シリアルデータ to CAN メッセージ変換ユニット

CU-SD1

本書の記載内容は、2023 年 9 月現在のものです。

概要

CU-SD1 は、RS-232 準拠データ出力を持つ機器を、CAN バスに接続しデータ出力を行うため、RS-232 準拠シリアルデータストリームから CAN メッセージ生成を行うユニットです。CU-SD1 に対して、変換を行いたい RS-232 準拠データフォーマットを、シリアルデータ変換条件ファイル.SCC (本書後述) により、事前に設定しておきます。こうすることで、CU-SD1 は、接続された機器からの RS-232 準拠データを受信したタイミングで CAN データに変換して CAN バスに送出することができます。この出力周期は、RS-232 準拠データの受信タイミングに従属するため、CU-SD1 自体の CAN 出力周期設定はできません。

基本機能は、

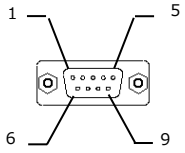
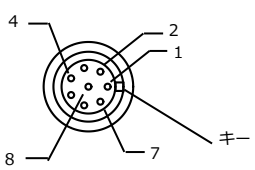
- データストリームからの変換対象データの抽出定義
- 変換データの形式定義
- 接続機器へのデータ転送要求定義

です。

RS-232 準拠信号を出力するセンサなどのデータを CAN メッセージに変換して CAN バスに送信する用途にご利用いただけます。

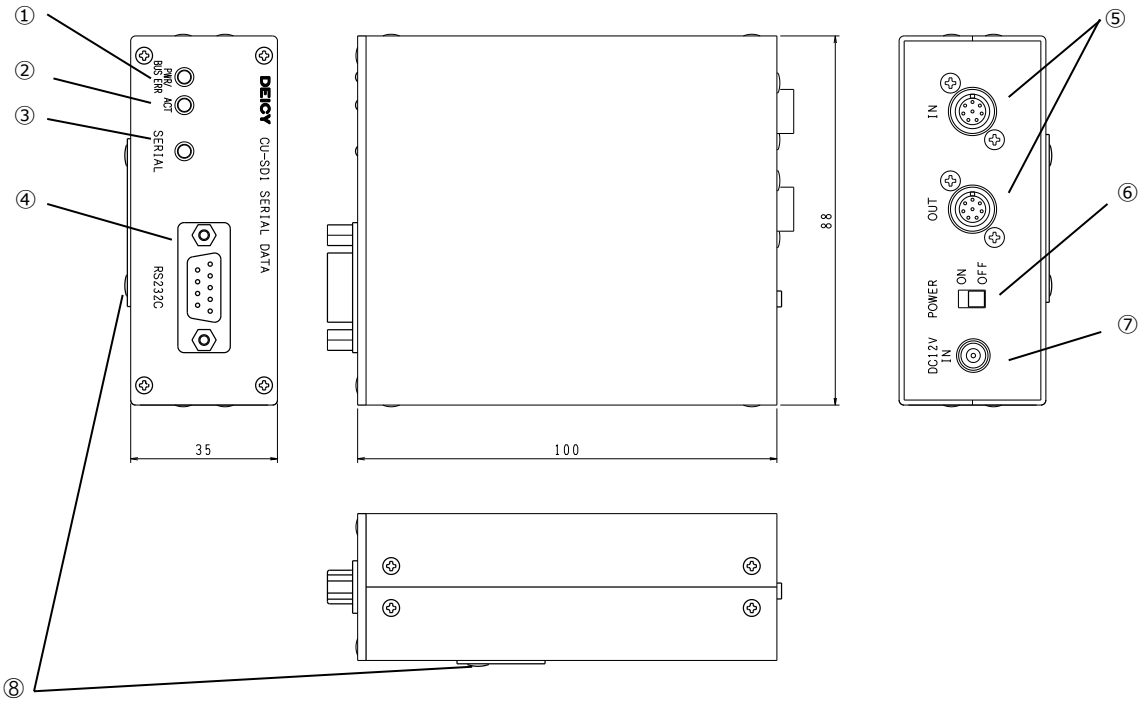
CU-SD1 本体仕様および取扱説明編

仕様

項目	内容
適合 CAN 規格	ISO 11898 CAN 2.0A/B
入力	RS-232 準拠ポート × 1
出力	CAN ポート × 1
転送速度	最大 115.2kbps シリアルデータ変換条件ファイルに従属
パリティ	無し, ODD, EVEN シリアルデータ変換条件ファイルに従属
語長	7, 8bit シリアルデータ変換条件ファイルに従属
ストップ bit	1, 2bit シリアルデータ変換条件ファイルに従属
変換条件ファイル設定	本機の CAN ポートに対して、変換条件ファイル設定モードへの変更メッセージを送信します。 その後、変換条件ファイルの転送は、シリアル通信によります。(CU-SD1 の Serial Port) 注意： シリアルデータ変換条件ファイル(.SCC) の記述内容は後述します。 転送された変換条件ファイルは内部不揮発領域に保持。不揮発領域に変換条件が存在しない場合は機能しません。
サンプリング	シリアルデータ着信基準
外部サンプリング	無し
自走出力 On/Off	ディップスイッチにより設定
CAN ボーレート設定	ディップスイッチにより設定 1Mbps, 500kbps, 250kbps, 125kbps, 83.3kbps, 65.2kbps
コネクタ	入力コネクタ：RS-232 準拠 D-sub 9pin オス コネクタ勘合面視　　2: RxD　3: TxD　5： GND  CAN コネクタ： IN/OUT　ヒロセ MXR-8R-8SA(71) 適合プラグ ヒロセ MXR-8P-8P(71) CAN 信号、同期パルス、電源  1： CAN_L 2： 12 V 3： 0 V 4： 外部同期_L 5： 外部同期_H 6： 0V 7： 12 V 8： CAN_H パネル面 キー位置は図のようにパネル面に向かって右側にあります。 電源ラインを使用する場合は、Pin2/7 Pin3/6 とも配線して下さい。
CAN ターミネータ	ディップスイッチにより設定
CAN メッセージ ID	8bit ディップスイッチにより設定 11bit/拡張 29bit 切り替え対応

	本ユニット固有の CAN ID メッセージ数は、ディップスイッチ設定 ID から ID+4 まで連続使用します。 シリアルデータ変換生成メッセージ数はシリアルデータ変換条件ファイルに従属します。 変換生成メッセージ ID はディップスイッチ設定 ID +5 以降を連続して使用します。 変換生成メッセージ ID は最大 6 個となります。 設定したすべてのメッセージ ID 中に記述できる変換生成シグナル要素数は合計で最大 20 個です。
表示 LED	PWR/BUS ERR 2 色 LED 起動時緑色点灯、ただし CAN エラー時は赤色点灯 PWR 緑色点滅は、変換条件設定モードであることを示します。(変換データの送信は行うことができません。) ACT 1 色 LED CAN メッセージ出力時青色点灯 SERIAL 1 色 LED シリアルデータ変換時データ着信緑色点灯(実質点滅)
電源スイッチ	POWER 小型スライドスイッチ On/Off はユニット内電源の On/Off に対応。CAN バスへの電源は常時供給。
電源	9 V DC ～ 15 V DC 供給方式： CAN バス経由で供給、または DC ジャックに供給 DC ジャック EIAJ RC5320A 適合 電圧区分 4 (CAN コネクタから給電しない場合に使用)
起動時間	電源投入から変換動作開始までの時間、最大 10 秒、または後述する DIP SW4-S13 On 時は設定モードスタンバイまでの時間
外形寸法	88W×35H×100D mm 突起物除く
質量	220 g
使用温度範囲	-20℃～+70℃ 結露無きこと
耐振動特性	別途規定
付属品	説明書、ゴム足

外形図および各部の名称



番号・名称		機能
①	PWR/BUSS ERR	電源表示 LED です。電源 On で緑色点灯、電源 Off で消灯。 緑色点滅は、変換条件設定モードであることを示します。(変換データの送信は行うことができません。) また、CAN エラー状態表示を兼ねます。エラー検出で赤色点灯。
②	ACT	CAN メッセージ出力時青色点灯。
③	SERIAL	シリアルデータ変換時、データ着信時緑色点灯(実質点滅)。
④	RS232C	シリアルデータ入力用 D-sub 9pin オスコネクタです。
⑤	IN/OUT	CAN 通信コネクタです。電源入力も併用できます。それぞれ IN/OUT と記載していますが等価機能を持ちます。
⑥	POWER	電源スイッチです。本体の電源を On/Off します。 本ユニットに入力された電源は、このスイッチの On/Off にかかわらず、IN/OUT コネクタから出力されます。
⑦	DC 12V IN	12 V DC 電源入力ジャックです。
⑧	ディップスイッチ部カバー	各種設定用ディップスイッチ部のカバーです。

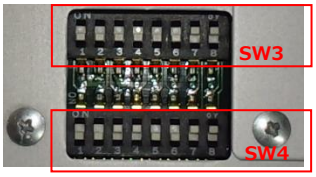
オプション

型式	品名・内容
CK-CU1-0.2	CU ユニット間接続ケーブル　0.2m 電源/外部パルス配線有り 両端 MXR-8P-8P(71)
CK-CU2-0.2	CU ユニット間接続ケーブル　0.2m 電源/外部パルス配線無し 両端 MXR-8P-8P(71)
CK-CU3-F1.5	CAN 通信接続ケーブル　1.5m 電源配線無し D-sub 9pin メス - MXR-8P-8P(71)
CK-CU4-F1.5	CAN 通信接続ケーブル　1.5m 接続した CU シリーズ本体への 12V DC 供給用先バラ分岐ケーブル付き D-sub 9pin メス - MXR-8P-8P(71)
CK-JEITA4L	DC 電源ケーブル先バラ 1.8m コネクタ L 型
US301210	AC アダプタ コネクタストレート

ディップスイッチ設定

設定用ディップスイッチ本体底面部に位置し、カバーを外して設定変更を行います。

↑リアパネル側電源コネクタ部

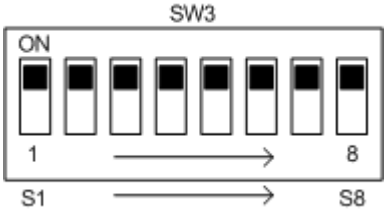


信号入力コネクタ部を手前方向としてカバーを外すと左図のように、上下 2 つのディップスイッチが見えます。
上部のディップスイッチが SW3、下部のディップスイッチが SW4 となります。
設定の変更は、必ず電源を Off にした状態で行って下さい。電源起動時にディップスイッチの情報を読み取り、対応した設定を行います。
下図のディップスイッチは、ノブが上方位置の時 On で 1、下方位置の時 Off で 0 とします。

↓フロントパネル側信号入力コネクタ部

① ベースメッセージ ID 設定関連 SW3

ベースメッセージ ID(各ユニットで使用する基本の CAN メッセージ ID)は、下記表より メッセージ ID = A × (B + C) で表します。



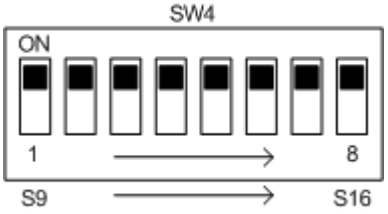
S1			S2 ～ S5		S6 ～ S8	
ディップ SW	意味	A	ディップ SW	B	ディップ SW	C
0	標準 ID	1	0 0 0 0	100	0 0 0	10
1	拡張 ID	10	0 0 0 1	200	0 0 1	20
			0 0 1 0	300	0 1 0	30
			0 0 1 1	400	0 1 1	40
					1 0 0	50
			1 1 0 1	1400	1 0 1	60
			1 1 1 0	1500	1 1 0	70
			1 1 1 1	1600	1 1 1	80

A、B および C は、10 進数表示です。

出荷時設定 00000000

- ベースメッセージ ID を設定するディップスイッチ S2～S8 の設定値が、後述のユニット ID となります。
- ユニット ID は、制御ブロードキャストメッセージにより特定のユニットだけに動作コマンドを送る場合に用います。(後述の「制御メッセージ」参照。)
- 本書で使用する「ブロードキャスト」とは、「同一の制御ブロードキャスト ID」を持つ機器に対してのブロードキャストのことを言います。
- ここで設定したベースメッセージ ID - 1 の番号を持つメッセージ ID は「リモートメッセージ」として本ユニットで予約されるため、同 CAN バス内の他の機器では使用しないで下さい。(リモートメッセージの内容については別途ドキュメントを用意しています。)

② ボーレート他設定関連 SW4



S9 ～ S11		S12		S13 (S14 は 未使用のため下表 X で表記)		S15 S16	
ディップ SW	ボーレート	ディップ SW	自走 On/Off CAN データ連続出力	ディップ SW		ディップ SW	CAN/同期パルス
0 0 0	1 Mbps	0	起動時停止	0X	電源 On 後、変換動作	0 0	終端抵抗 Off
0 0 1	500 kbps	1	CAN データ連続出力	1X	電源 On 後、設定モードに移行	1 1	終端抵抗 On
0 1 0	250 kbps				**<注記>**		
0 1 1	125 kbps				**変換動作を行うには SW13 は必ず**		
1 0 0	83.3 kbps				**0、つまり DIPS-SW Off として**		
1 0 1	62.5 kbps				**ください。**		
1 1 0	62.5 kbps						
1 1 1	62.5 kbps						

出荷時設定 00010000

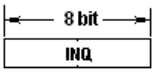
CAN メッセージ仕様

記述に関する注記： 以下、「受信」とは CU-SD1 にとって受信を意味し、ホスト PC から CU-SD1 に送信するメッセージのことを言います。「送信」とはその逆です。

問い合わせメッセージ

<メッセージ長：1 バイト><受信><不揮発領域保持しない>

メッセージ ID： ディップスイッチ設定 ID



INQ：問い合わせコード

00h CU-SD1 が生成するメッセージ数を問い合わせます。

01h CU-SD1 に記憶されている変換条件識別子を問い合わせます。

FFh CU-SD1 の Serial を変換条件設定モードに切り替えます。(PWR LED 緑色点滅となり、送信は行うことができません。)

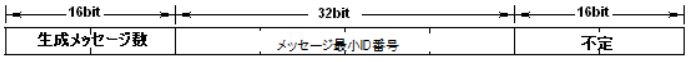
（Serial で CUSD1_CONDITION の終了タグを受信すると変換条件モードを解除します。）

応答メッセージ

<メッセージ長：8 バイト><送信>

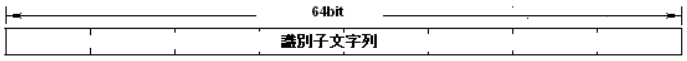
メッセージ ID： ディップスイッチ設定 ID+1

①問い合わせコードが 00h の時



生成メッセージ数 INT16 Little Endian　　メッセージ最小 ID 番号 INT32 Little Endian

②問い合わせコードが 01h の時



識別子文字列 8 文字が戻ります。なお、8 文字に満たない場合は null でパディングします。また、識別子に 8 文字以上記載している場合は先頭から 8 文字が戻ります。変換条件が設定されていない場合、或いは不正な場合は、”-empty-”7 文字が戻ります。

③問い合わせコードが FFh の時、応答メッセージは送信しません。

RS-232 準拠接続機器 条件設定実行メッセージ

<メッセージ長：2 バイト><受信>

メッセージ ID： ディップスイッチ設定 ID+2

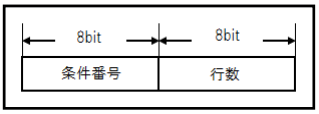
条件設定ファイルの CONDITION_SET 要素で設定した、CU-SD1 に接続した RS-232 準拠機器への条件設定文(RS-232 準拠機器の起動やなんらかの設定のため)を RS-232 準拠機器へ送信します。

条件番号は、Number 属性で指定された番号で、行間待ち時間は複数行のコマンドの行間待ち時間を msec 単位で記述します。

RS-232 準拠接続機器 条件設定応答メッセージ

<メッセージ長：2 バイト><送信>

メッセージ ID： ディップスイッチ設定 ID+3



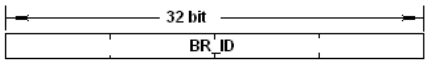
RS-232 準拠接続機器条件設定実行メッセージの応答を返します。条件の最終行を送信したときに、この応答を返します。

条件番号が未定義であった場合、行数に 0 を入れて返答します。

制御 ID メッセージ

<メッセージ長：4 バイト><受信><不揮発領域保持>

メッセージ ID：ディップスイッチ設定 ID+4



BR_ID：ブロードキャストメッセージ ID 番号 出荷時設定 0 （制御メッセージでの動作 OFF）

データ形式： 4 バイト符号無し整数（uint32） バイト並びリトルエンディアン

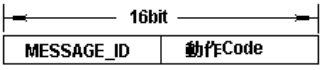
ID 番号 0 は、制御メッセージ ID 番号ではなく、制御メッセージに反応しない（禁止）を意味します。拡張 ID か否かは本体ディップスイッチ設定に従います。本体ディップスイッチ設定が基本 ID(1～2047 有効範囲)の場合で、2047 を越える ID 番号が振られた場合は、下位 12bit のみ有効となります。

※本メッセージは制御メッセージ ID 番号をモジュールに伝える機能を持ちます。

制御メッセージ

<メッセージ長：2 バイト><受信><不揮発領域保持しない>

メッセージ ID：制御メッセージで受信した ID



MESSAGE_ID： 1+7 ビットモジュール ID (1 バイト)

先頭 1 ビットは、特定のモジュールを対象としているか否かを示します。下位 7 ビットは個別ユニットを対象としている場合に設定するユニット ID で、ディップ SW3 の S2～S8 で設定したビットパターンの内容です。

00h～7F h ⇒ 個別モジュールを示します。 80h～FFh ⇒ 個別モジュールを対象としません。

動作 Code： 8 ビット (1 バイト)

00h 0000xxx0 ⇒ 送信停止 01h 0000xxx1 ⇒ 送信開始

ここで定義した動作 Code 以外無視し反応しません。

送信開始/停止はシリアルデータ CAN メッセージ変換条件ファイルによって送信定義されたメッセージに対して機能します。

シリアルデータ変換条件の設定

CAN インタフェースより「問い合わせメッセージ」で変換条件設定モードに切り替えた場合 (CAN ポートで FFh を受信)に有効となります。
その後、シリアル通信により、「シリアルデータ変換条件ファイル」を本機に送信します。
シリアル通信規格は、固定で、ボーレート 19200bps、Stop1、語長 8bit、パリティ無し、通信文は STX/ETX で囲み BCC を付加します。

注意： PC に市販の USB-シリアルインタフェースケーブルを接続して PC から CU-SD1 に通信文を送信する場合は、D-sub 9pin 両端メスのクロス配線のケーブルを使用して下さい。

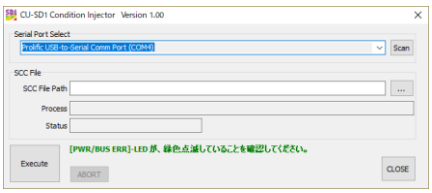
- 変換条件は 1 行毎に送ります。
- CU-SD1 からの応答は、ACK 又は NAK が戻ります。
- 前提として、変換条件は文法 Error が無いことが前提です。
- ここで使用する BCC は、STX の次のキャラクタから ETX(ETX を含む)までの排他的 OR 値（XOR）です。

- XML 宣言文行を受け取ると、現在設定されている内容初期化してから新たな設定条件として格納します。
- CUSD1_CONDITION の終了タグ行を送信すると、応答送信後、Serial は変換条件設定モードを解除します。

なお、シリアルデータ変換条件ファイルの内容については、次ページ以降に記述します。

作成したシリアルデータ変換条件ファイル*.scc 送信プログラム：

上記 CU-SD1 とのプロトコルに対応したシリアル通信プログラム CUSD1Condition.exe をご用意しています。
Windows 7 以降の PC で動作し、市販の USB-シリアル変換アダプタを利用します。
本プログラムについては弊社までお問い合わせください。



CU-SD1 シリアルデータ変換条件ファイル.SCC 仕様編

総則

以降は、CU-SD1 のシリアルデータ変換条件ファイル作成のための説明です。

ファイル形式：

ファイルは XML テキスト形式、拡張子.scc です。本書は CU-SD1 でのシリアルデータ変換条件を定義します。

記述した内容は CU-SD1 の Serial (RS232C コネクタ) を通して CU-SD1 に転送設定します。

記述総則：

XML 形式に準拠した書式で記述します。要素名はすべて大文字、属性名は小文字で記述することを基本とします。XML 言語では大文字小文字を厳密に区別するので注意が必要です。

また、XML 形式のコメントであれば、コメント文も記述可能とします。

なお、本体処理系の制限から、一行の長さを 256 文字に制限し、それ以上長い行を含むファイルはエラーとなります。

また、各設定項目の文字列にも最大長の制限があります。

なお、各項にて【省略可】とある項は省略が可能です。

テキストの中ではエスケープシーケンス（¥r¥n 等）による特殊記号の記述が可能です。

以下、特に断りなき場合、数値は 10 進数表記です。

ファイル構成要素

XML 宣言文

記述例	**<?xml version="1.0" encoding="Shift_JIS"?>**
機能	本ファイルが XML 言語仕様であることを示します。
備考	本項は上記内容にて固定です。本ファイルに多バイト(全角)文字は原則使用禁止ですが、コメントなどで使用するときは Shift_JIS とします

根要素

要素名	**CUSD1_CONDITION**	
機能	本ファイルが CU-SD1 の設定ファイルであることを示します	
属性	Date	【省略可】【文字列】 このファイルの管理のための日付を記述します。
	Time	【省略可】【文字列】 このファイルの管理のための時刻を記述します。SD1 本体には読み込まれません。
	Name	【文字列】 このファイルの識別子を 8 文字以内で記述します。
記述例	<CUSD1_CONDITION Date="10/15/2012" Time="12:10:02" Name="cnd-001"> : </CUSD1_CONDITION>	
備考	Date および Time の値は SD1 本体には読み込まれません。 Name の内容は SD1 に読み込まれ、CU-SD1 の CAN メッセージにて読み出すことが可能です。 本設定ファイルのこれ以下の部分は、この要素 CUSD1_CONDITION の内容として記述します。したがって、本ファイルはこのタグによる終了タグを以て終わりとなります。本要素は根要素ですので、１ファイルに１つのみ存在します。	

シリアル条件指定要素（CUSD1_CONDITION 要素の内容として記述）

要素名	**SERIAL**	
機能	RS-232 準拠シリアルポートの諸元を指定します。	
属性	Rate	【文字列】 転送レートを以下の数値の中から一つ指定します。 1200 , 2400 , 4800 , 9600 , 19200 , 38400 , 57600, 76800 , 115200
	Stop	【文字列】 ストップビットを以下の数値の中から一つ指定します 1 , 2
	Parity	【文字列】 パリティを以下から選択し、記述します。 odd , even , none
	Length	【省略可】【文字列】 語長を以下の数値の中から一つ選択します。 7 , 8 なお、本属性は省略可であり、省略時は 8 と見なします。
記述例	<SERIAL Rate="38400" Stop="1" Parity="none" Length="8"/>	

備考	本要素は閉じ括弧の直前にスラッシュを置くなどして、空要素タグとして記述します。 本要素が複数現れた場合、最後に現れた設定が有効となります。

文字列通信文定義ブロック開始要素（CUSD1_CONDITION 要素の内容として記述）

要素名	**CHR_STREAM**	
機能	シリアルポートで受信し CAN メッセージ変換出力する項目を含む文字列通信文の構成を定義します。	
属性	Delimiter	【文字】 項目区切り文字を、一文字で記述します。 また、文字で記述できないコードの場合、以下のエスケープ記号が記述可能です。 意味 / 記号 無し / ¥0 TAB / ¥t
	Terminator	【文字列】 行終端コードを以下のエスケープ記号で記述します。 意味 / 記号 ETX / ¥x03 （CU-SD1 独自記号） CR/LF / ¥r¥n CR / ¥r LF / ¥n
	Length	【文字列】 Char 属性に記述する先頭識別子の長さを指定します。最大 32。省略可能です。
	Char	【文字列】 通信文の先頭に付けられている識別子とする文字列を記述します。なお、文字列の先頭および末尾にスペースを記述しても無視します。Length 属性を省略した場合、Length として、この実長を採用します。
内容	後述する「MESSAGE 要素」を記述します。	
記述例	<CHR_STREAM Delimiter="," Terminator="¥r¥n" Char=”$GPGGA,” 　<MESSAGE ・・・・・・・・ </CHR_STREAM>	
備考	Char 属性に記述した文字列の文字数より Length 属性の指定が多い場合は、末尾にスペースを付加したものを採用します。逆に Length 属性の指定が少ない場合は末尾を削り採用します。 ETX を表すエスケープ記号は CU-SD1 独自のもので¥x 記号の次の二桁は 16 進数で表した ASCII コードを意味します。 本要素は、BIN_STREAM 要素数と合わせて 4 つまで記述可能です。	

バイナリ通信文定義ブロック開始要素（CUSD1_CONDITION 要素の内容）

要素名	**BIN_STREAM**	
機能	シリアルポートで受信し CAN メッセージ変換出力する項目を含むバイナリ通信文の構成を定義します。	
属性	Length	【文字列】 開始識別ビットパターンを含めた通信文の全長をバイト単位で記述します。
	Bin	【文字列】 バイナリ通信文の「開始識別ビットパターン」を意味し、HEX 文字列で記述します。最大 32 バイト分。
内容	後述する「MESSAGE 要素」を記述します。	
記述例	<BIN_STREAM Length="100" Bin="AA4411"> 　<MESSAGE ・・・・・・・・ </BIN_STREAM>	
備考	Bin 属性に記述する HEX 文字列の文字数は 2 の倍数であることが必要です。 本要素は、CHR_STREAM 要素数と合わせて 4 つまで記述可能です。	

CAN メッセージ定義ブロック開始要素（CHR_STREAM または BIN_STREAM 要素の内容）

要素名	**MESSAGE**	
機能	CAN ポートに送信するメッセージを定義します。	
属性	Name	【文字列】 CAN メッセージ名を文字列で記述します。記述したメッセージ名は本体内部では参照しません。
	RelativeId	【文字列】 相対 ID 番号を記述します。開始 ID 番号（本体ディップスイッチ設定 ID+5）との差分値です。
	Length	【文字列】 構成するメッセージの長さをバイト数で設定します。バイト数は 1～8 の範囲でなければいけません。
内容	後述する「SIGNAL 要素または SIGNAL_B 要素」を記述します。	
記述例	<MESSAGE Name="abc" RelativeId="0" Length="8"> 　<SIGNAL ・・・・・・・・・・ </MESSAGE>	
備考	本体ディップスイッチで設定した ID 番号から 5 個(ID 番号+0 ～ +4)は CU-SD1 固有の CAN メッセージとして使用します。したがって、RelativeId =”0”と記述した場合は、本体ディップスイッチ設定 ID 番号+5 が当該メッセージの ID 番号となります。同一の RelativeId を保つ複数の MESSAGE 要素で、異なる Length 属性を指定してはいけません。もし異なった Length を指定した場合は最大値を採用します。 本要素は 1 設定（1 つの.SCC ファイル）につき、6 個（6ID）まで指定可能です。	

文字列通信文 CAN メッセージ信号定義要素（MESSAGE 要素の内容）

要素名	**SIGNAL**	
機能	文字列通信文定義ブロック（CHR_STREAM）内に記載するメッセージ定義を構成する信号を定義します。	
属性	ItemNum	【数字文字列】 項目番号を記述します。着信した通信文を着信順に CHR_STREAM 要素 の Delimiter 属性によって区切った時の項目番号で、行識別子は含みません。項目番号は 1 から振ります。 項目は、Type 指定が Bit 並びと文字列の場合を除いて、当該文字列数値を有効数値桁落ちの無いように適宜変換して、いったん内部形式にて保持した後、Type 属性による変換を経て CAN 出力します。
	Position	【数字文字列】 構成する CAN メッセージ内の開始 bit 位置と bit 長をカンマで区切って記述します。
	Type	【文字列】、Endian は【省略可】【文字列】 CAN に出力する時の変換形式と Endian をカンマで区切って記述します。 ・変換形式は下記キーワード表から選択して記述します。

型 種別	キーワード	備考
2byte 整数	int16	範囲を越えた場合、最大値(32767)また最小値(-32768)とします。
2byte 符号なし整数	uint16	範囲を越えた場合、最大値とします。
4byte 整数	int32	範囲を越えた場合、最大値または最小値とします。
4byte 符号なし整数	uint32	範囲を越えた場合、最大値とします。
4byte 浮動小数点	float32	
8byte 浮動小数点	float64	
bit 並び	bit	16bit を越える場合でもバイト並びは無視します。
文字列	char	bit 長は 8bit の倍数必須、バイト並び無視します。

	指定した数値変換型式に変換できない場合、例えば数値表現文字以外を含む場合などは指定された数値型式が取り得る最大値に変換します。なお、bit 並びとは ASCII キャラクタに於ける 1 と 0 の文字列並びであり、シリアル受信するデータをそのままバイナリとして受け取るものではありません。シリアル受信したデータをそのままバイナリとして処理したい場合は BIN_STREAM 要素の中で SIGNAL_B を使います。 ・Endian は、“little”または“big”を記述します。タイプが char の場合は省略可能です。('0''1'の bit 文字ストリームは内部で二進数に変換した後 Endian 変換します。）
Coefficient	【省略可】【文字列】 換算係数の情報を、ビット重み、オフセットをカンマで区切って記述します。または変換 Table 名を記述します。変換 Table 名を記述した場合は、必ず TABLE 定義文で内容を指定する必要があります。ビット重み&オフセット設定と Table 設定を同時に指定することはできません。 本 Coefficient 属性が省略された場合、ビット重み 1 、オフセット 0 となります。
Unit	【省略可】【文字列】 信号の単位を文字列で記述します。記述した単位は本体内部では参照しません。
内容	【省略可】【文字列】 信号名を文字列で記述します。記述した信号名は本体内部では参照しません。省略し空要素でも可です。
記述例	<SIGNAL ItemNum="3" Position="0,32" Type="float32 , little" Coefficient="1 , 0" Unit="">　Signal1　</SIGNAL>
備考	バイナリ通信文定義ブロック(BIN_STREAM)内のメッセージ定義では本要素を記述することはできません。BIN_STREAM 内では SIGNAL_B 要素を記述してください。 同一の MESSAGE の中で bit 位置が重複するような設定が行われた場合、動作は不定となります。 本要素は SIGNAL_B 要素数と合わせて 1 設定 (1 つの.SCC ファイル) につき、20 個まで設定可能です。 変換 Table 名を記述した場合は、必ず TABLE 定義文で内容を指定する必要があります。ビット重み&オフセット設定と Table 設定を同時に指定することはできません。

バイナリ通信文 CAN メッセージ信号定義要素（MESSAGE 要素の内容）

要素名	**SIGNAL_B**	
機能	バイナリ通信文定義ブロック(BIN_STREAM)内に記載されるメッセージ定義を構成する信号を定義します。	
属性	Location	【文字列】 シリアルバイナリ通信文の変換開始バイト位置とバイト数を半角カンマで区切って記述します。変換開始バイト位置はバイナリ通信文開始定義ブロックで設定した「開始識別ビットパターン」の先頭からのバイト位置を意味します。バイト位置は先頭 1 から振られます。但し、先頭から「開始識別ビットパターン」バイト数+1 から使用でき、「開始識別ビットパターン」をメッセージ変換することはできません。なお、SrcType に記述される型式が数値型式の場合、バイト数は記述省略可能で、また記述があっても SrcType の変換形式指定によるバイト数が優先されます。
	Position	【文字列】 構成する CAN メッセージ内の開始 bit 位置と bit 長をカンマで区切って記述します。DstType に記述される型式が数値型式の場合、bit 長は記述省略可能で、また記述があっても DstType の変換形式指定による bit 長が優先されます。

	SrcType	【文字列】、Endian は【省略可】【文字列】 シリアルで入力された値の変換形式と Endian をカンマで区切って記述します。 ・変換形式は下記キーワード表から選択して記述します。

型 種別	キーワード	備考
2byte 整数	int16	範囲を越えた場合、最大値(32767)または最小値(-32768)とします。
2byte 符号なし整数	uint16	範囲を越えた場合、最大値とします。
4byte 整数	int32	範囲を越えた場合、最大値または最小値とします。
4byte 符号なし整数	uint32	範囲を越えた場合、最大値とします。
4byte 浮動小数点	float32	
8byte 浮動小数点	float64	
bit 並び	bit	16bit を越える場合でもバイト並びは無視します。
文字列	char	bit 長は 8bit の倍数必須、バイト並びは無視します。

		指定した数値変換型式に変換できない場合、例えば数値表現文字以外を含む場合などは指定した数値型式を取り得る最大値に変換します。なお、bit 並びとは、シリアル受信するデータをそのままバイナリとして処理します。そのため Position の bit 長が 8 の倍数であった場合には、実質的に「文字列」設定と同じ扱いになります。 ・Endian は、“little”または”big”を記述します。タイプが bit または char の場合は省略可能です。
	DstType	【文字列】、Endian は【省略可】【文字列】 CAN で出力する値の変換形式と Endian をカンマで区切って記述します。 ・変換形式は上記 SrcType と同じです。 ・Endian も上記 SrcType と同じです。
	Coefficient	【省略可】【文字列】 換算係数の情報を、ビット重み、オフセットをカンマで区切って記述します。または変換 Table 名を記述します。変換 Table 名を記述した場合は、必ず TABLE 定義文で内容を指定する必要があります。ビット重み&オフセット設定と Table 設定を同時に指定することはできません。 本 Coefficient 属性が省略された場合、ビット重み 1 、オフセット 0 となります。
	Unit	【省略可】【文字列】 信号の単位を文字列で記述します。記述した単位は本体内部では参照しません。
内容	【省略可】【文字列】 信号名を文字列で記述します。記述した信号名は本体内部では参照しません。省略して空要素でも可です。	
記述例	<SIGNAL_B Location=”10,2” Position=”0,32” SrcType=”int16,little” DstType=”int32,big” Coefficient="1 , 0" Unit="mV">Sig n al_B </SIGNAL_B>	
備考	文字列通信文定義ブロック（CHR_STREAM）内のメッセージ定義では記述することはできません。CHR_STREAM 内では SIGNAL 要素を記述してください。 同一の MESSAGE の中で bit 位置が重複するような設定を行った場合、動作は不定となります。 本要素は SIGNAL 要素数と合わせて 1 設定（1 つの.SCC ファイル）につき、20 個まで設定可能です。 変換 Table 名を記述した場合は、必ず TABLE 定義文で内容を指定する必要があります。ビット重み&オフセット設定と Table 設定を同時に指定することはできません。	

変換テーブル定義要素（CUSD1_CONDITION 要素の内容）

要素名	**TABLE**	
機能	受信した文字列を変換する場合に使用する変換テーブルを定義します。	
属性	Name	【文字列】 信号定義要素（SIGNAL または SIGNAL_B）の Coefficient 属性で記述したテーブル名を記述します。TABLE 名は 8 文字以内の英数字で構成し、最初の文字は数字以外として下さい。
	Undefined	【文字列または数値】 未定義の変換前値に遭遇した場合の変換後の値を定義します。省略はできません。8 文字以内で表記します。
内容	変換前の値と,変換後の値をカンマで区切って複数行(最大 4 行)列挙記述します。 ・一組の変換定義は CR/CF で区切られた別の行である必要があります。 ・変換前値は、変換する値を記述します。文字列扱いの場合は半角ダブルコーテーションで囲んで記述します。数値の場合、10 進数の場合はそのまま、2 進数表現は半角"b、16 進数表現は半角"h""をプレフィックス付加して記述します。 ・変換後値は、変換される値を記述します。文字列の場合は半角ダブルコーテーションで囲んで記述します。 ・テーブルの最大サイズは総文字数+行数+ 2 が 64 以下になるように指定してください。	
記述例	<TABLE name=”tbl1” Undefined=”-1”> “N” , 0 “S” , 1 </TABLE>	
備考	テーブル変換する値に一致する変換前値が定義されていない場合、Undefined 属性に指定された値が変換後値として採用されます。 TABLE 要素は最大 8 個まで定義できます。 結果の形式は SIGNAL_B の DstType または SIGNAL の Type に従って格納されます。その際、例えば変換先が int 形式で、Undefined に数値以外が設定されているなど、設定が矛盾する場合はゼロになります。 各テーブル要素の右辺左辺は統一している必要があります。例えば以下の様な設定はできません。 <TABLE Name="tbl1" Undefined="-1"> "N" , 0 15 , 1 </TABLE> (1 番目と 2 番目の第 1 項目（左辺）の形式が異なっています。) 数値は整数のみ(int32 範囲)が記述可能です。bin や hex であっても int32 の範囲である必要があります。	

データ要求定義要素（CUSD1_CONDITION 要素の内容）

要素名	**DATA_REQUEST**	
機能	CU-SD1 に接続した RS-232 準拠機器からデータを受け取るためにデータ送信要求が必要な場合に記述します。	
属性	Times	【文字列】 “Pon”、“Respond”または、”Both”を記述します。Pon 指定では、受信開始する最初だけ送信し、Respond 指定では、シリアル変換対象文字列を送信、Both ではその双方で送信します。 Respond は、CU-SD1 に接続した RS-232 準拠機器に対する Acknowledgement として使用できます。CHR_STREAM または BIN_STREAM で指定した変換対象文字列を受信して変換完了後、ここに記述した文字列を CU-SD1 の RS-232 準拠ラインから送信します。
	Type	【数字】 通信文の送信タイプを示すコードを記述します。コードは以下の通りです。

コード	意味
0	コマンドニモニック文字列をそのまま送出
1	コマンドニモニック文字列を STX/ETX で囲んで送信
2	コマンドニモニック文字列終端に C/R を付加して送信
3	コマンドニモニック文字列終端に C/R L/F を付加して送信
4	コマンドニモニック文字列を STX/ETX で囲み、BCC を付加して送信

内容	【文字列】 データ要求コマンドの文字列（コマンドニモニック）を記述します。文字列前の空白文字は無視します。文字列は最大 16 文字です。
記述例	<DATA_REQUEST Times=”Both” Type=”2”> “STR A” </ DATA_REQUEST>
備考	本文が複数行存在した場合は最終に記述した内容を参照します。 内容文字列では、¥r, ¥n, ¥xXX の 3 種類のエスケープが可能で、16 進数を記述可能です。ただし、¥x00（Null）の使用は禁止です。

データ停止定義要素（CUSD1_CONDITION 要素の内容）

要素名	**DATA_STOP**	
機能	CU-SD1 に接続した RS-232 準拠機器からのデータ送信を停止するためにコマンドが必要な場合に記述します。	
属性	Type	【数字】 通信文の送信タイプを示すコードを記述します。コードは以下の通りです。

コード	意味
0	コマンドニモニック文字列をそのまま送出
1	コマンドニモニック文字列を STX/ETX で囲んで送信
2	コマンドニモニック文字列終端に C/R を付加して送信
3	コマンドニモニック文字列終端に C/R L/F を付加して送信
4	コマンドニモニック文字列を STX/ETX で囲み、BCC を付加して送信

内容	【文字列】 データ停止要求コマンドの文字列（コマンドニモニック）を記述します。文字列前の空白文字を無視します。文字列は最大 16 文字です。
記述例	<DATA_STOP Type=”2”> “STP A” </DATA_STOP>
備考	本文は本体ディップスイッチ自走 ON/OFF 設定に感応します。自走 ON の時は無視し送信しません。 OFF の時は CU シリーズブロードキャストメッセージの収録停止指示を受信した時に送信します。 本文が複数行存在した場合は最終に記述した内容を参照します。 内容文字列では、¥r, ¥n, ¥xXX の 3 種類のエスケープが可能で、16 進数を記述可能です。ただし、¥x00（Null）の使用は禁止です。

RS-232 準拠接続機器条件設定要素（CUSD1_CONDITION 要素の内容）

要素名	**CONDITION_SET**	
機能	CU-SD1 に接続した RS-232 準拠機器のデータ送信開始などの送信内容と条件の設定を定義します。CU-SD1 の電源 On 時の自動送信に加えて、本ブロックを、CAN メッセージで条件設定実行文を受信した時に参照し RS-232 準拠機器に送信するような設定も可能です。	
属性	Number	【数字】 0～3 の範囲で数値を記述します。ここで記述した番号は CU-SD1 固有の条件設定実行メッセージの内容と一致した時に RS-232 準拠接続機器に一括送信します。 なお、0 を指定すると、CU-SD1 の電源 On 時に CONDITION_SET の内容として記述した文字列を自動的に送信します。
	Name	【文字列】 設定する機器名などを記述します。この項目は本体内では参照しません。
	Type	【数字】 通信文の送信タイプを示すコードを記述します。コードは以下の通りです。

コード	意味
0	コマンドニモニック文字列をそのまま送出
1	コマンドニモニック文字列を STX/ETX で囲んで送信
2	コマンドニモニック文字列終端に C/R を付加して送信
3	コマンドニモニック文字列終端に C/R L/F を付加して送信
4	コマンドニモニック文字列を STX/ETX で囲み、BCC を付加して送信

	Wait	【数字】 上記の Number 属性を 0 とした時のみ有効となります。CONDITION_SET の内容として記述した文字列行間の Wait 時間を msec 単位で記述できます。省略した場合は、100 msec となります。

内容	【文字列】 条件設定コマンドの文字列（コマンドニモニック）を複数行列挙記述します。複数行にわたる場合は CR/CF で区切った別の行である必要があります。文字列前の空白文字を無視します。 ・最大サイズは、総文字数 + 行数 + 2 が 64 以下になるように指定してください
記述例	<CONDITION_SET Number="0" Name="DeviceName" Type="2" > SLT19 18 17 SLT16 15 14 MODE "A" </CONDITION_SET>
備考	条件設定ブロックは、電源起動時（Number 属性が 0）、または CU-SD1 固有の条件設定実行 CAN メッセージを受信した場合に参照します。この要素は最大 4 個記述可能で、それぞれの要素においては Number 属性がすべて異なる必要があります。 内容文字列では、¥r, ¥n, ¥xXX の 3 種類のエスケープが可能で、16 進数を記述可能です。ただし、¥x00（Null）の使用は禁止です。

作成例

作成例 1： 受信した GPS NMEA フォーマットから時刻、緯度、経度、高度および速度を CAN メッセージに変換する場合

受信する通信文例 ⇒ $GPGGA,014534,3501.568,N,13546.945,E,2,09,1.0,80.7,M,34.1,M,0,0000*46
$GPVTG,351.8,T,358.3,M,000.0,N,000.1,K*4D

GGA 文から変換する対象項目は項目番号.1：時刻、項目番号 2：緯度、項目番号 4：経度、番号 7：平均海水面高度となります。
VTG 文から変換する対象項目は項目番号 7：対地速度(km/h)となります。
なお、下記記述例では北緯南緯種別および東経西経種別は変換していません。

```
<?xml version="1.0" encoding="Shift_JIS"?>
<CUSD1_CONDITION Date="10/15/2012" Time="12:10:02" Name="cnd-001">
  <SERIAL Rate="38400" Stop="1" Parity="none" Length="8"/>
  <CHR_STREAM Delimiter="," Terminator="¥r¥n" Length="7" Char="$GPGGA,">
    <MESSAGE RelativeId="0" Length="8">
      <SIGNAL ItemNum="1" Position="0,32" Type="uint32,little">
          Time(UTC)
      </SIGNAL>
      <SIGNAL ItemNum="2" Position="32,32" Type="float32,little">
          latitude
      </SIGNAL>
    </MESSAGE>
    <MESSAGE RelativeId="1" Length="8">
      <SIGNAL ItemNum="4" Position="0,32" Type="float32,little">
          longitude
      </SIGNAL>
      <SIGNAL ItemNum="7" Position="32,32" Type="float32,little">
          longitude
      </SIGNAL>
    </MESSAGE>
  </CHR_STREAM>
  <CHR_STREAM Delimiter="," Terminator="¥r¥n" Length="7" Char="$GPVTG,">
    <MESSAGE RelativeId="2" Length="4">
      <SIGNAL ItemNum="7" Position="0,32" Type="float32,little" Unit="Km/h">
          velocity
      </SIGNAL>
    </MESSAGE>
  </CHR_STREAM>
</CUSD1_CONDITION>
```

上記の例で送信する CAN メッセージは 3 個となります。ただし、NMEA フォーマットでは時刻は hhmmss.ss ですので、CAN メッセージ受信後、変換しないと正確な時分秒を表しません。また、同じく経度緯度も同様に度分で表していますので、分の単位で表すには変換が必要です。

改訂履歴

2023/9/4	Rev. 2.07	DATA_REQUEST 備考の Demand 関連記述削除
2021/7/12	Rev. 2.06	レイアウト修正
2020/5/12	Rev. 2.05	改訂履歴構成変更
2020/5/11	Rev. 2.04	キーワード修正（大文字/小文字）
2018/1/2	Rev. 2.03	DIP-SW4 SW13 の記述追加 シリアル通信プログラム CUSD1Condition.exe の記述追加 シリアル条件指定要素に 57600 (bps) を追加 その他記述補足
2013/6/3	Rev. 2.02	DATA_REQUEST 要素 Times 属性の Demand を 廃止し Respond に変更 CONDITION_SET 要素に Wait 属性を追加 DATA_REQUEST, DATA_STOP, CONDITION_SET の 記述内容文字列にエスケープの使用可能を追記
2013/2/20	Rev. 2.01	外形図を差し替え各部の名称を追記 BCC の内容追記 語句の修正
2013/2/14	Rev. 2.00	仕様書とシリアルデータ変換条件ファイル補足資料を まとめ更新
2013/2/12	Rev.1.03	生成メッセージ ID 数を 6 に、シグナル数 20 に拡張 変換テーブル定義要素を追加
2013/1/7	Rev.1.02	生成メッセージ ID 数/シグナル数の記述に追記
2012/12/24	Rev.1.01	変換条件ファイルフォーマット 1.12 相当 キーワードの大文字小文字の不統一を修正 (属性文字列 name→Name) STOP 1.5bit の廃止 ¥xnn（C 言語風 16 進記述方法）不可 変換テーブル関連項目削除
2012/10/26	Rev.1.00	初版（変換条件ファイルフォーマット 1.10 相当）