

リファレンスマニュアル

A-L i n k   D L L

# 目 次

## 第 1 章 関数一覧

## 第 2 章 関数仕様

2-1 デジタル入出力ユニット関数 .....	2-1
2-2 アナログ入出力ユニット関数 .....	2-15
2-3 位置決めユニット関数 .....	2-28
2-4 エンコーダ・カウンタユニット関数 .....	2-36
2-5 シリアルユニット関数 .....	2-42
2-6 ちび丸君シリーズ アナログ入出力 ユニット関数 .....	2-52
2-7 ALD シリーズ アナログ入出力ユニット関数 .....	2-58
2-8 制御補助関数 .....	2-69

## 第 1 章 関数一覧

### 1) デジタル入出力ユニット

ユニットタイプ 別記号	関 数	機 能
DIO	AL_DIO_Open( )	ユニットをオープンします
	AL_DIO_Close( )	ユニットをクローズします
	AL_DIO_InData( )	ユニットの 16 ビットデータを取得します
	AL_DIO_InDataBit( )	ユニットの指定したビットからデータを取得します
	AL_DIO_OutData( )	ユニットに 16 ビットデータを出力します
	AL_DIO_OutDataBit( )	ユニットの指定したビットにデータを出力します
	AL_DIO_InOData( )	ユニットに出力したデータを取得します
	AL_DIO_InODataBit( )	ユニットの指定したビットの出力データを取得します
	AL_DIO_SetRenewData( )	DI データの変化検知エリアの 16 ビット設定をします(*1)
	AL_DIO_SetRenewBit( )	DI データの変化検知エリアのビット設定をします(*1)
	AL_DIO_GetRenewData( )	DI データの変化検知エリアの 16 ビット設定を取得します(*1)
	AL_DIO_GetRenewBit( )	DI データの変化検知エリアのビット設定を取得します(*1)
	AL_DIO_GetRenewStat( )	DI データの変化検知エリアの変化を取得します(*1)
	AL_DIO_GetCondition( )	ユニットの通信状態を取得します

### 2) アナログ入出力ユニット

ユニットタイプ 別記号	関 数	機 能
ADA	AL_ADA_Open( )	ユニットをオープンします
	AL_ADA_Close( )	ユニットをクローズします
	AL_ADA_InValue( )	ユニットの指定チャンネルから 12 ビットデータを取得します
	AL_ADA_InValueFull( )	ユニットの指定チャンネルから 16 ビットデータを取得します
	AL_ADA_OutValue( )	ユニットの指定チャンネルに 12 ビットデータを出力します
	AL_ADA_OutValueFull( )	ユニットの指定チャンネルに 16 ビットデータを出力します
	AL_ADA_InOValue( )	ユニットの指定チャンネルに出力した 12 ビットデータを取得します
	AL_ADA_InOValueFull( )	ユニットの指定チャンネルに出力した 16 ビットデータを取得します
	AL_ADA_SetFilter( )	ユニットのフィルタ設定を行います
	AL_ADA_GetFilter( )	ユニットのフィルタ設定を取得します
	AL_ADA_GetInRange( )	ユニットの入力レンジ設定を取得します
	AL_ADA_GetOutRange( )	ユニットの出力レンジ設定を取得します
	AL_ADA_GetCondition( )	ユニットの通信状態を取得します

### 3) 位置決めユニット

ユニットタイプ 別記号	関 数	機 能
AXSA	AL_AXSA_Open( )	ユニットをオープンします
	AL_AXSA_Close( )	ユニットをクローズします
	AL_AXSA_PutCmd( )	ユニットへの各種コマンドをセットします
	AL_AXSA_Answer( )	ユニットの応答(ステータス)を取得します
	AL_AXSA_AnsWith( )	ユニットの応答(ステータス)と位置データを取得します
	AL_AXSA_GetStatus( )	常時アップロード時にステータスと位置データを取得します
	AL_AXSA_GetCondition( )	ユニットの通信状態を取得します

## 4) エンコーダ・カウンタユニット

ユニットタイプ別記号	関 数	機 能
ENC	AL_ENC_Open( )	ユニットをオープンします
	AL_ENC_Close( )	ユニットをクローズします
	AL_ENC_PutCmd( )	ユニットにコマンドを送信します
	AL_ENC_GetCondition( )	ユニットの通信状態を取得します

## 5) シリアルユニット

ユニットタイプ別記号	関 数	機 能
SIO	AL_SIO_Open( )	ユニットをオープンします
	AL_SIO_Close( )	ユニットをクローズします
	AL_SIO_Config( )	シリアル通信の通信設定を行います
	AL_SIO_PutData	データを送信します
	AL_SIO_GetData( )	データを受信します
	AL_SIO_ClearError( )	シリアル通信時のエラーをクリアします
	AL_SIO_CheckBuffer( )	受信バッファに取得されているデータ数を取得します
	AL_SIO_ClearBuffer( )	受信バッファをクリアします
	AL_SIO_GetCondition( )	スレーブの通信状態を取得します

## 6) ちび丸君シリーズ アナログ入出力ユニット

ユニットタイプ別記号	関 数	機 能
ADAC	AL_ADAC_Open( )	ユニットをオープンします
	AL_ADAC_Close( )	ユニットをクローズします
	AL_ADAC_InValue( )	ユニットの指定チャンネルから 12 ビットデータを取得します
	AL_ADAC_OutValue( )	ユニットの指定チャンネルに 12 ビットデータを出力します
	AL_ADAC_SetFilter( )	ユニットのフィルタ設定を行います
	AL_ADAC_GetCondition( )	ユニットの通信状態を取得します

## 7) ALD シリーズ アナログ入出力ユニット

ユニットタイプ別記号	関 数	機 能
ADAD	AL_ADAD_Open( )	ユニットをオープンします
	AL_ADAD_Close( )	ユニットをクローズします
	AL_ADAD_InValue( )	ユニットの指定チャンネルから 13 ビットデータを取得します
	AL_ADAD_InValueFull( )	ユニットの指定チャンネルから 16 ビットデータを取得します
	AL_ADAD_OutValue( )	ユニットの指定チャンネルに 13 ビットデータを出力します
	AL_ADAD_OutValueFull( )	ユニットの指定チャンネルに 16 ビットデータを出力します
	AL_ADAD_InOValue( )	ユニットの指定チャンネルに出力した 13 ビットデータを取得します
	AL_ADAD_InOValueFull( )	ユニットの指定チャンネルに出力した 16 ビットデータを取得します
	AL_ADAD_SetFilter( )	ユニットのフィルタ設定を行います
	AL_ADAD_GetFilter( )	ユニットのフィルタ設定を取得します
	AL_ADAD_GetCondition( )	ユニットの通信状態を取得します

## 8) 制御補助

関 数	機 能
AL_BeginPeriod( )	Windows 内部タイマの分解能を変更します
AL_EndPeriod( )	変更した Windows 内部タイマを標準のものに戻します
AL_CheckBoard( )	PCI ホート、PC カード のオープンの状態を調べます
AL_CheckStatusLED( )	PCI ホート、PC カード の LED の状態を調べます
AL_ClearStatusLED( )	PCI ホート、PC カード の LED を消灯させます
AL_GetErrorCount( )	PCI ホート、PC カード のエラーを取得します(*1)

( \* 1 ) A-Link Ver2 通信 IC 搭載ボード使用時のみサポート (PCILZ10-0~PCILZ13-0、CRDLZ02-0)

## 第2章 関数仕様

### 2-1 デジタル入出力ユニット関数

#### AL\_DIO\_Open 関数

---

機能	デジタル入出力ユニットをオープンします		
書式	int AL_DIO_Open(int Lid)		
引数	Lid	:	ロジカル ID
戻り値	AL_ER_OK	:	正常
	AL_ER_OPENDEVICE	:	ボードが使用できない状態にあります
	AL_ER_ALREADYOPEN	:	すでにオープンしています
	AL_ER_INICONFIG	:	ini ファイル記述に誤りがあります
	AL_ER_CREATESEMAPH	:	DLL 内部セマフォ生成に失敗しました
説明	<p>ロジカル ID で指定したデジタル入出力ユニットをオープンします。 ロジカル ID は ALink.ini に記述したものを指定しなければなりません。 この関数コール後、指定デジタル入出力ユニットにアクセス可能となります。 デジタル入出力ユニットを使用の際には、必ずコールする必要があります。 ボード設定に誤りがある場合、全ボードをアクセス不可とします。 この場合、関数は [AL_ER_OPENDEVICE] を返します。</p>		

---

AL\_DIO\_Close 関数

---

**機能** デジタル入出力ユニットをクローズします

**書式** int AL\_DIO\_Close(int Lid)

**引数** Lid : ロジカル ID

**戻り値**

AL_ER_OK	: 正常
AL_ER_NOTOPEN	: オープンされていません
AL_ER_INICONFIG	: ini ファイル記述に誤りがあります

**説明** ロジカル ID で指定したデジタル入出力ユニットをクローズします。  
ロジカル ID は ALink.ini に記述したものを指定しなければなりません。  
この関数コール後は、指定デジタル入出力ユニットにアクセス不可になります。  
アプリケーションの終了時に必ずコールする必要があります。

---

AL\_DIO\_InData 関数

---

**機能**

デジタル入出力ユニットの入力から 16 ビットデータを取得します

**書式**

```
int AL_DIO_InData(int Lid, unsigned short *InDat)
```

**引数**

Lid : ロジカル ID  
InDat : 16 ビットデータを格納するポインタ

**戻り値**

AL\_ER\_OK : 正常  
AL\_ER\_NOTOPEN : オープンされていません  
AL\_ER\_INICONFIG : ini ファイル記述に誤りがあります

**説明**

ロジカル ID で指定したデジタル入出力ユニットから 16 ビットデータを取得します。  
ロジカル ID は ALink.ini に記述したものを指定しなければなりません。



---

AL\_DIO\_InDataBit 関数

---

**機能**

デジタル入出力ユニットの入力から指定したビットのデータを取得します

**書式**

```
int AL_DIO_InDataBit(int Lid, int BitNo, int *InBit)
```

**引数**

Lid : ロジカル ID  
BitNo : ビット番号 (0-15)  
InBit : ビットデータを格納するポインタ  
ビットデータ (ALDIO\_ON : 1、ALDIO\_OFF : 0)

**戻り値**

AL\_ER\_OK : 正常  
AL\_ER\_NOTOPEN : オープンされていません  
AL\_ER\_INICONFIG : ini ファイル記述に誤りがあります  
AL\_ER\_INVALIDPARAM : 無効な引数です

**説明**

ロジカル ID で指定したデジタル入出力ユニットからビット番号で指定したビットデータを取得します。  
ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

---

AL\_DIO\_OutData 関数

---

**機能**

デジタル入出力ユニットの出力に 16 ビットデータを出力します

**書式**

```
int AL_DIO_OutData(int Lid, unsigned short OutDat)
```

**引数**

Lid : ロジカル ID  
OutDat : 16 ビット出力データ

**戻り値**

AL_ER_OK	: 正常
AL_ER_NOTOPEN	: オープンされていません
AL_ER_INICONFIG	: ini ファイル記述に誤りがあります

**説明**

ロジカル ID で指定したデジタル入出力ユニットに 16 ビットデータを出力します。  
ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

---

AL\_DIO\_OutDataBit 関数

---

**機能**

デジタル入出力ユニットの出力に指定したビットにデータを出力します

**書式**

```
int AL_DIO_OutDataBit(int Lid, int BitNo, int OutBit)
```

**引数**

Lid : ロジカル ID  
BitNo : ビット番号 (0-15)  
OutBit : ビットデータ (ALDIO\_ON : 1、ALDIO\_OFF : 0)

**戻り値**

AL\_ER\_OK : 正常  
AL\_ER\_NOTOPEN : オープンされていません  
AL\_ER\_INICONFIG : ini ファイル記述に誤りがあります  
AL\_ER\_INVALIDPARAM : 無効な引数です

**説明**

ロジカル ID で指定したデジタル入出力ユニットのビット番号で指定したビットにビットデータを出力します。  
ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

---

AL\_DIO\_In0Data 関数

---

**機能**

デジタル入出力ユニットの出力から 16 ビットデータを取得します

**書式**

```
int AL_DIO_In0Data(int Lid, unsigned short *InDat)
```

**引数**

Lid : ロジカル ID  
InDat : 16 ビットデータを格納するポインタ

**戻り値**

AL\_ER\_OK : 正常  
AL\_ER\_NOTOPEN : オープンされていません  
AL\_ER\_INICONFIG : ini ファイル記述に誤りがあります

**説明**

ロジカル ID で指定したデジタル入出力ユニットの出力 16 ビットデータを取得します。  
ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

---

AL\_DIO\_In0DataBit 関数

---

**機能**

デジタル入出力ユニットの出力から指定したビットのデータを取得します

**書式**

```
int AL_DIO_In0DataBit(int Lid, int BitNo, int *InBit)
```

**引数**

Lid : ロジカル ID  
BitNo : ビット番号 (0-15)  
InBit : ビットデータを格納するポインタ  
ビットデータ (ALDIO\_ON : 1、ALDIO\_OFF : 0)

**戻り値**

AL\_ER\_OK : 正常  
AL\_ER\_NOTOPEN : オープンされていません  
AL\_ER\_INICONFIG : ini ファイル記述に誤りがあります  
AL\_ER\_INVALIDPARAM : 無効な引数です

**説明**

ロジカル ID で指定したデジタル入出力ユニットの出力から指定したビットのデータを取得します。  
ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

---

AL\_DIO\_SetRenewData 関数

---

**機能**

DI データの変化検知エリアを 16 ビット設定します。(\*1)

**書式**

```
int AL_DIO_SetRenewData(int Lid, unsigned short OutDat)
```

**引数**

Lid : ロジカル ID  
OutDat : 16 ビット検知エリアデータ

**戻り値**

AL_ER_OK	: 正常
AL_ER_NOTOPEN	: オープンされていません
AL_ER_INICONFIG	: ini ファイル記述に誤りがあります
AL_ER_NOT36	: A-Link Ver2 通信 IC 搭載ボードが使用されていません

**説明**

ロジカル ID で指定した DI の変化を検知するエリアを設定します。  
このエリア内のデータが変化または既に設定されている他の検知エリアが変化した場合  
AL\_DIO\_GetRenewStat( )により取得できます。  
ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

( \* 1 ) A-Link Ver2 通信 IC 搭載ボード使用時のみサポート (PCILZ10-0~PCILZ13-0、CRDLZ02-0)

## AL\_DIO\_SetRenewBit 関数

**機能**

DI データの変化検知エリアをビット設定します。(\*1)

**書式**

```
int AL_DIO_SetRenewBit(int Lid, int BitNo, int OutBit)
```

**引数**

Lid : ロジカル ID  
 BitNo : ビット番号 (0-15)  
 OutBit : ビットデータ (検知エリア)  
           ビットデータ (ALDIO\_ON : 1、ALDIO\_OFF : 0)

**戻り値**

AL\_ER\_OK : 正常  
 AL\_ER\_NOTOPEN : オープンされていません  
 AL\_ER\_INICONFIG : ini ファイル記述に誤りがあります  
 AL\_ER\_INVALIDPARAM : 無効な引数です  
 AL\_ER\_NOT36 : A-Link Ver2 通信 IC 搭載ボードが使用されていません

**説明**

ロジカル ID で指定した DI の変化を検知するビットエリアを設定します。  
 このエリア内のデータが変化または既に設定されている他の検知エリアが変化した場合、  
 AL\_DIO\_GetRenewStat( )により取得できます。  
 ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

( \* 1 ) A-Link Ver2 通信 IC 搭載ボード使用時のみサポート (PCILZ10-0~PCILZ13-0、CRDLZ02-0)

---

AL\_DIO\_GetRenewData 関数

---

**機能**

DI データの変化検知エリアの 16 ビット設定を取得します。(\*1)

**書式**

```
int AL_DIO_GetRenewData(int Lid, unsigned short *InDat)
```

**引数**

Lid : ロジカル ID  
InDat : 16 ビット検知エリアデータを格納するポインタ

**戻り値**

AL_ER_OK	: 正常
AL_ER_NOTOPEN	: オープンされていません
AL_ER_INICONFIG	: ini ファイル記述に誤りがあります
AL_ER_NOT36	: A-Link Ver2 通信 IC 搭載ボードが使用されていません

**説明**

ロジカル ID で指定した DI の設定された変化検知エリアを取得します。  
このエリア内のデータが変化または既に設定されている他の検知エリアが変化した場合、  
AL\_DIO\_GetRenewStat( )により取得できます。  
ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

( \* 1 ) A-Link Ver2 通信 IC 搭載ボード使用時のみサポート (PCILZ10-0~PCILZ13-0、CRDLZ02-0)



## AL\_DIO\_GetRenewData 関数

**機能**

DI データの変化検知エリアのビット設定を取得します。(\*1)

**書式**

```
int AL_DIO_GetRenewBit(int Lid, int BitNo, int *InBit)
```

**引数**

Lid : ロジカル ID  
BitNo : ビット番号 (0-15)  
InBit : ビットデータ (検知エリア) を格納するポインタ  
ビットデータ (ALDIO\_ON : 1、ALDIO\_OFF : 0)

**戻り値**

AL\_ER\_OK : 正常  
AL\_ER\_NOTOPEN : オープンされていません  
AL\_ER\_INICONFIG : ini ファイル記述に誤りがあります  
AL\_ER\_INVALIDPARAM : 無効な引数です  
AL\_ER\_NOT36 : A-Link Ver2 通信 IC 搭載ボードが使用されていません

**説明**

ロジカル ID で指定した DI の設定した変化検知ビットエリアを取得します。  
このエリア内のデータが変化または既に設定されている他の検知エリアが変化した場合、  
AL\_DIO\_GetRenewStat( )により取得できます。  
ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

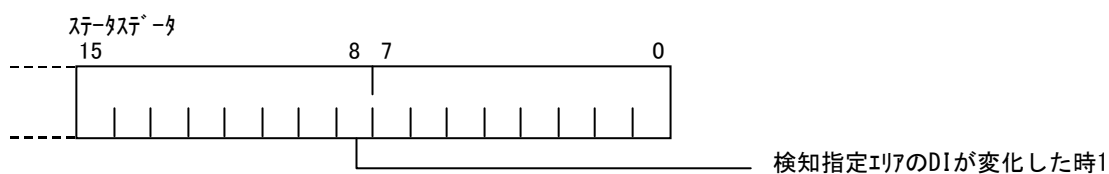
( \* 1 ) A-Link Ver2 通信 IC 搭載ボード使用時のみサポート (PCILZ10-0~PCILZ13-0、CRDLZ02-0)

## AL\_DIO\_GetRenewStat 関数

**機能** DI データの変化検知エリアの変化を取得します。(\*1)

**書式** int AL\_DIO\_GetRenewStat(WORD \*Stat)

**引数** Stat : 検知エリアの変化状態データを格納するポインタ  
ステータステータ



**戻り値**

AL_ER_OK	: 正常
AL_ER_NOTOPEN	: オープンされていません
AL_ER_INICONFIG	: ini ファイル記述に誤りがあります
AL_ER_NOT36	: A-Link Ver2 通信 IC 搭載ボードが使用されていません

**説明** 検知指定エリアの DI が変化したかどうかの状態を取得します。  
ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

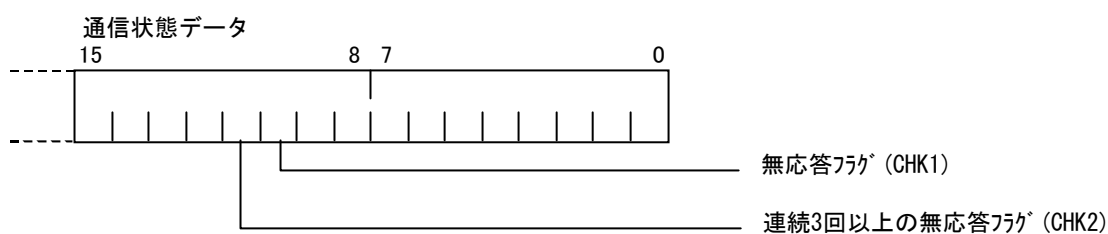
(\* 1) A-Link Ver2 通信 IC 搭載ボード使用時のみサポート (PCILZ10-0~PCILZ13-0、CRDLZ02-0)

## AL\_DIO\_GetCondition 関数

**機能** デジタル入出力ユニットの A-Link 通信状態を取得します

**書式** int AL\_DIO\_GetCondition(int Lid, int \*Cond)

**引数** Lid : ロジカル ID  
Cond : 通信状態データを格納するポインタ



**戻り値** AL\_ER\_OK : 正常  
AL\_ER\_NOTOPEN : オープンされていません  
AL\_ER\_INICONFIG : ini ファイル記述に誤りがあります

**説明** ロジカル ID で指定したデジタル入出力ユニットの通信状態を取得します。  
ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

無応答フラグ (CHK1) とは、1 回でも無応答があれば立ちます。  
連続 3 回以上の無応答フラグ (CHK2) とは、無応答状態 (CHK1) が  
3 回以上連続で発生した場合に立ちます。  
CHK1、CHK2 とも復帰すると消えます。

## 2-2 アナログ入出力ユニット関数

### AL\_ADA\_Open 関数

---

**機能**            アナログ入出力ユニットをオープンします

**書式**            `int AL_ADA_Open(int Lid)`

**引数**            `Lid`            : ロジカル ID

**戻り値**

<code>AL_ER_OK</code>	: 正常
<code>AL_ER_OPENDEVICE</code>	: ボードが使用できない状態にあります
<code>AL_ER_ALREADYOPEN</code>	: すでにオープンしています
<code>AL_ER_INICONFIG</code>	: ini ファイル記述に誤りがあります
<code>AL_ER_CREATESEMAPH</code>	: DLL 内部セマフォ生成に失敗しました

**説明**

ロジカル ID で指定したアナログ入出力ユニットをオープンします。  
ロジカル ID は ALink.ini に記述したものを指定しなければなりません。  
この関数コール後、指定したアナログ入出力ユニットにアクセス可能となります。  
アナログ入出力ユニットを使用の際は、必ずコールする必要があります。  
ボード設定に誤りがある場合、全ボードをアクセス不可とします。  
この場合、関数は `[AL_ER_OPENDEVICE]` を返します。

---

AL\_ADA\_Close 関数

---

**機能**            アナログ入出力ユニットをクローズします

**書式**            `int AL_ADA_Close(int Lid)`

**引数**            `Lid`            : ロジカル ID

**戻り値**            `AL_ER_OK`                                : 正常  
                    `AL_ER_NOTOPEN`                                : オープンされていません  
                    `AL_ER_INICONFIG`                                : ini ファイル記述に誤りがあります

**説明**            ロジカル ID で指定したアナログ入出力ユニットをクローズします。  
                    ロジカル ID は ALink.ini に記述したものを指定しなければなりません。  
                    この関数コール後は、指定したアナログ入出力ユニットにアクセス不可になります。  
                    アプリケーションの終了時に必ずコールする必要があります。

---

AL\_ADA\_InValue 関数

---

**機能**

アナログ入出力ユニットから AD 値を取得します

**書式**

```
int AL_ADA_InValue(int Lid, int Ch, int *InVal)
```

**引数**

Lid : ロジカル ID  
Ch : チャンネル番号 (1- )  
InVal : AD 値を格納するポインタ

**戻り値**

AL\_ER\_OK : 正常  
AL\_ER\_NOTOPEN : オープンされていません  
AL\_ER\_INICONFIG : ini ファイル記述に誤りがあります  
AL\_ER\_INVALIDPARAM : 無効な引数です

**説明**

ロジカル ID で指定したアナログ入出力ユニットの指定したチャンネルから AD データを取得します。  
InVal に格納される値は A/D 変換後の 12 ビットデータとなります。  
ロジカル ID は ALink.ini に記述したものを指定しなければなりません

---

AL\_ADA\_OutValue 関数

---

**機能**            アナログ入出力ユニットに DA 値を出力します

**書式**            `int AL_ADA_OutValue(int Lid, int Ch, int OutVal)`

**引数**

Lid	: ロジカル ID
Ch	: チャンネル番号 (1- )
OutVal	: 出力する DA 値

**戻り値**

AL_ER_OK	: 正常
AL_ER_NOTOPEN	: オープンされていません
AL_ER_INICONFIG	: ini ファイル記述に誤りがあります
AL_ER_INVALIDPARAM	: 無効な引数です

**説明**

ロジカル ID で指定したアナログ入出力ユニットの指定したチャンネルに DA 値を出力します。  
OutVal の値は D/A 変換前の 12 ビットデータとなります。  
ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

---

AL\_ADA\_InValueFull 関数

---

**機能**

アナログ入出力ユニットから 16 ビットデータを全て取得します

**書式**

```
int AL_ADA_InValueFull(int Lid, int Ch, unsigned short *InDat)
```

**引数**

Lid : ロジカル ID  
Ch : チャンネル番号 (1- )  
InDat : 16 ビットデータを格納するポインタ

**戻り値**

AL\_ER\_OK : 正常  
AL\_ER\_NOTOPEN : オープンされていません  
AL\_ER\_INICONFIG : ini ファイル記述に誤りがあります  
AL\_ER\_INVALIDPARAM : 無効な引数です

**説明**

ロジカル ID で指定したアナログ入出力ユニットの指定した AD チャンネルから 16 ビットデータを取得します。  
ロジカル ID は ALink.ini に記述したものを指定しなければなりません。



---

AL\_ADA\_OutValueFull 関数

---

**機能**

アナログ入出力ユニットへ 16 ビットデータを出力します

**書式**

```
int AL_ADA_OutValueFull(int Lid, int Ch, unsigned short OutDat)
```

**引数**

Lid : ロジカル ID  
Ch : チャンネル番号 (1- )  
OutDat : 16 ビット出力データ

**戻り値**

AL_ER_OK	: 正常
AL_ER_NOTOPEN	: オープンされていません
AL_ER_INICONFIG	: ini ファイル記述に誤りがあります
AL_ER_INVALIDPARAM	: 無効な引数です

**説明**

ロジカル ID で指定したアナログ入出力ユニットの指定した DA チャンネルに 16 ビットデータを出力します。  
ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

---

AL\_ADA\_InOValue 関数

---

**機能** アナログ入出力ユニットの出力中の DA 値を取得します

**書式** `int AL_ADA_InOValue(int Lid, int Ch, int *InVal)`

**引数**

Lid	: ロジカル ID
Ch	: チャンネル番号 (1- )
InVal	: DA 値を格納するポインタ

**戻り値**

AL_ER_OK	: 正常
AL_ER_NOTOPEN	: オープンされていません
AL_ER_INICONFIG	: ini ファイル記述に誤りがあります
AL_ER_INVALIDPARAM	: 無効な引数です

**説明** ロジカル ID で指定したアナログ入出力ユニットの指定した DA チャンネルから出力中の DA 値を取得します。  
ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

---

AL\_ADA\_In0ValueFull 関数

---

**機能**

アナログ入出力ユニットの出力中の DA 値を 16 ビット全て取得します

**書式**

```
int AL_ADA_In0ValueFull(int Lid, int Ch, unsigned short *InDat)
```

**引数**

Lid : ロジカル ID  
Ch : チャンネル番号 (1- )  
InDat : 16 ビットデータを格納するポインタ

**戻り値**

AL\_ER\_OK : 正常  
AL\_ER\_NOTOPEN : オープンされていません  
AL\_ER\_INICONFIG : ini ファイル記述に誤りがあります  
AL\_ER\_INVALIDPARAM : 無効な引数です

**説明**

ロジカル ID で指定したアナログ入出力ユニットの指定した DA チャンネルから出力中の 16 ビットデータを取得します。  
ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

## AL\_ADA\_SetFilter 関数

**機能** アナログ入出力ユニットの AD 入力フィルタ設定を行います

**書式** `int AL_ADA_SetFilter(int Lid, int Ch, int Filter, int Cut, int Sample)`

**引数**

Lid : ロジカル ID  
 Ch : チャンネル番号 (1- )  
 Filter : フィルタタイプ  
 Cut : 最大/最小値除外  
 Sample : サンプル回数 (0-15)

## フィルタタイプ

Filter	内容
ALADA_FLTR_KIND_NOT	フィルタ なし
ALADA_FLTR_KIND_SIMP	フィルタ 単純平均
ALADA_FLTR_KIND_MOVE	フィルタ 移動平均

## 最大/最小除外

Cut	内容
ALADA_FLTR_MXMN_NOT	最大/最小値除外しない
ALADA_FLTR_MXMN_CUT	最大/最小値除外する

## サンプル回数 (0-15)

2 の (Sample + 1) 乗で設定される

**戻り値**

AL\_ER\_OK : 正常  
 AL\_ER\_NOTOPEN : オープンされていません  
 AL\_ER\_INICONFIG : ini ファイル記述に誤りがあります  
 AL\_ER\_INVALIDPARAM : 無効な引数です  
 AL\_ER\_COMMANDTIMEOUT : コマンドタイムアウト (応答が返ってきません)

**説明**

ロジカル ID で指定したアナログ入出力ユニットのフィルタ設定を行います。  
 ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

## AL\_ADA\_GetFilter 関数

**機能** アナログ入出力ユニットの AD 入力フィルタ設定を取得します

**書式** `int AL_ADA_GetFilter(int Lid, int Ch, int *Filter, int *Cut, int *Sample)`

**引数**

Lid : ロジカル ID  
 Ch : チャンネル番号 (1- )  
 Filter : フィルタタイプを格納するポインタ  
 Cut : 最大/最小値除外を格納するポインタ  
 Sample : サンプル回数を格納するポインタ

## フィルタタイプ

Filter	内容
ALADA_FLTR_KIND_NOT	フィルタ なし
ALADA_FLTR_KIND_SIMP	フィルタ 単純平均
ALADA_FLTR_KIND_MOVE	フィルタ 移動平均

## 最大/最小除外

Cut	内容
ALADA_FLTR_MXMN_NOT	最大/最小値除外しない
ALADA_FLTR_MXMN_CUT	最大/最小値除外する

## サンプル回数 (0-15)

q2 の (Smple + 1) 乗で設定される

**戻り値**

AL\_ER\_OK : 正常  
 AL\_ER\_NOTOPEN : オープンされていません  
 AL\_ER\_INICONFIG : ini ファイル記述に誤りがあります  
 AL\_ER\_INVALIDPARAM : 無効な引数です  
 AL\_ER\_COMMANDTIMEOUT : コマンドタイムアウト (応答が返ってきません)

**説明**

ロジカル ID で指定したアナログ入出力ユニットのフィルタ設定を行います。  
 ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

## AL\_ADA\_GetInRange 関数

**機能** アナログ入出力ユニットの AD 入力レンジ設定を取得します

**書式** `int AL_ADA_GetInRange(int Lid, int Ch, int *Range)`

**引数**

Lid : ロジカル ID  
 Ch : チャンネル番号 (1- )  
 Range : レンジ設定を格納するポインタ

## レンジ設定

Range	内容
ALADA_RANGE_0005	0 ~ 5 [V]
ALADA_RANGE_0105	1 ~ 5 [V]
ALADA_RANGE_0010	0 ~ 10 [V]
ALADA_RANGE_0505	-5 ~ 5 [V]
ALADA_RANGE_1010	-10 ~ 10 [V]
ALADA_RANGE_0020	0 ~ 20 [mA]
ALADA_RANGE_0420	4 ~ 20 [mA]

**戻り値**

ERROR\_OK : 正常  
 ERROR\_NOTOPEN : オープンされていません  
 ERROR\_INICONFIG : ini ファイル記述に誤りがあります  
 ERROR\_INVALIDPARAM : 無効な引数です  
 AL\_ER\_COMMANDTIMEOUT : コマンドタイムアウト (応答が返ってきません)

**説明**

ロジカル ID で指定したアナログ入出力ユニットの AD 入力レンジ設定を取得することができます。  
 ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

## AL\_ADA\_GetOutRange 関数

**機能** アナログ入出力ユニットの DA 出力レンジ設定を取得します

**書式** `int AL_ADA_GetOutRange(int Lid, int Ch, int *Range)`

**引数**

Lid : ロジカル ID  
 Ch : チャンネル番号 (1- )  
 Range : レンジ設定を格納するポインタ

## レンジ設定

Range	内容
ALADA_RANGE_0005	0 ~ 5 [V]
ALADA_RANGE_0105	1 ~ 5 [V]
ALADA_RANGE_0010	0 ~ 10 [V]
ALADA_RANGE_0505	-5 ~ 5 [V]
ALADA_RANGE_1010	-10 ~ 10 [V]
ALADA_RANGE_0020	0 ~ 20 [mA]
ALADA_RANGE_0420	4 ~ 20 [mA]

**戻り値**

ERROR\_OK : 正常  
 ERROR\_NOTOPEN : オープンされていません  
 ERROR\_INICONFIG : ini ファイル記述に誤りがあります  
 ERROR\_INVALIDPARAM : 無効な引数です  
 AL\_ER\_COMMANDTIMEOUT : コマンドタイムアウト (応答が返ってきません)

**説明**

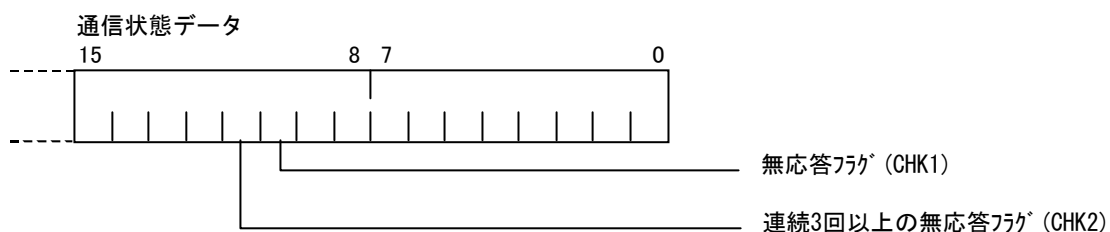
ロジカル ID で指定したアナログ入出力ユニットの DA 出力レンジ設定を取得することができます。  
 ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

## AL\_ADA\_GetCondition 関数

**機能** アナログ入出力ユニットの A-Link 通信状態を取得します

**書式** `int AL_ADA_GetCondition(int Lid, int *Cond)`

**引数** Lid : ロジカル ID  
Cond : 取得する通信状態データを格納するポインタ



**戻り値** AL\_ER\_OK : 正常  
AL\_ER\_NOTOPEN : オープンされていません  
AL\_ER\_INICONFIG : ini ファイル記述に誤りがあります

**説明** ロジカル ID で指定したアナログ入出力ユニットの通信状態を取得します。  
ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

無応答フラグ (CHK1) とは、1 回でも無応答があれば立ちます。  
連続 3 回以上の無応答フラグ (CHK2) とは、無応答状態 (CHK1) が  
3 回以上連続で発生した場合に立ちます。  
CHK1、CHK2 とも復帰すると消えます。



## 2-3 位置決めユニット関数

### AL\_AXSA\_Open 関数

---

**機能** 位置決めユニットをオープンします

**書式** int AL\_AXSA\_Open(int Lid)

**引数** Lid : 位置決めロジカル ID

**戻り値**

AL_ER_OK	: 正常
AL_ER_OPENDEVICE	: ボードが使用できない状態にあります
AL_ER_ALREADYOPEN	: すでにオープンしています
AL_ER_INICONFIG	: ini ファイル記述に誤りがあります
AL_ER_CREATESEMAPH	: DLL 内部セマフォ生成に失敗しました
AL_ER_CREATEMAP	: DLL 内部マップファイル作成に失敗しました

**説明**

ロジカル ID で指定した位置決めユニットをオープンします。  
ロジカル ID は ALink.ini に記述したものを指定しなければなりません。  
この関数コール後、位置決めユニットにアクセス可能となります。  
位置決めユニットを使用の際には、必ずコールする必要があります。  
ボード設定に誤りがある場合、全ボードをアクセス不可とします。  
この場合、関数は [AL\_ER\_OPENDEVICE] を返します。

---

AL\_AXSA\_Close 関数

---

**機能** 位置決めユニットをクローズします

**書式** int AL\_AXSA\_Close(int Lid)

**引数** Lid : ロジカル ID

**戻り値**

AL_ER_OK	: 正常
AL_ER_NOTOPEN	: オープンされていません
AL_ER_INICONFIG	: ini ファイル記述に誤りがあります

**説明** ロジカル ID で指定した位置決めユニットをクローズします。  
ロジカル ID は ALink.ini に記述したものを指定しなければなりません。  
この関数コール後は、位置決めユニットにアクセス不可になります。  
アプリケーションの終了時に必ずコールする必要があります。

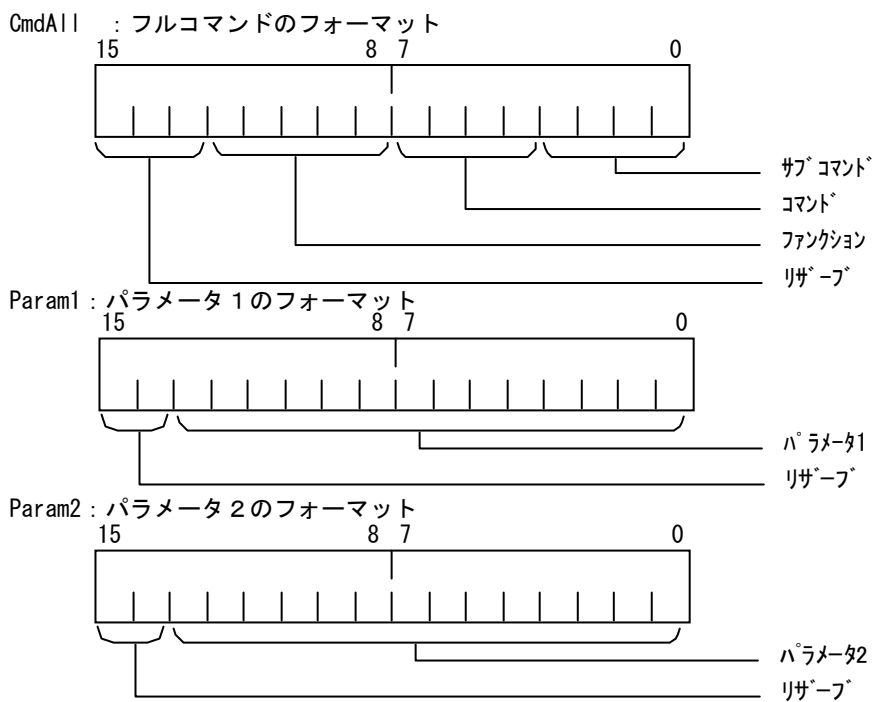
## AL\_AXSA\_PutCmd 関数

**機能** 位置決めユニットへの各種コマンドを送信します

**書式** int AL\_AXSA\_PutCmd(int Lid, int JNo, WORD CmdAll, WORD Param1, WORD Param2)

**引数**

Lid : ロジカル ID  
 JNo : 軸番号 (1- )  
 CmdAll : フルコマンド  
 Param1 : パラメータ 1  
 Param2 : パラメータ 2



\* コマンド機能詳細は別紙「ユーザーズマニュアル ALU シリーズ位置決めユニット 第11章コマンドの詳細」を参照して下さい。

**戻り値**

AL\_ER\_OK : 正常  
 AL\_ER\_NOTOPEN : オープンされていません  
 AL\_ER\_INICONFIG : ini ファイル記述に誤りがあります  
 AL\_ER\_INVALIDPARAM : 無効な引数

**説明**

ロジカル ID で指定した位置決めユニットの各種動作設定を行います。  
 軸を動作させる前に必ず動作設定を行って下さい。  
 ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

## AL\_AXSA\_Answer 関数

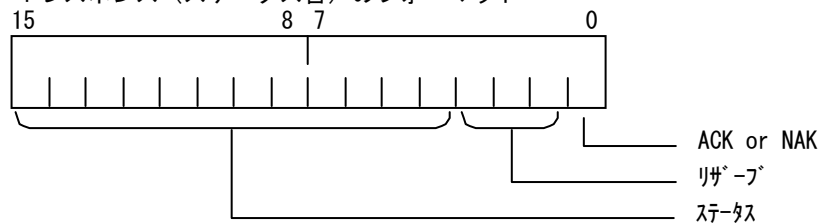
**機能** 位置決めユニットのレスポンス（ステータス含）を取得します

**書式** int AL\_AXSA\_Answer(int Lid, int JNo, WORD \*Status, WORD \*ComErr)

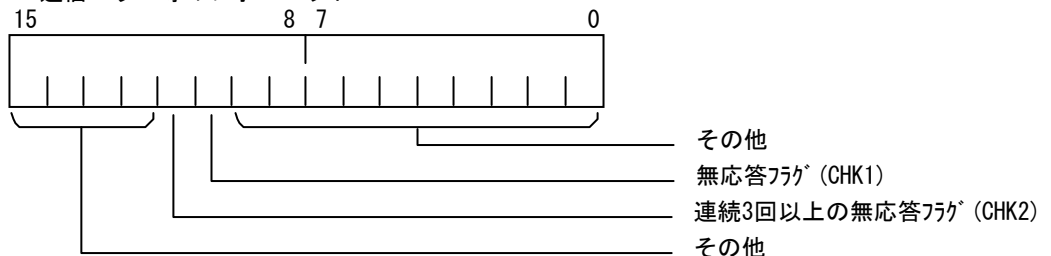
**引数**

Lid : ロジカル ID  
 JNo : 軸番号 (1- )  
 Status : レスポンス（ステータス含）を格納するポインタ  
 ComErr : 通信エラー等を格納するポインタ

Status : レスポンス（ステータス含）のフォーマット



ComErr : 通信エラー等のフォーマット



\* ステータス詳細は別紙「ユーザーズマニュアル [位置決めユニット ソフト編]」を参照して下さい。

**戻り値**

AL\_ER\_OK : 正常  
 AL\_ER\_NOTOPEN : オープンされていません  
 AL\_ER\_INICONFIG : ini ファイル記述に誤りがあります  
 AL\_ER\_INVALIDPARAM : 無効な引数です  
 AL\_ER\_NOTREPLY : 応答が返ってきていません

**説明** ロジカル ID で指定した位置決めユニットからのレスポンスとステータス・通信エラーを取得します。  
 ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

無応答フラグ (CHK1) とは、1 回でも無応答があれば立ちます。

連続 3 回以上の無応答フラグ (CHK2) とは、無応答状態 (CHK1) が 3 回以上連続で発生した場合に立ちます。

CHK1、CHK2 とともに復帰すると消えます。

## AL\_AXSA\_Answer 関数

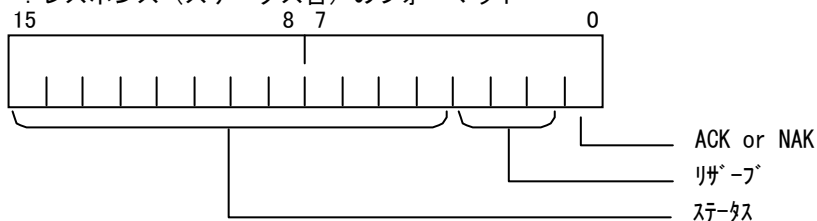
**機能** 位置決めユニットのレスポンス（ステータス含）と位置データを取得します

**書式** int AL\_AXSA\_Answer(int Lid, int JNo, WORD \*Status, long \*JData, WORD \*ComErr)

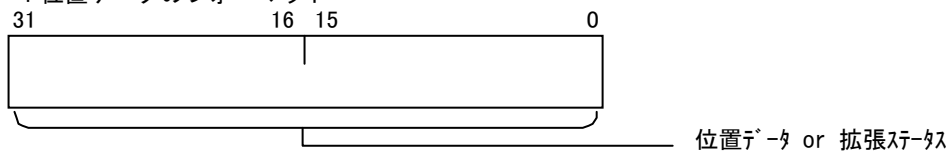
**引数**

Lid : ロジカル ID  
 JNo : 軸番号 (1- )  
 Status : レスポンス（ステータス含）を格納するポインタ  
 JData : 位置データを格納するポインタ  
 ComErr : 通信エラー等を格納するポインタ

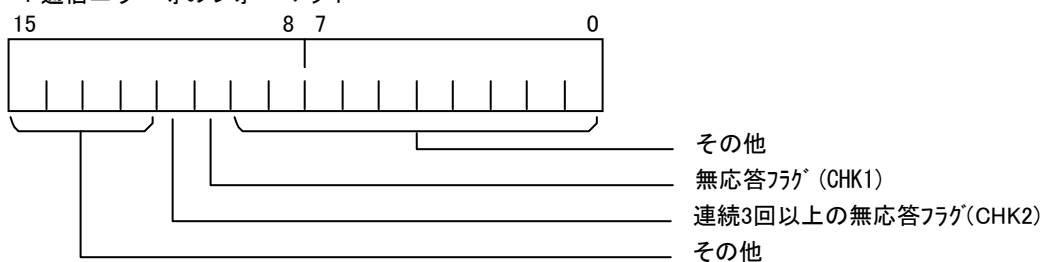
Status : レスポンス（ステータス含）のフォーマット



Data : 位置データのフォーマット



ComErr : 通信エラー等のフォーマット



\* ステータス詳細は別紙「ユーザズマニュアル [ 位置決めユニット ソフト編 ]」を参照して下さい。

**戻り値**

AL\_ER\_OK : 正常  
 AL\_ER\_NOTOPEN : オープンされていません  
 AL\_ER\_INICONFIG : ini ファイル記述に誤りがあります  
 AL\_ER\_INVALIDPARAM : 無効な引数です  
 AL\_ER\_NOTREPLY : 応答が返ってきていません

**説明**

ロジカル ID で指定した位置決めユニットからのレスポンスとステータス・位置データ・通信エラーを取得します。

ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

無応答フラグ (CHK1) とは、1 回でも無応答があれば立ちます。

連続 3 回以上の無応答フラグ (CHK2) とは、無応答状態 (CHK1) が 3 回以上連続で発生した場合に立ちます。

CHK1、CHK2 とも復帰すると消えます。

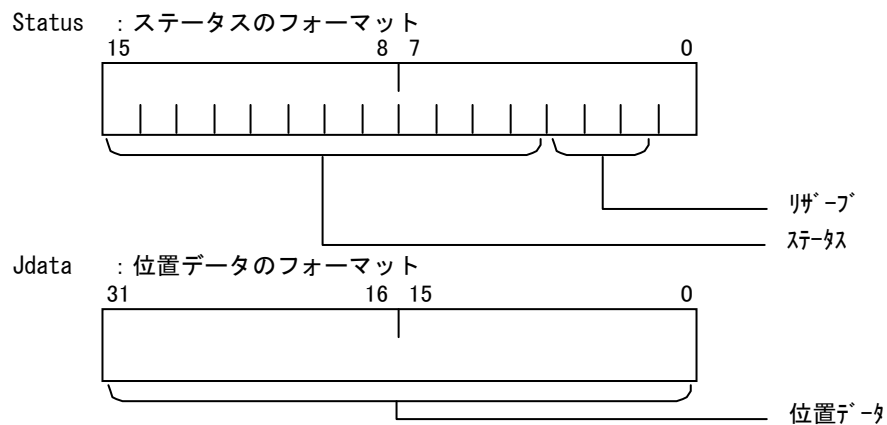
## AL\_AXSA\_GetStatus 関数

**機能** 位置決めユニットから常時アップロード中のステータスと位置データを取得します

**書式** `int AL_AXSA_GetStatus(int Lid, int JNo, WORD *Status, long *JData)`

**引数**

Lid : ロジカル ID  
 JNo : 軸番号 (1- )  
 Status : ステータスを格納するポインタ  
 JData : 位置データを格納するポインタ



\* ステータス詳細は別紙「ユーザーズマニュアル [位置決めユニット ソフト編]」を参照して下さい。

**戻り値**

AL_ER_OK	: 正常 (ステータス、位置データともに取得)
AL_ER_NOTOPEN	: オープンされていません
AL_ER_INICONFIG	: ini ファイル記述に誤りがあります
AL_ER_INVALIDPARAM	: 無効な引数です
AL_ER_NOTREPLY	: アップロード待ちです
AL_ER_ONLYSTAT	: ステータスのみ取得します
AL_ER_ONLYJDATA	: 位置データのみ取得します

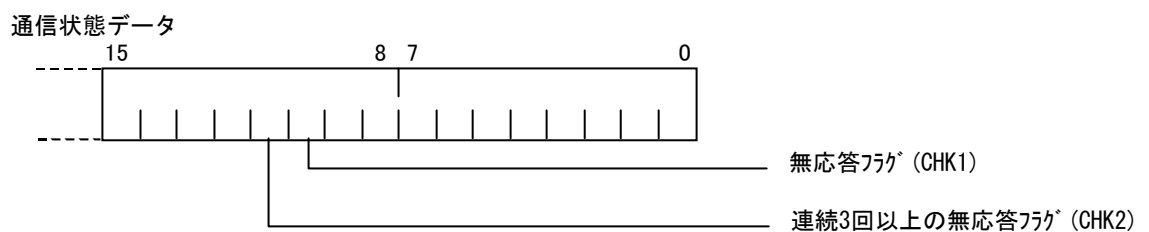
**説明** ロジカル ID で指定した位置決めユニットから常時アップロード中のステータス、位置データを取得します。  
 ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

## AL\_AXSA\_GetCondition 関数

**機能** 位置決めユニットの A-Link 通信状態を取得します

**書式** int AL\_AXSA\_GetCondition(int Lid, int \*Cond)

**引数** Lid : ロジカル ID  
Cond : 取得する通信状態データを格納するポインタ



**戻り値** AL\_ER\_OK : 正常  
AL\_ER\_NOTOPEN : オープンされていません  
AL\_ER\_INICONFIG : ini ファイル記述に誤りがあります

**説明** ロジカル ID で指定した位置決めユニットの通信状態を取得します。  
ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

無応答フラグ (CHK1) とは、1 回でも無応答があれば立ちます。

連続 3 回以上の無応答フラグ (CHK2) とは、無応答状態 (CHK1) が 3 回以上連続で発生した場合に立ちます。

CHK1、CHK2 とも復帰すると消えます。



## 2-4 エンコーダ・カウンタユニット関数

### AL\_ENC\_Open 関数

---

**機能**

エンコーダ・カウンタユニットをオープンします

**書式**

int AL\_ENC\_Open(int Lid)

**引数**

Lid : ロジカル ID

**戻り値**

- AL\_ER\_OK : 正常
- AL\_ER\_OPENDEVICE : ボードが使用できない状態にあります
- AL\_ER\_ALREADYOPEN : すでにオープンしています
- AL\_ER\_INICONFIG : ini ファイル記述に誤りがあります
- AL\_ER\_CREATESEMAPH : DLL 内部セマフォ生成失敗

**説明**

ロジカル ID で指定したエンコーダ・カウンタユニットをオープンします。  
ロジカル ID は ALink.ini に記述したものを指定しなければなりません。  
この関数コール後、エンコーダ・カウンタユニットにアクセス可能となります。  
エンコーダ・カウンタユニットを使用の際には、必ずコールする必要があります。  
ボード設定に誤りがある場合、全ボードをアクセス不可とします。  
この場合、関数は[AL\_ER\_OPENDEVICE]を返します。

---

AL\_ENC\_Close 関数

---

機能	エンコーダ・カウンタユニットをクローズします		
書式	int AL_ENC_Close(int Lid)		
引数	Lid	: ロジカル ID	
戻り値	AL_ER_OK	: 正常	
	AL_ER_NOTOPEN	: オープンされていません	
	AL_ER_INICONFIG	: ini ファイル記述に誤りがあります	
説明	ロジカル ID で指定したエンコーダ・カウンタユニットをクローズします。 ロジカル ID は ALink.ini に記述したものを指定しなければなりません。 この関数コール後は、エンコーダ・カウンタユニットにアクセス不可になります。 アプリケーションの終了時に必ずコールする必要があります。		

## AL\_ENC\_PutCmd 関数

## 機能

エンコーダ・カウンタユニットへの各種コマンドをセットします

## 書式

int AL\_ENC\_PutCmd(int Lid, int Ch, int Cmd, long Param, WORD \*Status, long \*Data)

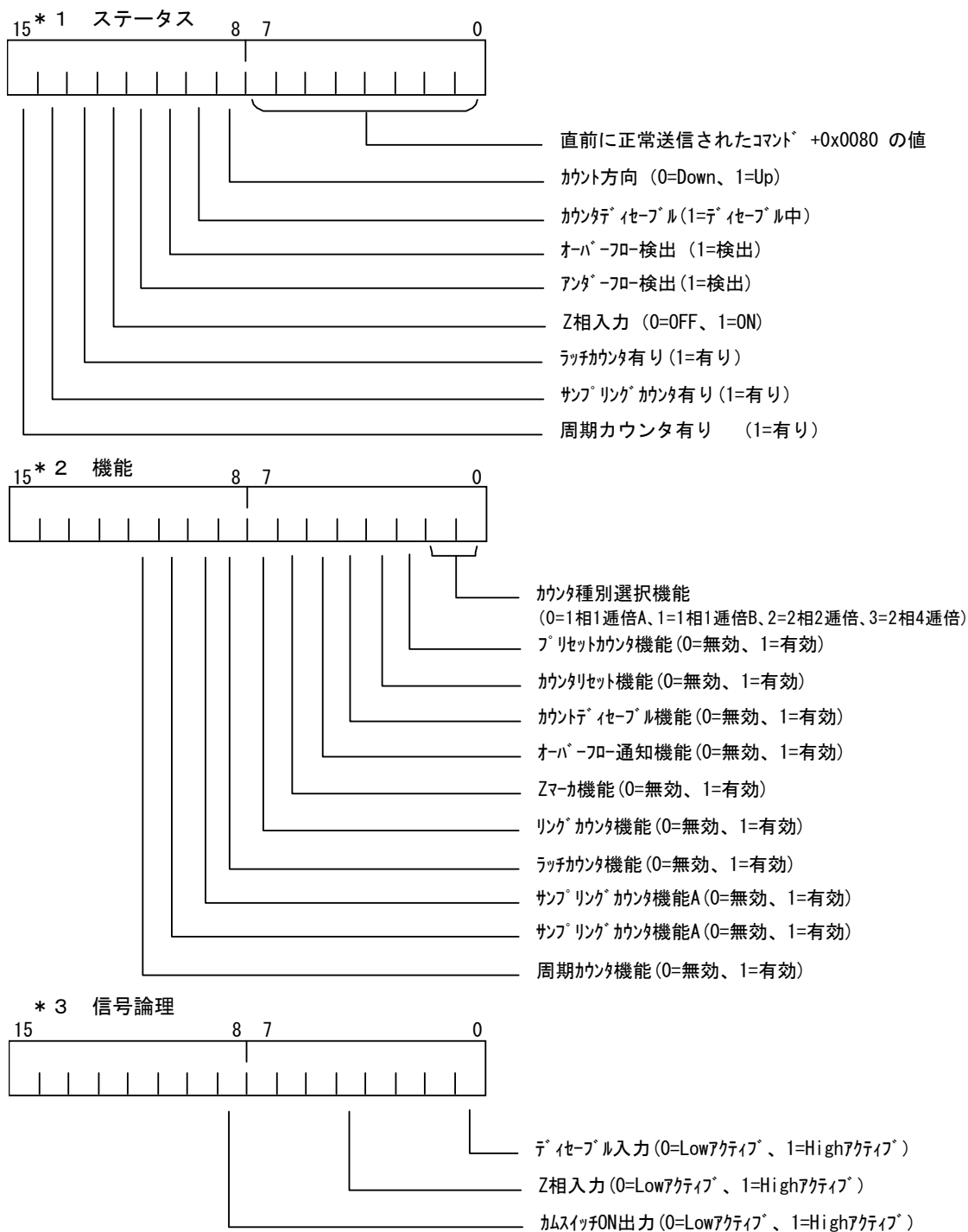
## 引数

Lid : ロジカル ID  
 Ch : チャンネル番号 (1-)  
 Cmd : コマンド  
 Param : パラメータ  
 Status : ステータス(\*1)を格納するポインタ  
 Data : データを格納するポインタ

## コマンドとパラメータ

Cmd	内容	Param	Data
ALENC_CMD_GET_CNT	現在位置応答指令	- - -	カウンタ値
ALENC_CMD_RST_CNT	カウンタリセット指令	カウンタ値	- - -
ALENC_CMD_DBL_CNT	カウンタダブル指令	- - -	- - -
ALENC_CMD_RST_OVRFLW	オーバーフローリセット指令	- - -	- - -
ALENC_CMD_GET_1CYCCNT	1周値読み出し要求	- - -	カウンタ値
ALENC_CMD_GET_LCHCNT	ラッチカウンタ読み出し要求	- - -	カウンタ値
ALENC_CMD_GET_SMPCNT	サンプリングカウンタ読み出し要求	- - -	カウンタ値
ALENC_CMD_GET_CYCLAST	周期カウンタ前値読み出し要求	- - -	カウンタ値
ALENC_CMD_GET_CYCCURR	周期カウンタ現在値読み出し要求	- - -	カウンタ値
ALENC_CMD_SET_RNGCNT	リングカウンタ設定指令	カウンタ値	- - -
ALENC_CMD_SET_CAMCNT	カムスイッチ ON カウンタ設定指令	カウンタ値	- - -
ALENC_CMD_SET_SMPTIM	サンプリングタイマ設定指令	タイマ値	- - -
ALENC_CMD_SET_CYCTIM	周期タイマ設定指令	タイマ値	- - -
ALENC_CMD_GET_RNGCNT	リングカウンタ読み出し要求	- - -	カウンタ値
ALENC_CMD_GET_CAMCNT	カムスイッチ ON カウンタ読み出し要求	- - -	カウンタ値
ALENC_CMD_GET_SMPTIM	サンプリングタイマ読み出し要求	- - -	タイマ値
ALENC_CMD_GET_CYCTIM	周期タイマ読み出し要求	- - -	タイマ値
ALENC_CMD_SET_FUNC	機能設定指令	機能 (*2)	- - -
ALENC_CMD_GET_FUNC	機能読み出し要求	- - -	機能 (*2)
ALENC_CMD_SET_SIG	信号論理設定指令	信号論理 (*3)	- - -
ALENC_CMD_GET_SIG	信号論理読み出し要求	- - -	信号論理 (*3)

- - - : 無視します



戻り値	AL_ER_OK	: 正常
	AL_ER_NOTOPEN	: オープンされていません
	AL_ER_INICONFIG	: ini ファイル記述に誤りがあります
	AL_ER_INVALIDPARAM	: 無効な引数です
	AL_ER_COMMANDTIMEOUT	: コマンドタイムアウト（応答が返ってきません）

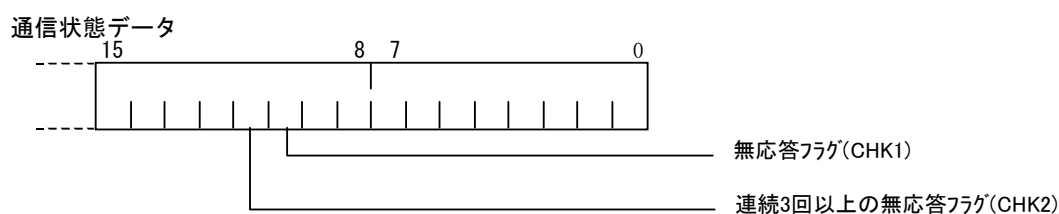
**説明**                    ロジカル ID で指定したエンコーダ・カウンタユニットに各種コマンドを送信します。  
                             ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

## AL\_ENC\_GetCondition 関数

**機能** エンコーダ・カウンタユニットの A-Link 通信状態を取得します

**書式** `int AL_ENC_GetCondition(int Lid, int *Cond)`

**引数** Lid : ロジカル ID  
Cond : 取得する通信状態データを格納するポインタ



**戻り値** AL\_ER\_OK : 正常  
AL\_ER\_NOTOPEN : オープンされていません  
AL\_ER\_INICONFIG : ini ファイル記述に誤りがあります

**説明** ロジカル ID で指定したエンコーダ・カウンタユニットの通信状態を取得します。  
ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

無応答フラグ (CHK1) とは、1 回でも無応答があれば立ちます。  
連続 3 回以上の無応答フラグ (CHK2) とは、無応答状態 (CHK1) が  
3 回以上連続で発生した場合に立ちます。  
CHK1、CHK2 とも復帰すると消えます。

## 2-5 シリアルユニット関数

### AL\_SIO\_Open 関数

---

**機能** シリアルユニットをオープンします

**書式** int AL\_SIO\_Open(int Lid)

**引数** Lid : シリアルロジカル ID

**戻り値**

AL_ER_OK	: 正常
AL_ER_OPENDEVICE	: ボードが使用できない状態にあります
AL_ER_ALREADYOPEN	: すでにオープンしています
AL_ER_INICONFIG	: ini ファイル記述に誤りがあります
AL_ER_CREATESEMAPH	: DLL 内部セマフォ生成に失敗しました
AL_ER_CREATBUF	: DLL 内部バッファ確保に失敗しました

**説明**

ロジカル ID で指定したシリアルユニットをオープンします。  
ロジカル ID は ALink.ini に記述したものを指定しなければなりません。  
この関数コール後、シリアルユニットにアクセス可能となります。  
シリアルユニットを使用の際には、必ずコールする必要があります。  
ボード設定に誤りがある場合、全ボードをアクセス不可とします。  
この場合、関数は [AL\_ER\_OPENDEVICE] を返します。

---

AL\_SIO\_Close 関数

---

**機能** シリアルユニットをクローズします

**書式** int AL\_SIO\_Close(int Lid)

**引数** Lid : ロジカル ID

**戻り値**

AL_ER_OK	: 正常
AL_ER_NOTOPEN	: オープンされていません
AL_ER_INICONFIG	: ini ファイル記述に誤りがあります

**説明** ロジカル ID で指定したシリアルユニットをクローズします。  
ロジカル ID は ALink.ini に記述したものを指定しなければなりません。  
この関数コール後は、シリアルユニットにアクセス不可になります。  
アプリケーションの終了時に必ずコールする必要があります。



## AL\_SIO\_Config 関数

**機能** シリアル通信の通信設定を行います

**書式** int AL\_SIO\_Config(int Lid, int Ch, int Boud, int Bit, int Parity, int Stop,  
int RcvBuf, int SndBuf)

**引数**

Lid	: ロジカル ID
Ch	: チャンネル番号 (1- )
Boud	: ボーレート設定
Bit	: データビット設定
Parity	: パリティ設定
Stop	: ストップビット設定
RcvBuf	: 受信バッファサイズ (0 ~ 40960)
SndBuf	: 送信バッファサイズ (0 ~ 40960)

## ボーレート設定

Boud	内容
ALSIO_CONF_BOUD4800	4800 bps
ALSIO_CONF_BOUD9600	9600 bps
ALSIO_CONF_BOUD19200	19200 bps
ALSIO_CONF_BOUD38400	38400 bps

## データビット設定

Bit	内容
ALSIO_CONF_BIT7	7ビット
ALSIO_CONF_BIT8	8ビット

## パリティ設定

Parity	内容
ALSIO_CONF_PTYNOT	なし
ALSIO_CONF_PTYEVEN	偶数
ALSIO_CONF_PTYODD	奇数

## ストップビット設定

Stop	内容
ALSIO_CONF_STP1	1ビット
ALSIO_CONF_STP2	2ビット

戻り値	AL_ER_OK	: 正常
	AL_ER_NOTOPEN	: オープンされていません
	AL_ER_INICONFIG	: ini ファイル記述に誤りがあります
	AL_ER_INVALIDPARAM	: 無効な引数です

**説明**

ロジカル ID で指定したシリアルユニットの通信設定を行います。  
この関数を呼ぶことで、DLL 内部の受信スレッドが起動し受信が開始されます。  
シリアルユニットを使用する場合は必ず呼ぶようにします。  
受信、送信のバッファはデフォルトで 4096 となっています。これ以下を指定した場合、バッファサイズは 4096 に設定されます。また、40960 を超えた値を指定した場合は 40960 となります。受信バッファはシステムで唯一のバッファとして作成されるため、複数プロセスでの使用の場合は、必ず同じサイズを指定するようにして下さい。  
ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

---

AL\_SIO\_PutData 関数

---

**機能**

シリアル通信でバイト列を送信します

**書式**

```
int AL_SIO_PutData(int Lid, int Ch, char *Buffer, int Size, int *RetSize, int TimeOut)
```

**引数**

Lid : ロジカル ID  
Ch : チャンネル番号 (1- )  
Buffer : 送信するデータバッファの先頭ポインタ  
Size : 送信するデータのサイズ  
RetSize : 実際に送信したデータのサイズを格納するポインタ  
TimeOut : タイムアウトまでの時間 [ms] (0 の時無限待ち)

**戻り値**

AL_ER_OK	: 正常
AL_ER_NOTOPEN	: オープンされていません
AL_ER_INICONFIG	: ini ファイル記述に誤りがあります
AL_ER_INVALIDPARAM	: 無効な引数です
AL_ER_COMMAND	: 通信エラーが発生しました
AL_ER_NOTREPLY	: ユニットとの通信に異常がありました
AL_ER_COMMANDTIMEOUT	: タイムアウトで終了しました

**説明**

ロジカル ID で指定したシリアルユニットを介したシリアル通信を用いてデータの送信を行います。

ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

データ送信が成功した場合は、送信したデータのサイズを RetSize に格納し [AL\_ER\_OK] を返します。

各種エラーが生じた場合は、エラー発生までに送信したデータサイズを RetSize に格納します。

[AL\_ER\_COMMAND] で通信エラー状態となった場合は、ClearError 関数でエラーを解除するようにして下さい。

---

AL\_SIO\_GetData 関数

---

**機能**

シリアル通信でバイト列を受信します

**書式**

```
int AL_SIO_GetData(int Lid, int Ch, char *Buffer, int Size, int *RetSize, int TimeOut)
```

**引数**

Lid : ロジカル ID  
Ch : チャンネル番号 (1- )  
Buffer : 受信データを格納するバッファの先頭ポインタ  
Size : 受信するデータのサイズ  
RetSize : 実際に受信したデータのサイズを格納するポインタ  
TimeOut : タイムアウトまでの時間 [ms] (0 の時無限待ち)

**戻り値**

AL_ER_OK	: 正常
AL_ER_NOTOPEN	: オープンされていません
AL_ER_INICONFIG	: ini ファイル記述に誤りがあります
AL_ER_INVALIDPARAM	: 無効な引数です
AL_ER_COMMAND	: 通信エラーが発生しました
AL_ER_NOTREPLY	: ユニットとの通信に異常がありました
AL_ER_COMMANDTIMEOUT	: タイムアウトで終了しました

**説明**

ロジカル ID で指定したシリアルユニットのチャンネルの受信バッファからデータを取得します。  
ロジカル ID は ALink.ini に記述したものを指定しなければなりません。  
データ受信が成功した場合は、受信したデータのサイズを RetSize に格納し [AL\_ER\_OK] を返します。  
各種エラーが生じた場合は、エラー発生までに受信したデータサイズを RetSize に格納します。  
[AL\_ER\_COMMAND] で通信エラー状態となった場合は、ClearError 関数でエラーを解除するようにして下さい。

---

AL\_SIO\_ClearError 関数

---

**機能** シリアルユニットの通信エラー状態を解除します

**書式** int AL\_SIO\_ClearError(int Lid, int Ch)

**引数** Lid : ロジカル ID  
Ch : 軸番号 (1- )

**戻り値** AL\_ER\_OK : 正常  
AL\_ER\_NOTOPEN : オープンされていません  
AL\_ER\_INICONFIG : ini ファイル記述に誤りがあります  
AL\_ER\_INVALIDPARAM : 無効な引数です

**説明** ロジカル ID で指定したシリアルユニットの通信エラーを解除します。  
エラー解除を実行する際には、データ受送信を終了してから行って下さい。  
ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

---

AL\_SIO\_CheckBuffer 関数

---

**機能** DLL 内部の受信バッファに格納されたデータのサイズを取得します

**書式** `int AL_SIO_CheckBuffer(int Lid, int Ch, int *Size)`

**引数**

Lid	: ロジカル ID
Ch	: 軸番号 (1- )
Size	: データサイズを格納するポインタ

**戻り値**

AL_ER_OK	: 正常
AL_ER_NOTOPEN	: オープンされていません
AL_ER_INICONFIG	: ini ファイル記述に誤りがあります
AL_ER_INVALIDPARAM	: 無効な引数です

**説明** DLL 内部の受信バッファに格納されたデータのサイズを取得します。  
受信中は不定値となります。  
ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

---

AL\_SIO\_ClearBuffer 関数

---

**機能**

DLL 内部の受信バッファに格納されたデータを削除します

**書式**

```
int AL_SIO_ClearBuffer(int Lid, int Ch)
```

**引数**

Lid : ロジカル ID  
Ch : 軸番号 (1- )

**戻り値**

AL_ER_OK	: 正常
AL_ER_NOTOPEN	: オープンされていません
AL_ER_INICONFIG	: ini ファイル記述に誤りがあります
AL_ER_INVALIDPARAM	: 無効な引数です

**説明**

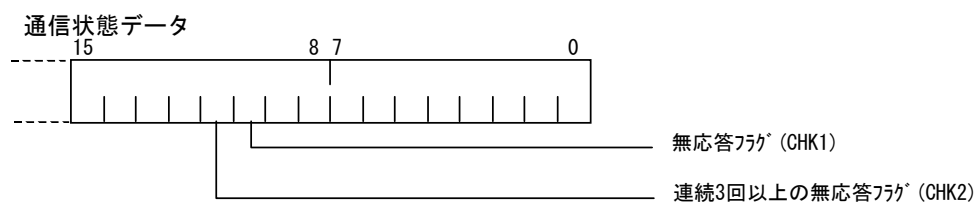
DLL 内部の受信バッファに格納されたデータを削除します。  
受信中の呼び出しは行わないで下さい。  
ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

## AL\_SIO\_GetCondition 関数

**機能** シリアルユニットの A-Link 通信状態を取得します

**書式** int AL\_SIO\_GetCondition(int Lid, int \*Cond)

**引数** Lid : ロジカル ID  
Cond : 取得する通信状態データを格納するポインタ



**戻り値** AL\_ER\_OK : 正常  
AL\_ER\_NOTOPEN : オープンされていません  
AL\_ER\_INICONFIG : ini ファイル記述に誤りがあります

**説明** ロジカル ID で指定したシリアルユニットの通信状態を取得します。  
ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

無応答フラグ (CHK1) とは、1 回でも無応答があれば立ちます。  
連続 3 回以上の無応答フラグ (CHK2) とは、無応答状態 (CHK1) が  
3 回以上連続で発生した場合に立ちます。  
CHK1、CHK2 とも復帰すると消えます。



## 2-6 ちび丸君シリーズ アナログ入出力 ユニット関数

### AL\_ADAC\_Open 関数

---

**機能**           ちび丸君シリーズ アナログ入出力ユニットをオープンします

**書式**           int AL\_ADAC\_Open(int Lid)

**引数**           Lid       : ロジカル ID

**戻り値**

AL_ER_OK	: 正常
AL_ER_OPENDEVICE	: ボードが使用できない状態にあります
AL_ER_ALREADYOPEN	: すでにオープンしています
AL_ER_INICONFIG	: ini ファイル記述に誤りがあります
AL_ER_CREATESEMAPH	: DLL 内部セマフォ生成に失敗しました

**説明**

ロジカル ID で指定したちび丸君シリーズ アナログ入出力ユニットをオープンします。  
ロジカル ID は ALink.ini に記述したものを指定しなければなりません。  
この関数コール後、指定したちび丸君シリーズ アナログ入出力ユニットにアクセス可能となります。

ちび丸君シリーズ アナログ入出力ユニットを使用の際には、必ずコールする必要があります。  
ボード設定に誤りがある場合、全ボードをアクセス不可とします。  
この場合、関数は[AL\_ER\_OPENDEVICE]を返します。

---

AL\_ADAC\_Close 関数

---

**機能**           ちび丸君シリーズ アナログ入出力ユニットをクローズします

**書式**           int AL\_ADAC\_Close(int Lid)

**引数**           Lid       : ロジカル ID

**戻り値**       AL\_ER\_OK                               : 正常  
              AL\_ER\_NOTOPEN                        : オープンされていません  
              AL\_ER\_INICONFIG                     : ini ファイル記述に誤りがあります

**説明**           ロジカル ID で指定したちび丸君シリーズ アナログ入出力ユニットをクローズします。  
              ロジカル ID は ALink.ini に記述したものを指定しなければなりません。  
              この関数コール後は、指定したちび丸君シリーズ アナログ入出力ユニットにアクセス不可になります。  
              アプリケーションの終了時に必ずコールする必要があります。

---

AL\_ADAC\_InValue 関数

---

**機能**           ちび丸君シリーズ アナログ入力ユニットから AD 値を取得します

**書式**           int AL\_ADAC\_InValue(int Lid, int Ch, int \*InVal)

**引数**

Lid	: ロジカル ID
Ch	: チャンネル番号 (1- )
InVal	: AD 値を格納するポインタ

**戻り値**

AL_ER_OK	: 正常
AL_ER_NOTOPEN	: オープンされていません
AL_ER_INICONFIG	: ini ファイル記述に誤りがあります
AL_ER_INVALIDPARAM	: 無効な引数です

**説明**

ロジカル ID で指定したちび丸君シリーズ アナログ入力ユニットの指定したチャンネルから AD データを取得します。

InVal に格納される値は A/D 変換後の 12 ビットデータとなります。

ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

---

AL\_ADAC\_OutValue 関数

---

**機能**

ちび丸君シリーズ アナログ出力ユニットに DA 値を出力します

**書式**

```
int AL_ADAC_OutValue(int Lid, int Ch, int OutVal)
```

**引数**

Lid : ロジカル ID  
Ch : チャンネル番号 (1- )  
OutVal : 出力する DA 値

**戻り値**

AL_ER_OK	: 正常
AL_ER_NOTOPEN	: オープンされていません
AL_ER_INICONFIG	: ini ファイル記述に誤りがあります
AL_ER_INVALIDPARAM	: 無効な引数です

**説明**

ロジカル ID で指定したちび丸君シリーズ アナログ出力ユニットの指定したチャンネルに DA 値を出力します。  
OutVal の値は D/A 変換前の 12 ビットデータとなります。  
ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

# AL\_ADAC\_SetFilter 関数

**機能**           ちび丸君シリーズ アナログ入力ユニットの AD 入力フィルタ設定を行います

**書式**           int AL\_ADAC\_SetFilter(int Lid, int Ch, int Filter)

**引数**

Lid       : ロジカル ID

Ch       : チャンネル番号 (1- )

Filter   : フィルタタイプ

## フィルタタイプ

Filter	内容
ALADAC_FLTR_NOT	フィルタ なし
ALADAC_FLTR_MOVE	フィルタ 移動平均

**戻り値**

AL\_ER\_OK                               : 正常

AL\_ER\_NOTOPEN                        : オープンされていません

AL\_ER\_INICONFIG                      : ini ファイル記述に誤りがあります

AL\_ER\_INVALIDPARAM                  : 無効な引数です

AL\_ER\_COMMANDTIMEOUT                : コマンドタイムアウト (応答が返ってきません)

**説明**

ロジカル ID で指定したちび丸君シリーズ アナログ入力ユニットの AD 入力フィルタ設定を行います。

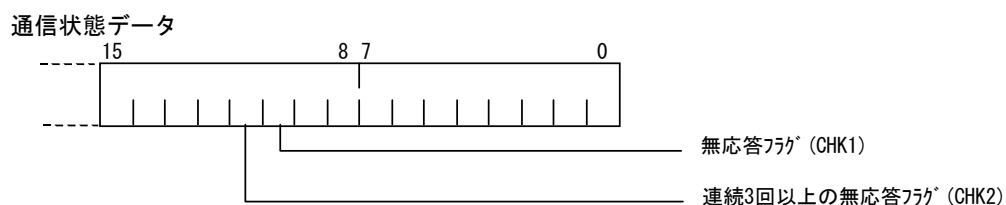
ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

## AL\_ADAC\_GetCondition 関数

**機能**           ちび丸君シリーズ アナログ入出力ユニットの A-Link 通信状態を取得します

**書式**           int AL\_ADAC\_GetCondition(int Lid, int \*Cond)

**引数**           Lid       : ロジカル ID  
                   Cond     : 取得する通信状態データを格納するポインタ



**戻り値**       AL\_ER\_OK                               : 正常  
                   AL\_ER\_NOTOPEN                    : オープンされていません  
                   AL\_ER\_INICONFIG                 : ini ファイル記述に誤りがあります

**説明**           ロジカル ID で指定したちび丸君シリーズ アナログ入出力ユニットの通信状態を取得します。  
                   ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

無応答フラグ (CHK1) とは、1 回でも無応答があれば立ちます。  
 連続 3 回以上の無応答フラグ (CHK2) とは、無応答状態 (CHK1) が  
 3 回以上連続で発生した場合に立ちます。  
 CHK1、CHK2 とも復帰すると消えます。

## 2-7 ALD シリーズ アナログ入出力ユニット関数

### AL\_ADAD\_Open 関数

---

機能	ALD シリーズ アナログ入出力ユニットをオープンします		
書式	int AL_ADAD_Open(int Lid)		
引数	Lid	:	ロジカル ID
戻り値	AL_ER_OK	:	正常
	AL_ER_OPENDEVICE	:	ボードが使用できない状態にあります
	AL_ER_ALREADYOPEN	:	すでにオープンしています
	AL_ER_INICONFIG	:	ini ファイル記述に誤りがあります
	AL_ER_CREATESEMAPH	:	ライブラリ内部セマフォ生成に失敗しました
説明	<p>ロジカル ID で指定したアナログ入出力ユニットをオープンします。 ロジカル ID は ALink.ini に記述したものを指定しなければなりません。 この関数コール後、指定したアナログ入出力ユニットにアクセス可能となります。 アナログ入出力ユニットを使用の際には、必ずコールする必要があります。 ボード設定に誤りがある場合、全ボードをアクセス不可とします。 この場合、関数は [AL_ER_OPENDEVICE] を返します。</p>		

---

AL\_ADAD\_Close 関数

---

機能	ALD シリーズ アナログ入出力ユニットをクローズします		
書式	int AL_ADAD_Glose(int Lid)		
引数	Lid	: ロジカル ID	
戻り値	AL_ER_OK	: 正常	
	AL_ER_NOTOPEN	: オープンされていません	
	AL_ER_INICONFIG	: ini ファイル記述に誤りがあります	
説明	ロジカル ID で指定したアナログ入出力ユニットをクローズします。 ロジカル ID は ALink.ini に記述したものを指定しなければなりません。 この関数コール後は、指定したアナログ入出力ユニットにアクセス不可になります。 アプリケーションの終了時に必ずコールする必要があります。		



---

AL\_ADAD\_InValue 関数

---

**機能**

ALD シリーズ アナログ入出力ユニットから AD 値を取得します

**書式**

```
int AL_ADAD_InValue(int Lid, int Ch, int *InVal, int Mode)
```

**引数**

Lid : ロジカル ID  
Ch : チャンネル番号 (1- )  
InVal : AD 値を格納するポインタ  
Mode : スキャンモード (0 : 通常 1 : 高速) ディップスイッチにあわせてます

**戻り値**

AL\_ER\_OK : 正常  
AL\_ER\_NOTOPEN : オープンされていません  
AL\_ER\_INICONFIG : ini ファイル記述に誤りがあります  
AL\_ER\_INVALIDPARAM : 無効な引数です

**説明**

ロジカル ID で指定したアナログ入出力ユニットの指定したチャンネルから AD データを取得します。  
InVal に格納される値は A/D 変換後の 13 ビットデータとなります。  
ロジカル ID は ALink.ini に記述したものを指定しなければなりません

---

AL\_ADAD\_OutValue 関数

---

**機能**

ALD シリーズ アナログ入出力ユニットに DA 値を出力します

**書式**

```
int AL_ADAD_OutValue(int Lid, int Ch, int OutVal, int Mode)
```

**引数**

Lid : ロジカル ID  
Ch : チャンネル番号 (1- )  
OutVal : 出力する DA 値  
Mode : スキャンモード (0 : 通常 1 : 高速) ディップスイッチにあわせてます

**戻り値**

AL\_ER\_OK : 正常  
AL\_ER\_NOTOPEN : オープンされていません  
AL\_ER\_INICONFIG : ini ファイル記述に誤りがあります  
AL\_ER\_INVALIDPARAM : 無効な引数です

**説明**

ロジカル ID で指定したアナログ入出力ユニットの指定したチャンネルに DA 値を出力します。  
OutVal の値は D/A 変換前の 13 ビットデータとなります。  
ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

---

AL\_ADAD\_InValueFull 関数

---

**機能** ALD シリーズ アナログ入出力ユニットから 16 ビットのデータを全て取得します

**書式** `int AL_ADAD_InValueFull(int Lid, int Ch, unsigned short *InDat)`

**引数**

Lid	: ロジカル ID
Ch	: チャンネル番号 (1- )

注) スキャンモードが「通常」のとき、チャンネル番号は「1」のみ有効です

InDat : 16 ビットデータを格納するポインタ

**戻り値**

AL_ER_OK	: 正常
AL_ER_NOTOPEN	: オープンされていません
AL_ER_INICONFIG	: ini ファイル記述に誤りがあります
AL_ER_INVALIDPARAM	: 無効な引数です

**説明** ロジカル ID で指定したアナログ入出力ユニットの指定した AD チャンネルから 16 ビットのデータを取得します。  
ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

---

AL\_ADAD\_OutValueFull 関数

---

**機能** ALD シリーズ アナログ入出力ユニットへ 16 ビットのデータを出力します

**書式** int AL\_ADAD\_OutValueFull(int Lid, int Ch, unsigned short OutDat)

**引数** Lid : ロジカル ID  
Ch : チャンネル番号 (1- )

注) スキャンモードが「通常」のとき、チャンネル番号は「1」のみ有効です

OutDat : 16 ビット出力データ

**戻り値** AL\_ER\_OK : 正常  
AL\_ER\_NOTOPEN : オープンされていません  
AL\_ER\_INICONFIG : ini ファイル記述に誤りがあります  
AL\_ER\_INVALIDPARAM : 無効な引数です

**説明** ロジカル ID で指定したアナログ入出力ユニットの指定した DA チャンネルに 16 ビットのデータ  
を出力します。  
ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

---

AL\_ADAD\_InOValue 関数

---

**機能** ALD シリーズ アナログ入出力ユニットの出力中の DA 値を取得します

**書式** int AL\_ADAD\_InOValue(int Lid, int Ch, int \*InVal)

**引数**

Lid	: ロジカル ID
Ch	: チャンネル番号 (1- )
InVal	: DA 値を格納するポインタ

注) スキャンモードが「高速」のときのみ使用可能です

**戻り値**

AL_ER_OK	: 正常
AL_ER_NOTOPEN	: オープンされていません
AL_ER_INICONFIG	: ini ファイル記述に誤りがあります
AL_ER_INVALIDPARAM	: 無効な引数です

**説明** ロジカル ID で指定したアナログ入出力ユニットの指定した DA チャンネルから出力中の DA 値を取得します。  
ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

---

AL\_ADAD\_In0ValueFull 関数

---

**機能** ALD シリーズ アナログ入出力ユニットの出力中の DA 値を 16 ビット全て取得します

**書式** `int AL_ADAD_In0ValueFull(int Lid, int Ch, unsigned short *InDat)`

**引数**

Lid	: ロジカル ID
Ch	: チャンネル番号 (1- )
InDat	: 16 ビットデータを格納するポインタ

注) スキャンモードが「通常」のとき、チャンネル番号は「1」のみ有効です

**戻り値**

AL_ER_OK	: 正常
AL_ER_NOTOPEN	: オープンされていません
AL_ER_INICONFIG	: ini ファイル記述に誤りがあります
AL_ER_INVALIDPARAM	: 無効な引数です

**説明** ロジカル ID で指定したアナログ入出力ユニットの指定した DA チャンネルから出力中の 16 ビットのデータを取得します。  
ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

## AL\_ADAD\_SetFilter 関数

**機能**

ALD シリーズ アナログ入出力ユニットの AD 入力フィルタ設定を行います

**書式**

```
int AL_ADAD_SetFilter(int Lid, int Ch, int Filter, int Cut, int Sample, int Mode)
```

**引数**

Lid : ロジカル ID  
 Ch : チャンネル番号 (1- )  
 Filter : フィルタタイプ  
 Cut : 最大/最小値除外  
 Sample : サンプル回数 (0-6)  
 Mode : スキャンモード (0 : 通常 1 : 高速) ディップスイッチにあわせてます

## フィルタタイプ

Filter	内容
ALADA_FLTR_KIND_NOT	フィルタ なし
ALADA_FLTR_KIND_SIMP	フィルタ 単純平均
ALADA_FLTR_KIND_MOVE	フィルタ 移動平均

## 最大/最小除外

Cut	内容
ALADA_FLTR_MXMN_NOT	最大/最小値除外しない
ALADA_FLTR_MXMN_CUT	最大/最小値除外する

## サンプル回数 (0-6)

2 の (Smple + 1) 乗で設定される

**戻り値**

AL\_ER\_OK : 正常  
 AL\_ER\_NOTOPEN : オープンされていません  
 AL\_ER\_INICONFIG : ini ファイル記述に誤りがあります  
 AL\_ER\_INVALIDPARAM : 無効な引数です  
 AL\_ER\_COMMANDTIMEOUT : コマンドタイムアウト (応答が返ってきません)

**説明**

ロジカル ID で指定したアナログ入出力ユニットのフィルタ設定を行います。  
 ロジカル ID は ALink.ini に記述したものを指定しなければなりません。

## AL\_ADAD\_GetFilter 関数

**機能**

ALD シリーズ アナログ入出力ユニットの AD 入力フィルタ設定を取得します

**書式**

```
int AL_ADAD_GetFilter(int Lid, int Ch, int *Filter, int *Cut, int *Sample, int Mode)
```

**引数**

Lid : ロジカル ID  
 Ch : チャンネル番号 (1- )  
 Filter : フィルタタイプを格納するポインタ  
 Cut : 最大/最小値除外を格納するポインタ  
 Sample : サンプル回数を格納するポインタ  
 Mode : スキャンモード (0 : 通常 1 : 高速) ディップスイッチにあわせませす

## フィルタタイプ

Filter	内容
ALADA_FLTR_KIND_NOT	フィルタ なし
ALADA_FLTR_KIND_SIMP	フィルタ 単純平均
ALADA_FLTR_KIND_MOVE	フィルタ 移動平均

## 最大/最小除外

Cut	内容
ALADA_FLTR_MXMN_NOT	最大/最小値除外しない
ALADA_FLTR_MXMN_CUT	最大/最小値除外する

## サンプル回数 (0-6)

2 の (Smple + 1) 乗で設定される

**戻り値**

AL\_ER\_OK : 正常  
 AL\_ER\_NOTOPEN : オープンされていません  
 AL\_ER\_INICONFIG : ini ファイル記述に誤りがあります  
 AL\_ER\_INVALIDPARAM : 無効な引数です  
 AL\_ER\_COMMANDTIMEOUT : コマンドタイムアウト (応答が返ってきません)

**説明**

ロジカル ID で指定したアナログ入出力ユニットのフィルタ設定を行います。  
 ロジカル ID は ALink.ini に記述したものを指定しなければなりません。



---

AL\_ADAD\_GetCondition 関数

---

<b>機能</b>	ALD シリーズ アナログ入出力ユニットの A-Link 通信状態を取得します		
<b>書式</b>	int AL_ADAD_GetCondition(int Lid, int *Cond, int Mode)		
<b>引数</b>	Lid	: ロジカル ID	
	Cond	: 取得する通信状態データを格納するポインタ	
	Mode	: スキャンモード (0: 通常 1: 高速) ディップスイッチにあわせてます	
<b>戻り値</b>	AL_ER_OK	: 正常	
	AL_ER_NOTOPEN	: オープンされていません	
	AL_ER_INICONFIG	: ini ファイル記述に誤りがあります	
<b>説明</b>	ロジカル ID で指定したアナログ入出力ユニットの通信状態を取得します。 ロジカル ID は ALink.ini に記述したものを指定しなければなりません。 連続 3 回以上の無応答が発生した場合に 1 となり、復帰すると 0 となります。		

## 2-8 制御補助関数

### AL\_BeginPeriod 関数

---

**機能** Windows システムタイマの分解能を変更します

**書式** `int AL_BeginPeriod(void)`

**引数** なし

**戻り値** AL\_ER\_OK : 正常  
上記以外 : エラー

**説明** Windows システムタイマの精度を変更します。  
この関数では、内部で Win32API 関数 `timeBegeinPeriod()` をコールし Window 内部タイマの分解能を最小の 1[ms] に設定します。  
特に内部でハンドシェイクを持つ軸ユニット制御関数では関数コールの所要時間が変化します。  
この関数をコールした場合は必ず `AL_EndPeriod` 関数を呼びタイマを戻すようにして下さい。  
`AL_ER_OK` 以外の戻り値はエラーとなります。エラーコードは `timeBeginPeriod()` のものと同様となります。

---

AL\_EndPeriod 関数

---

**機能** 変更した Windows システムタイマを標準のものに戻します

**書式** int AL\_EndPeriod(void)

**引数** なし

**戻り値** AL\_ER\_OK : 正常  
上記以外 : エラー

**説明** 変更した Windows システムタイマを標準のものに戻します。  
この関数では、内部で Win32API 関数 timeEndPeriod() をコールし Window 内部タイマの分解能を Windows 標準のものに戻します。  
AL\_BeginPeriod 関数を使用した場合、必ずこの関数でタイマを戻すようにします。  
AL\_ER\_OK 以外の戻り値はエラーとなります。エラーコードは timeEndPeriod() のものと同様となります。

---

AL\_CheckBoard 関数

---

<b>機能</b>	PCI ボード、PC カードのオープンの状態を調べます	
<b>書式</b>	int AL_CheckBoard(int Board)	
<b>引数</b>	Board	: ボード番号 (1-4)
<b>戻り値</b>	AL_ER_OK	: 正常
	AL_ER_LOADDEVICE	: 下位 DLL ロードに失敗しました
	AL_ER_OPENDEVICE	: ボードのオープンに失敗しました
	AL_ER_INVALIDPARAM	: 無効な引数です
	AL_ER_INICONFIG	: 設定ファイルに誤りがあります
<b>説明</b>	PCI ボード、PC カードの状態を調べます。 正常である場合、ALink.ini で記された状態で利用可能であることを示します。	

## AL\_CheckStatusLED 関数

**機能** PCI ボード、PC カードの LED の状態を調べます。

**書式** int AL\_CheckStatusLED(int Board, int Line, WORD \*Status)

**引数**

Board	: ボード番号
Line	: ライン番号
Status	: LED ステータス (返値) を格納するポインタ

## LED ステータス

D0	: CHK1 エラー	(0=エラーなし、1=エラー発生)
D1	: CHK2 エラー	(0=エラーなし、1=エラー発生)

**戻り値**

AL_ER_OK	: 正常
AL_ER_LOADDEVICE	: 下位 DLL ロードに失敗しました
AL_ER_OPENDEVICE	: ボードのオープンに失敗しました
AL_ER_INVALIDPARAM	: 無効な引数です
AL_ER_INICONFIG	: 設定ファイルに誤りがあります

**説明** PCI ボード、PC カードの LED の状態を調べます。  
正常である場合、LED ステータスを返します。

---

AL\_ClearStatusLED 関数

---

<b>機能</b>	PCI ボード、PC カードの LED を消灯させます。	
<b>書式</b>	int AL_ClearStatusLED(int Board, int Line)	
<b>引数</b>	Board	: ボード番号
	Line	: ライン番号
<b>戻り値</b>	AL_ER_OK	: 正常
	AL_ER_LOADDEVICE	: 下位 DLL ロードに失敗しました
	AL_ER_OPENDEVICE	: ボードのオープンに失敗しました
	AL_ER_INVALIDPARAM	: 無効な引数です
	AL_ER_INICONFIG	: 設定ファイルに誤りがあります
<b>説明</b>	PCI ボード、PC カードの LED を消灯させます。 正常である場合、LED が消灯します。	

---

AL\_GetErrorCount 関数

---

<b>機能</b>	PCI ボード、PC カードのエラーを取得します。(* 1)		
<b>書式</b>	int AL_GetErrorCount(int Board, int Line, WORD ChkNo, WORD *Count)		
<b>引数</b>	Board	: ボード番号	
	Line	: ライン番号	
	ChkNo	: 取得特する CHK 1、CHK2 を指定 (0=CHK1、1=CHK2)	
	Count	: CHK1 または CHK2 のエラー発生カウントを格納するポインタ	
<b>戻り値</b>	AL_ER_OK	: 正常	
	AL_ER_LOADDEVICE	: 下位 DLL ロードに失敗しました	
	AL_ER_OPENDEVICE	: ボードのオープンに失敗しました	
	AL_ER_INVALIDPARAM	: 無効な引数です	
	AL_ER_INICONFIG	: 設定ファイルに誤りがあります	
<b>説明</b>	PCI ボード、PC カードの CHK1、CHK2 のエラー発生カウントを取得します。 エラーは CHK1、2 とも 255 までカウントされ、この関数の取得時にクリアされます。		
	(* 1) A-Link Ver2 通信 IC 搭載ボード使用時のみサポート (PCILZ04-0~PCILZ07-0)		

## このマニュアルについて

- (1) 本書の内容の一部または全部を当社からの事前の承諾を得ることなく、無断で複写、複製、掲載することは固くお断りします。
- (2) 本書の内容に関しては、製品改良のためお断りなく、仕様などを変更することがありますのでご了承下さい。
- (3) 本書の内容に関しては万全を期しておりますが、万一ご不審な点や誤りなどお気づきのことがございましたらお手数ですが巻末記載の弊社までご連絡下さい。その際、巻末記載の書籍番号も併せてお知らせ下さい。

### 改訂履歴

日時	バージョン	変更点
2002.04.01	Rev 1.00	初版
2003.03.25	Rev 1.01	PC カード対応
2003.12.08	Rev 1.02	A-Link Ver2 通信 IC 搭載ボード対応
2003.12.24	Rev 1.03	下位 DLL(A-Link Ver2 通信 IC)関数追加
2004.07.29	Rev 1.04	ちび丸君シリーズアナログ入出力ユニット対応
2005.03.03	Rev 1.10	全面改訂
2006.07.24	Rev 1.11	住所変更(本社)
2007.10.25	Rev 1.12	住所変更
2008.11.12	Rev 1.13	ALD シリーズ アナログ入出力ユニット対応、ロゴ変更
2014.02.26	Rev 1.14	住所表記変更