって与えられます。インデクスは、CARID から EngGetCarIdIndex()関数で取得することができます。

(3) 戻り値

戻り値	意味
0	正常に設定できた。
(-1)	CARID が未登録であった。

(4) 説明

キャリア情報のスロット数を設定します。

3.14.2.24 EngGetCarSlotCount() スロット数の取得 EngGetCarSlotCountX() インデクスでのスロット数の取得

(1) 呼出書式

[C, C++]

```
API int APIX EngGetCarSlotCount(
    char    *carid,           // CARID が格納されている領域のポインタ
    int     *count           // 取得したスロット数格納用領域ポインタ
);
API int APIX EngGetCarSlotCountX(
    int     index,          // キャリアのインデクス
    int     *count         // 取得したスロット数格納用領域ポインタ
);
```

[.NET VB]

```
Function EngGetCarSlotCount (
    ByVal carid As String,
    ByRef count As Int32) As Int32
Function EngGetCarSlotCountX (
    ByVal index As Int32,
    ByRef count As Int32) As Int32
```

[.NET C#]

```
int EngGetCarSlotCount(
    byte[] carid,
    ref int count );
int EngGetCarSlotCountX(
    int index,
    ref int count );
```

(2) 引数

carid

キャリア ID が格納されている領域のポインタです。

count

取得した CARID のスロット数を格納する領域のポインタです。

index

キャリア情報のインデクスです。登録時に EngAllocCarInfo()関数によって与えられます。インデクスは、CARID から EngGetCarIdIndex()関数で取得することができます。

(3) 戻り値

戻り値	意味
0	正常に取得できた。
(-1)	CARID が未登録であった。

(4) 説明

キャリア情報のスロット数を取得します。

3.14.2.25 EngGetCarList() 全登録キャリア ID 取得関数

(1) 呼出書式

[C, C++]

```
API int APIX EngGetCarList(
    TTEXT_DLIST **list           // 取得リスト格納ポインタの格納ポインタ
);
```

[.NET VB]

```
Function EngGetCarList (
    ByRef list As IntPtr) As Int32
```

[.NET C#]

```
int EngGetCarList(
    IntPtr list );
```

(2) 引数

list

取得できた CARID が格納されている TTEXT_DLIST 構造体のポインタを格納する領域のポインタです。

(3) 戻り値

戻り値	意味
0	正常に取得できた。
(-1)	取得できなかった。

(4) 説明

システムに登録されている全 CARID(キャリア ID) リストを TTEXT_DLIST 構造体に取り出すための関数です。

info->name_list には NULL が設定されます。(本情報には定義名がないため)

取得した情報の処理が終了した後、DshFreeTText_DLIST()関数で list 内部の情報格納用に使用されているメモリを開放してください。

TTEXT_DLIST 構造体は次のとおりです。

```
typedef struct{
    int         count;           // 取得できた ID 数
    char       **id_list;       // 取得できた ID 格納用配列
    char       **name_list;     // 取得できた名前格納ポインタ配列
}TTEXT_DLIST;
```


3.14.2.26 EngGetCarId() インデクスから CARID (キャリア ID) の取得

(1) 呼出書式

[C, C++]

```
API int APIX EgnGetCarId(
    int index,           // キャリア情報のインデクス
    char *carid         // 取得した CARID を格納する領域のポインタ
);
```

[.NET VB]

```
Function EngGetCarId (
    ByVal index As Int32,
    ByVal carid As String) As Int32
```

[.NET C#]

```
int EngGetCarId(
    int index,
    byte[] carid );
```

(2) 引数

index

キャリア情報のインデクスです。登録時に EngAllCarInfo()関数によって与えられます。インデクスは、CARID から EngGetCarIdIndex()関数で取得することができます。

carid

CARID を格納するための領域のポインタです。

(3) 戻り値

戻り値	意味
0	正常に取得できた。
(-1)	取得できなかった。(未登録)

(4) 説明

キャリア情報のインデクスから CARID (キャリア ID) を取得し、carid に格納します。正常に取得できた場合は関数戻り値として 0 が返却されます。

3.14.2.27 EngGetCarIdIndex() CARID (キャリア ID) からインデクスの取得

(1) 呼出書式

[C, C++]

```
API int APIX EgnGetCarIdIndex(
    char *carid,           // CARID が格納されている領域のポインタ
    int *index            // 取得したインデクスを格納するための領域のポインタ
);
```

[.NET VB]

```
Function EngGetCarIdIndex (
    ByVal carid As String,
    ByRef index As Int32) As Int32
```

[.NET C#]

```
int EngGetCarIdIndex(
    byte[] carid,
    ref int index );
```

(2) 引数

carid

インデクスを取得したい対象の CARID が格納されている領域のポインタです。

index

取得したインデクスの値を格納するための領域のポインタです。

(3) 戻り値

戻り値	意味
0	正常に取得できた。
(-1)	取得できなかった。(未登録)

(4) 説明

carid に指定される CARID (キャリア ID) からキャリア情報インデクスを取得するための関数です。取得されたインデクスは index で指定された領域に格納されます。正常に取得できた場合は関数戻り値として 0 が返却されます。

3.14.3 CAR キャリア関連ライブラリ関数

3.14.3.1 DshFreeTCAR_INFO() - キャリア情報構造体メモリの開放

(1) 呼出書式

[C, C++]

```
API void APIX DshFreeTCAR_INFO(  
    TCAR_INFO *pinfo          // メモリを開放したいキャリア情報が格納されている構造体のポインタ  
);
```

[.NET VB]

```
Sub DshFreeTCAR_INFO (  
    ByRef info As dsh_info.TCAR_INFO)
```

[.NET C#]

```
void DshFreeTCAR_INFO(  
    ref TCAR_INFO info );
```

(2) 引数

pinfo

メモリを解放したいキャリア情報構造体のポインタです。

(3) 戻り値

なし。

(4) 説明

TCAR_INFO 構造体内で情報格納用に使用されているメモリを全て解放します。
開放した後、TCAR_INFO の内容を全て 0 で初期設定します。
pinfo が NULL ならば、何も処理しません。

3.14.3.2 DshCopyTCAR_INFO() - キャリア情報構造体メモリのコピー

(1) 呼出書式

[C, C++]

```
API int APIX DshCopyTCAR_INFO(
    TCAR_INFO *dinfo,           // 北°-先のポインタ
    TCAR_INFO *sinfo           // 北°-元のポインタ
);
```

[.NET VB]

```
Function DshCopyTCAR_INFO (
    ByRef info As dsh_info.TCAR_INFO,
    ByRef sinfo As dsh_info.TCAR_INFO) As Int32
```

[.NET C#]

```
int DshCopyTCAR_INFO(
    ref TCAR_INFO info,
    ref TCAR_INFO sinfo );
```

(2) 引数

dinfo

キャリア情報のコピー先構造体メモリのポインタです。

sinfo

コピー元のキャリア情報が格納されている構造体メモリのポインタです。

(3) 戻り値

戻り値	意味
0	正常に北°-できた。
(-1)	sinfo または dinfo の値が NULL だったので北°-できなかった。

(4) 説明

sinfo が指す TCAR_INFO 構造体内に格納されているキャリア情報を dinfo が指定する TCAR_INFO 構造体にコピーします。

dinfo 内のメンバーで新しいメモリが必要なものは本関数が取得します。

dinfo 内メンバーで確保されたメモリは、使用后、DshFreeTCAR_INFO() 関数を使って開放してください。

3.14.3.3 DshInitCarInfo() TCAR_INFOの初期設定

(1) 呼出書式

[C, C++]

```
API int APIX DshInitCarInfo(
    TCAR_INFO *info,           // TCAR_INFO 構造体のポインタ
    char *carid,              // キャリア ID
    char *usage,              // Usage 名
    int map_status,           // map_status
    int id_status,            // id_status
    int acc_status,           // acc_status
    char *location,           // location
    int slot_count            // slot count
);
```

[.NET VB]

```
Sub DshInitCarInfo (
    ByRef info As dsh_info.TCAR_INFO,
    ByVal carid As String,
    ByVal usage As String,
    ByVal map_status As Int32,
    ByVal id_status As Int32,
    ByVal acc_status As Int32,
    ByVal location As String,
    ByVal slot_count As Int32)
End Sub
```

[.NET C#]

```
void DshInitCarInfo(
    ref TCAR_INFO info,
    byte[] carid,
    byte[] usage,
    int map_status,
    int id_status,
    int acc_status,
    byte[] location,
    int slot_count );
```

(2) 引数

info

TCAR_INFO 構造体のポインタです。このメンバーを初期設定します。

carid

設定するキャリア ID (文字列) です。

usage

材料の種類です。

map_status

キャリアの MAP STATUS です。

id_status

キャリアの ID STATUS です。

acc_status

キャリアの ACCESSING STATUS です。

location

現在の location ID です。

slot_count

Slot 数です。

(3) 戻り値

なし。

(4) 説明

本関数は APP が OFFLINE でキャリア情報を生成する際に使用することができます。

最初に info 内をクリアします。そして、引数で指定された情報を info 内に設定します。

メモリが必要なメンバーについてはメモリを確保し情報をコピーします。

3.14.3.4 DshInitCarSlotInfo() キャリアスロット情報の初期設定

(1) 呼出書式

[C, C++]

```
API void APIX DshInitCarSlotInfo(
    T SLOT_INFO *info,           // スロット情報構造体のポインタ
    int slotid,                 // スロット ID
    char *mid,                  // mid(material id)
    char *substid,              // substrate id
    char *substloc              // 現在の基板位置
);
```

[.NET VB]

```
Sub DshInitCarSlotInfo (
    ByRef sinfo As dsh_info.T SLOT_INFO,
    ByVal slotid As Int32,
    ByVal mid As String,
    ByVal substid As String,
    ByVal substloc As String)
```

[.NET C#]

```
void DshInitCarSlotInfo(
    ref T SLOT_INFO sinfo,
    int slotid,
    byte[] mid,
    byte[] substid,
    byte[] substloc );
```

(2) 引数

info
初期設定するキャリアスロット情報の構造体ポインタです。

slotid
このスロットに与えられたスロット ID です。

mid
このスロットに含まれる基板の material (ロット) ID です。

substid
このスロットに含まれる基板の ID です。

substloc
このスロットに含まれる基板の location 位置です。

(3) 戻り値

なし。

(4) 説明

最初に T SLOT_INFO 構造体のメモリを確保し、info に設定します。そして引数に与えられた情報を info 内の各メンバーに設定します。

info 自体のメモリを確保するとともに、メモリが必要なメンバーについてはメモリを確保し情報をコピーします。

ここで確保されたメモリはDshFreeTCAR_INFO()関数でTCAR_INFO構造体内メモリが開放されるときに同時に開放されます。

3.14.3.5 DshPutCarSlotInfo() キャリア情報にスロット情報を設定

(1) 呼出書式

[C, C++]

```
API int APIX DshPutCarSlotInfo(
    TCAR_INFO    *info,           // キャリア情報構造体のポインタ
    TSLOT_INFO   *sinfo          // スロット情報構造体のポインタ
);
```

[.NET VB]

```
Function DshPutCarSlotInfo (
    ByRef info As dsh_info.TCAR_INFO,
    ByRef sinfo As IntPtr) As Int32
```

[.NET C#]

```
int DshPutCarSlotInfo(
    ref TCAR_INFO info,
    ref TSLOT_INFO sinfo );
```

(2) 引数

info

スロット情報を設定するキャリア情報構造体のポインタです。

sinfo

設定したいスロット情報が格納されている構造体のポインタです。

(3) 戻り値

戻り値	意味
0	正常に設定できた。
(-1)	予約されているか外数を超えていたので設定できなかった。

(4) 説明

先に DshInitCarInfo() で初期設定されたキャリア情報構造体の slot_list 上に 1 個のスロット情報を加えます。

slot_count で予約された数の slot_list 上には本関数が実行される順にスロット情報を追加していきます。

slot_count 以内の追加であれば設定後 0 を返却します。

もし、slot_count 数を超える場合は、設定しないで (-1) を返却します。

3.14.3.6 DshPutCarSlotInfoCopy() キャリア情報にスロット情報をコピーして設定

(1) 呼出書式

[C, C++]

```
API int APIX DshPutCarSlotInfoCopy(
    TCAR_INFO    *info,           // キャリア情報構造体のポインタ
    TSLOT_INFO   *sinfo          // スロット情報構造体のポインタ
);
```

[.NET VB]

```
Function DshPutCarSlotInfoCopy (
    ByRef info As dsh_info.TCAR_INFO,
    ByRef sinfo As dsh_info.TSLOT_INFO) As Int32
```

[.NET C#]

```
int DshPutCarSlotInfoCopy(
    ref TCAR_INFO info,
    ref TSLOT_INFO sinfo);
```

(2) 引数

info

スロット情報を設定するキャリア情報構造体のポインタです。

sinfo

設定したいスロット情報が格納されている構造体のポインタです。

(3) 戻り値

戻り値	意味
0	正常に設定できた。
(-1)	予約されているか外数を超えていたので設定できなかった。

(4) 説明

先に DshInitCarInfo() で初期設定されたキャリア情報構造体の slot_list 上に 1 個のスロット情報を加えます。

本関数は、sinfo の情報を別メモリにコピーした上で追加します。

(3.14.4.13 の DshPutCarSlotInfo() の場合は sinfo の情報をそのまま追加します。

slot_count で予約された数の slot_list 上には本関数が実行される順にスロット情報を追加していきます。slot_count 以内の追加であれば設定後 0 を返却します。

もし、slot_count 数を超える場合は、設定しないで (-1) を返却します。

3.14.3.7 DshInitCarContent() キャリアコンテンツマップ情報の初期設定

(1) 呼出書式

[C, C++]

```
API void APIX DshInitCarContent(
    TCACT_CONTENT *info,           // キャリア情報構造体のポインタ
    int count                       // コンテントマップ (lotid, substid)数
);
```

[.NET VB]

```
Sub DshInitCarContent (
    ByRef info As dsh_info.TCACT_CONTENT,
    ByVal count As Int32)
```

[.NET C#]

```
void DshInitCarContent(
    ref TCACT_CONTENT info,
    int count);
```

(2) 引数

info

キャリアコンテンツマップ情報を格納する構造体のポインタです。

count

格納するコンテンツマップ数です。(後で設定する lotid, 構造体情報の数)

(3) 戻り値

なし。

(4) 説明

info で指定される TCACT_CONTENT 構造体を初期設定します。

info 内に格納するロット ID と基板 ID の数を count で指定します。本関数は、info の lotid, substid メンバーに count 数分のポインタ配列領域を設けます。

TCACT_INFO 構造体内に対するロット、基板情報の設定は DshPutCarContent()関数を使用してください。本 ContentMap 情報は S3F17 メッセージの"ContentMap"属性に関連する情報です。

3.14.3.8 DshPutCarContent() キャリアコンテンツマップ情報の設定

(1) 呼出書式

[C, C++]

```
API int APIX DshPutCarContent(
    TCACT_CONTENT *info,           // キャリア情報構造体のポインタ
    int order,                     // 設定情報の順番(0,1,2,...)
    char *lotid,                  // ロット ID(文字列)の格納ポインタ
    char *substid                 // 基板 ID(文字列)の格納ポインタ
);
```

[.NET VB]

```
Function DshPutCarContent (
    ByRef info As dsh_info.TCACT_CONTENT,
    ByVal order As Int32,
    ByVal mid As String,
    ByVal substid As String) As Int32
```

[.NET C#]

```
int DshPutCarContent(
    ref TCACT_CONTENT info,
    int order,
    byte[] lotid,
    byte[] substid);
```

(2) 引数

info

キャリアコンテンツマップ情報を格納する構造体のポインタです。

order

情報を設定する配列の順位(0,1,2,...)です。

lotid

設定したいロット ID (文字列) です。

substid

設定したい基板 ID (文字列) です。

(3) 戻り値

戻り値	意味
0	正常に設定できた。
(-1)	予約されているカウント数を超えていたので設定できなかった。

(4) 説明

info で指定される TCACT_CONTENT 構造体内の lotid[], substid[] 配列の order 番目の配列にロット ID, 基板 ID を設定します。

lotid, substid 何れも NULL の場合は設定しません。

指定配列順位 order が info 内の count 以上の値の場合は設定しないで(-1)を返却します。