

PASTA

1.0

ユーザーズ マニュアル



ご使用の前に

PASTA1.0(以下本製品)のご使用前には、必ず本書「PASTA1.0 ユーザーズマニュアル」を良くお読みいただき、内容を理解したうえで正しくお使いください。

- (1) 本書の内容につきましては、予告無く変更することがあります。
- (2) 本製品はトヨタ自動車株式会社が著作権を持ち、製造元は株式会社エル・アンド・エフ、販売元は株式会社チップワンストップとなります。
- (4) 本製品を使用して得られた実験データにつきましては、正確性や再現性等を保証するものではありませんので、お客様の責任においてご活用ください。
- (4) 本書についてご不明点やお気づきの点、本製品のサポート内容などにつきましては、販売元へお問い合わせください。

使用上の注意

本製品は研究・実験用途を想定しておりますので、お客様による分解・改造をされた場合でも製品サポートは継続させていただきますが、それが原因で本製品が故障した場合には有償修理となる場合がございますので予めご承知おきください。

付属の AC アダプタ内部には高電圧部があり危険ですので、如何なる場合にも分解・改造はしないでください。

お客様が用意された電源で使用中に本体が故障した場合は、有償修理となる場合がありますのでご注意ください。

発煙・異臭・異音など何か異常を感じた場合や、異物が内部に入ってしまった場合などは、そのまま使用せず、すぐに電源コードを抜き販売元にご相談ください。

以下の行為は、故障・怪我・感電・火傷・火災などの事故原因になりますのでおやめください。

●本製品の取り扱い、使用場所等について

- 1) 使用時は、トランクの蓋を完全に開け、指詰め防止金具がロックした事を確認する。
- 2) 不安定な場所（ぐらついた台の上や傾いた所など）に置かない。
- 3) 湿気やほこりの多い場所、ガスの充満した場所で使用しない。
- 4) 本製品の上に乗ったり、重いものを置いたり、倒したり落としたりしない。
- 5) 本製品の内部に異物を入れたりしない。
- 6) 周囲で可燃性ガスのスプレーを使用しない。
- 7) 使用後は、指詰め防止金具のロックを外し、トランクの蓋をゆっくり閉める。
- 8) 本製品を移動する場合は、電源など全てのケーブルを外す。
- 9) 本製品が変形する場合があるので、トランクの蓋を開いたまま持ち上げない、移動しない。

●本製品のケーブル等の扱いについて

- 1) 電源プラグはコンセントに根元まで差し込み、長期間差したままにしない。
- 2) ケーブルをトランクに挟まないようにする。
- 3) ケーブルの上に重量物や鋭利なものを乗せない。
- 4) ケーブルを無理に曲げたり、ねじったり、引っ張ったりしない。
- 5) 傷ついたケーブルや変形したコネクタは使用しない。
- 6) ほこりや液体などの異物が付着したまま使用しない。

使用制限

本製品は研究・実験・学習用途を想定しておりますので、航空宇宙機器、幹線通信機器、原子力制御機器、医療機器など、きわめて高い信頼性・安全性が必要とされる用途、又は自動車・列車・船舶・航空機などの運行に直接関わる装置・防災防犯装置・各種安全装置など機能・精度などにおいて高い信頼性・安全性が必要とされる用途には使用しないでください。

ドライバ、ユーティリティ、 マニュアル類のダウンロード

本装置で用いられている ECU の回路図や、デフォルトでインストールされているファームウェアのソースコードはオープンになっています。回路図、プロジェクトファイル、ユーティリティ類、マニュアル類の最新版は、下記の URL からダウンロードすることができます。

下記サイトに掲載された内容は、予告なしに更新される場合があります。

<https://github.com/pasta-auto>

本書内での PASTA の定義

本書で「PASTA」と示している場合、当製品を指します。特注仕様や互換機も PASTA と呼ばれることがあります。この説明書に書かれている仕様とは異なる場合がありますので、ご注意ください。

本書内での Windows の定義

本書で「Windows」と表記する場合は、Windows10 の Microsoft がサービス提供をしているバージョンを指します。

Microsoft のサポート対象外のバージョンにつきましては、本装置でもサポート対象外となりますのでご注意ください。

目次

1. 開梱図と本体と付属品.....	6
PASTA 上部	8
PASTA 下部	9
2. 基本操作の説明	11
電源ケーブルの接続と主電源オンオフ	11
エンジンスタートとエンジンストップ	12
3. しくみの解説.....	13
4. ECU の説明	15
5. 付属品の使用方法	17
肩掛けベルト	17
簡易鍵.....	18
AC アダプタと電源ケーブル.....	19
OBD II -Dsub 変換ケーブル	20
USB-CAN 変換ボード.....	22
付属の CAN ケーブルと J/C コネクタを使用して、J/C の渡りケーブルを作ってみる	27
クリッピング用分岐コネクタで信号を分岐させて CAN データをモニタしてみる.....	31
付属ダウンロードケーブルを使って、ファームウェアを書き換えてみる.....	33
6. 付録	45
PASTA 基本仕様	45
ECU 基本仕様.....	46
AC アダプタ仕様.....	47
LFA6U 仕様	48
CAN ID リスト.....	49
CARLA 公開版.....	50
7. お問い合わせ先.....	51
販売元.....	51
製造元.....	51
開発元.....	51

1. 開梱図と本体と付属品

本章では PASTA1.0（以降 PASTA と呼びます）の本体および同梱の付属品についてご説明します。万一足りないものがありましたら販売元までご連絡ください。単品形式があるものについては販売元にお問い合わせいただきましたら単品で購入することが可能です。

図 1.1 PASTA 外観



表 1.1 同梱物品

名称	個数	単品形式
PASTAトランク本体	1	-
PASTAトランク用肩掛けベルト	1	-
PASTAトランク用鍵	1	-
ACアダプタ	1	PASTA-ACDC
ACアダプタケーブル	1	PASTA-ACCJ <日本:JP, 北米:US, 欧州:EU, 英国:UK 豪州:AU 中国:CN>
OBDII-Dsub変換ケーブル	1	PASTA-ODS9
USB-CAN変換ボード + J/Cケーブル	1	PASTA-LFA6U
J/Cコネクタ (単品)	20	PASTA-JCCN
J/C 4分岐コネクタ	1	PASTA-JCJC4
CANケーブル 5m (先バラ)	1	PASTA-CANC5
クリッピングポイント用ケーブル	5	PASTA-CLC
クリッピング用分岐コネクタ (単品)	20	PASTA-CLCN
ECUダウンロードケーブル	1	PASTA-DLC1
ECUモニタケーブル	1	PASTA-MNC1
USBシリアル変換ケーブル	2	REX-USB60F
ECU用ネームラベル	4	PASTA-NAME
付属CD	1	PASTA-CD
PASTA開発環境	1	
ECU ファームウェアプロジェクト		
ユーザーズマニュアル(本書)	1	-

※上記付属品は、標準仕様のものを示しています。

※本体・付属品の仕様、付属品の種類・数量は予告なく変更される場合があります。

PASTA 上部

PASTA のアタッシュケースを開くと、アタッシュケースの上部にはシミュレートされた車両の状態や振る舞いを示すパネルと、車両をコントロールするためのコントローラが配置されています。まず上部について説明します。

図のように、アタッシュケース上部には3つのモニタがあります。それぞれのモニタにおいて、PASTA でシミュレートされている車両（以降仮想車両と呼びます）の状態を表示します。仮想車両の状態は、アタッシュケース下部の ECU が CAN を用いて共有した情報からシミュレートしています。図のように、モニタの下には仮想車両をコントロールするためのコントローラ群が配置されています。これらのコントローラは車両をシミュレートしている3つの CPU ボードのいずれかに接続されていて、仮想車両をコントロールすることができます。車両の状態はアタッシュケース下部にある ECU に送信され、ECU たちは CAN メッセージを通して車両の状態を共有します。

図 1.2 PASTA 上部のアクチュエータ部



図 1.3 PASTA 上部のコントローラ部



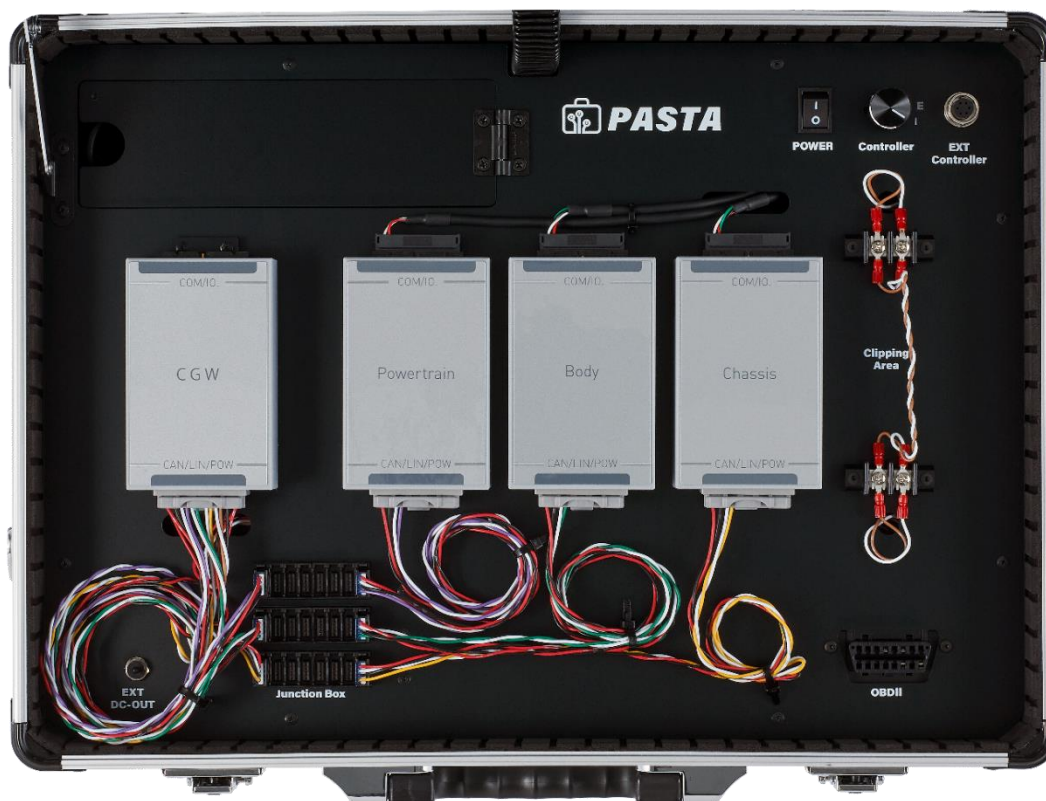
PASTA 下部

アタッシュケースの下部について説明します。下部には主に実車両内のネットワークをエミュレートするようデザインされ、4 個の仕様がオープンになっている PASTA 独自の ECU である Whitebox ECU（以降、簡単のため ECU と呼びます）で構成されています。一番左の ECU は、出荷時は CGW（Central GateWay）とラベルがついています。CGW とは、車両を診断するための OBDII ポートからの通信を監視したり、その他の ECU 同士が CAN プロトコルを用いた通信を中継したりフィルタリングしたりする役割をします。CGW ECU の隣は Powertrain ECU、その隣は Body ECU、一番右は Chassis ECU とラベルがついています。これら 3 つの ECU はそれぞれが CGW ECU に CAN バスで接続されていて、CGW を介して CAN メッセージを送受信することで他の ECU と車両に関する様々な情報を共有します。

PASTA 下部には車両内に実装されたセキュリティ技術を容易に評価できるよう、代表的な攻撃の窓口となるアタックサーフェスが準備されています。最も代表的なアタックサーフェスである

OBDII ポート、車両内部に存在する ECU のハブとなる Junction Box、物理的に CAN のワイヤを分岐することで CAN メッセージを注入できる Clipping Area があります。その他には、主電源、PASTA でシミュレートされている車両をコントロールするための外部コントローラを接続するポート、内部のコントローラと外部のコントローラのどちらを利用するか決めるためのセレクタが配置されています。

図 1.4 PASTA 下部



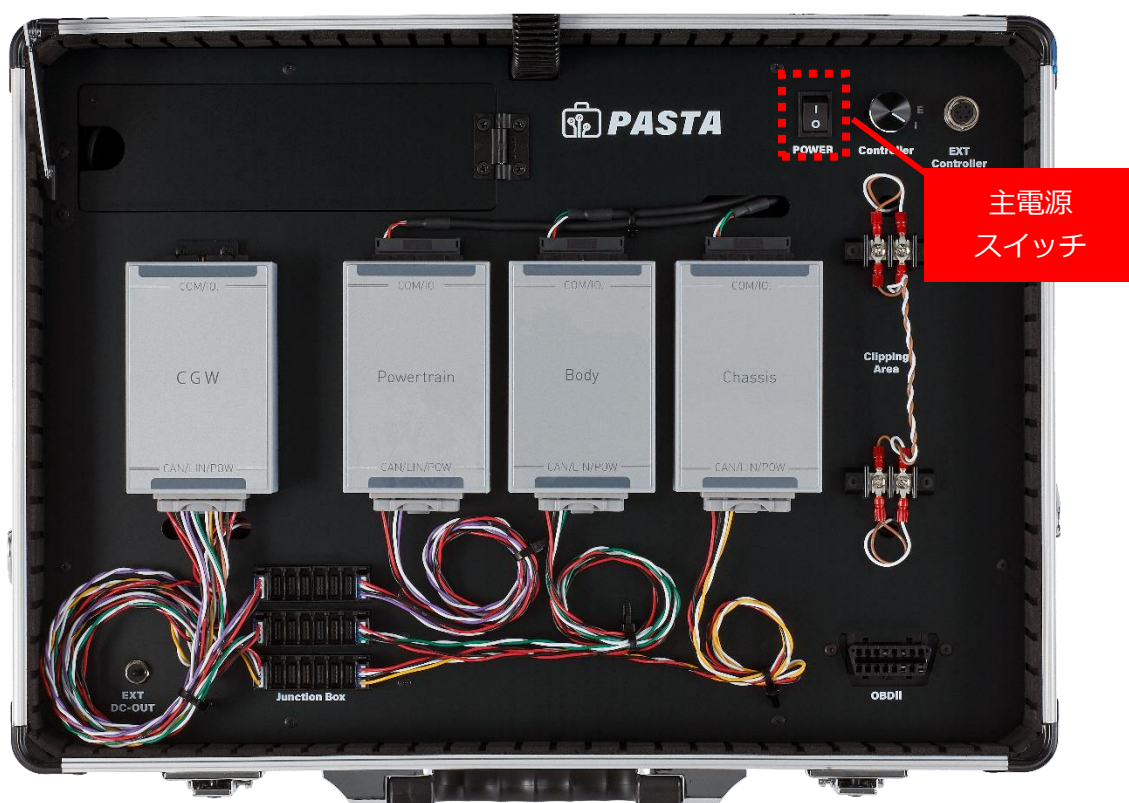
2. 基本操作の説明

電源ケーブルの接続と主電源オンオフ

PASTA の電源を入れる前に、まずは電源ケーブルをコンセントに接続します。電源ケーブルを接続した後、図の四角に囲まれた主電源スイッチを入れて電源をオンにします。これにより PASTA 全体に給電され、様々な操作が可能になります。なお、ここで説明する操作手順は PASTA 特設サイトに掲載中のデモ動画でも観ることができますので合わせてご覧ください。内容は、予告なしに更新される場合があります。

<https://www.chip1stop.com/sp/products/toyota-pasta>

図 2.1 主電源の位置



エンジンスタートとエンジンストップ

ここでは基本的な操作の例として、仮想車両を運転してみましょう。主電源を入れた段階では、まだ PASTA 内の車両のエンジンはスタートしていません。エンジンをスタートするためには、図 2.2 の(1)に示す Ignition ボタンを押します。Ignition ボタンを押すとボタンが青く点灯します。ボタンを押すと PASTA 内でシミュレートされている自動車のエンジンがかかった状態となり、[図 1.2](#) に示した PASTA 上部の右側のモニタが点灯し、自動車の車速やシフトポジションの状態を示します。

まず図 2.2 の(2)にある Shift Position を操作し、アクチュエータ部に表示されているシフトポジションを走行可能な状態である“D”にします。次に図 2.2 の(3)にある Accelerator のスライダを上をスライドしていきます。PASTA が出荷時の状態であれば、アクチュエータ部の左側のパネル内の ACCELERATOR の値が、0%~100%の間を示します。数値が高いほどアクセルを踏み込んだ状態で、アクチュエータ部の右側のパネルの車速が上昇します。

図 2.2 イグニッション・シフトポジション・アクセル



3. しくみの解説

ここでは PASTA の出荷時の状態で PASTA のコントローラを操作した際、どのようにメッセージが流れて仮想車両が動作するかを解説します。

実車に装着されている ECU の数は、数十からときには百を超えるものもあるようです。PASTA は可搬性を重視し、小型化する事が開発の際の重要な指針の一つとして定められていました。そのため実車とは違い、機能ごとに Powertrain 系、Body 系、Chassis 系と 3 つに分け、各 ECU は、一般的な実車両の場合には複数の ECU に分かれている機能をまとめた役割を担っています。つまり ECU の数は物理的に 3 つでも、1 つの ECU には複数の CAN ID が割り振られていますので、実車同様に多様な CAN メッセージが CAN バスに流れます。

また、仮想車両のアクチュエータも機能ごとに 3 つに分け、アクチュエータの動作はアクチュエータ CPU ボード(LF74)に接続された LCD で表現し、Chassis 系にはステアリングやアクセル/ブレーキなどを模擬する物理スイッチ類が接続されています。

具体的な例として、アクセルを操作したときのメッセージフローを、[図 3.1](#) を用いて説明します。まず、(1)でアクセルのスライダを上スライドし、アクセルをどの程度踏み込んだかの量（以降踏み量と呼びます）を模擬します。すると、Chassis をシミュレートする CPU ボードにどの程度アクセルが踏まれたかが伝わります。Chassis をシミュレートする CPU ボードは(2)の経路で Chassis ECU に対し、アクセルの踏み量を伝えます。その後、Chassis ECU は踏み量の情報を CAN メッセージに変換し、(3)と(4)の経路で、CGW を介して Powertrain ECU に踏み量の情報を伝えます。Powertrain ECU は CAN メッセージで得られた踏み量の情報を(5)の経路を通して Powertrain をシミュレートする CPU ボードに踏み量の情報を伝えます。Powertrain のシミュレータは踏み量の情報からエンジンの回転などをシミュレートし、車速を計算します。その後、その車速の情報を(5)の経路を通して Powertrain ECU に伝え、Powertrain ECU は車速の情報を CAN メッセージに変換し、(4)と(3)の経路を通して Chassis ECU に伝えます。そして Body ECU は車速の情報を(2)の経路を通して Chassis をシミュレートする CPU ボードに伝え、車速がモニタに表示されます。

図 3.1 アクセルを操作した際のメッセージのフロー



4. ECU の説明

PASTA の ECU は実車の ECU とは異なり、教育や研究の目的のため、ファームウェアのリプログラミングやデバッグがしやすくなっています。図 4.1 が、ケースを取り外したときの ECU の基盤です。リプログラミング用途として RC232C ポート(1)、デバッグ用途として JTAG ポート(2)が装備されています。その他、リプログラミングの際に操作をするディップスイッチ(3)、CAN の I/O がそれぞれ 4 個(4)、アクチュエータを操作するための I/O(5)が装備されています。

図 4.1 ECU の構成

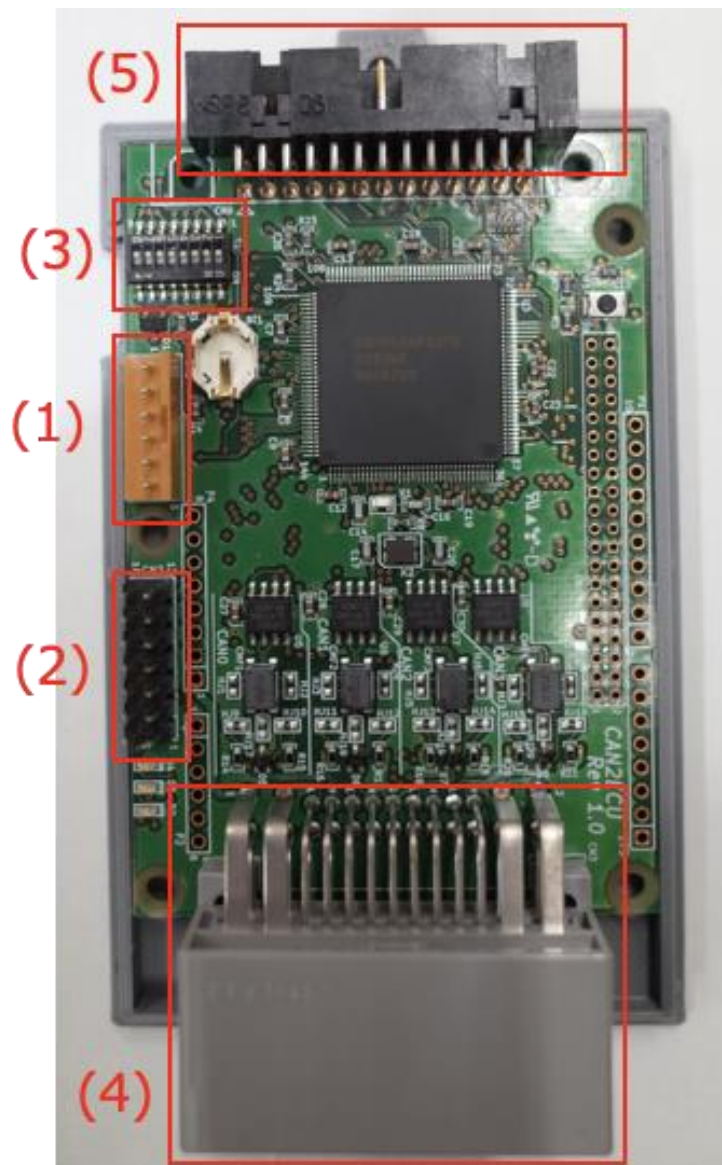


図 4.2 のように、PASTA の ECU は、PASTA に接続したまま ECU の筐体のカバーを容易に取り外せるようになっています。これは、基板上の RC232C ポートやディップスイッチなどにアクセスしやすく、リプログラミングやデバッグをしやすくするためと、基板そのものに物理的なアクセスをしたセキュリティ評価が可能になることを目的としています。

図 4.2 ECU のカバーを取り外す様子



5. 付属品の使用方法

肩掛けベルト

図 5.1 肩掛けベルトの装着



PASTA には肩掛けベルトが付属します。可搬性の高い PASTA ですが重量は 8kg 程度ありますので、肩掛けベルトを使用する事で腕に負担を掛けず楽に運ぶことができます。

取付けは左右のアタッシュケースのフックに、肩掛けベルトのフックを引っ掛けるだけです。使用中に外れないよう返しが付いていますので、外す時は返しを指で押しながら外してください。身長に合わせ長さ調整ができ、肩当てもついた本格仕様ですので、ぜひ使ってみてください。

簡易鍵

図 5.2 簡易鍵



PASTA には簡易鍵が付いています。鍵を締めなくとも、PASTA にはパッチン錠が二か所に付いているので、運搬中に突然開いてしまうような事はありません。しかし運送業者などに運んでもらう時や展示中の展示時間外など、第三者が興味本位でアタッシュケースを開けてしまうのを防止するのに鍵締めは有効です。なお付属の鍵は簡易的なものですので、ピッキングなどには無力であることをご理解の上ご使用ください。

AC アダプタと電源ケーブル

図 5.3 PASTA 内部に格納された AC アダプタ



図 5.4 AC アダプタを取り出した様子



PASTA の電源は、AC アダプタを採用しています。この AC アダプタはアタッシュケース下部のパネル下に収納したまま使用できますので、紛失したり置き場所に困ったりしないようなデザインになっています。パネル上の収納フタを開けると AC アダプタが確認でき、取り外しや交換が可能です。電圧は普通乗用車と同じ DC12V ですので、適合する DC プラグ(付録参照)に変換できれば、車用の外部バッテリーなどからも給電可能です。

OBD II -Dsub 変換ケーブル

PASTA には OBDII コネクタと Dsub9S(メス)コネクタを接続する、OBD II -Dsub 変換ケーブルが付属されます。Dsub コネクタは業界標準のピンアサインとなっており、イントリピッドコントロールシステムズ社の ValueCAN をはじめ、多くの CAN インタフェース製品との接続が可能となっています。

ValueCAN とは、同社 Vehicle Spy(ネットワーク診断/モニタ/ECU シミュレーションなどのソフトウェア)専用の USB-CAN インタフェースです。他社製品を本変換ケーブルの Dsub コネクタに接続する場合は、接続前にピンアサインが一致している事を確認してください。

OBD II -Dsub 変換ケーブルは、二股に分かれて OBDII(メス)コネクタと Dsub9S(メス)コネクタが接続されていますので、Dsub コネクタを使用しながら OBDII コネクタもご使用いただけます。

図 5.5 OBDII-Dsub 変換ケーブル



表 5.1 OBDII コネクタピンアサイン

PIN	DESCRIPTION
4	GND
6	CAN_H
14	CAN_L
16	12V

図 5.6 OBD-II コネクタ

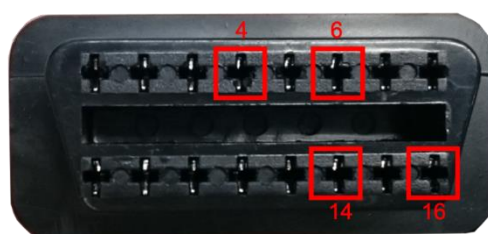
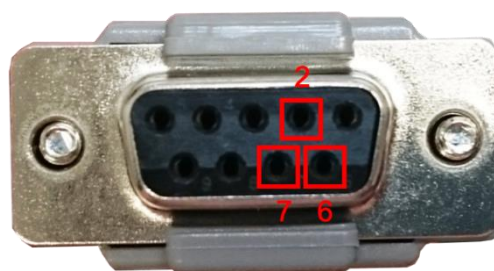


表 5.2 ValueCAN 互換コネクタピンアサイン

PIN	DESCRIPTION
2	CAN_L
6	GND
7	CAN_H

図 5.7 Dsub コネクタ



USB-CAN 変換ボード

接続方法とモニタソフトの使い方（SL-CAN）

PASTA には USB-CAN 変換ボード(以下 LFA6U)が付属しますので、すぐに CAN バスのモニタや任意データの送信などが行えます。PASTA はテストベッドである性質上、物理的な攻撃に対する耐性評価が行われることも想定してデザインされています。例えば PASTA に電氣的な負荷をかけた場合でも PC を安全に CAN バスに接続できるよう、LFA6U には CAN インタフェース側に絶縁デバイスが実装され、異常電圧や高ノイズが USB ラインへ回り込まないように保護対策が取られています。

図 5.8 USB-CAN 変換ボード（LFA6U）



LFA6U の USB インタフェースは、Windows 標準ドライバ上で仮想シリアル COM ポート(以下 COM ポート)として機能します。コマンド体系は SLCAN に類似した ASCII 形式を使用しますので、特別な制御ソフトを用意する必要は無く、COM ポートへアクセスできる TeraTerm などの汎用ターミナルソフトから簡単に使う事ができます。もちろんご自身で使いやすいアプリケーションを作ることも容易です。

SLCAN とは

serial line CAN -interface の略で、CAN アダプタとの通信に使用する簡易な ASCII プロトコルです。LAWICEL AB 社によって開発され、オープンソースで公開されているプロトコル記述を通じてデファクトスタンダードとなっています。実装レベルでは派生も多く、LFA6U も完全互換ではないため、基本的なコマンドはサポートしていますがサポートしていないコマンドもありますのでご注意ください。

では実際に、LFA6U を使用して PASTA 内を流れる CAN データをモニタしてみましょう。LFA6U のセットアップはとても簡単で、Windows PC の USB ポートに LFA6U を挿入するだけで自動的に認識されます。ただし LFA6U は基板全体を収縮チューブで覆った簡易絶縁ですので、取り扱いには注意してください。

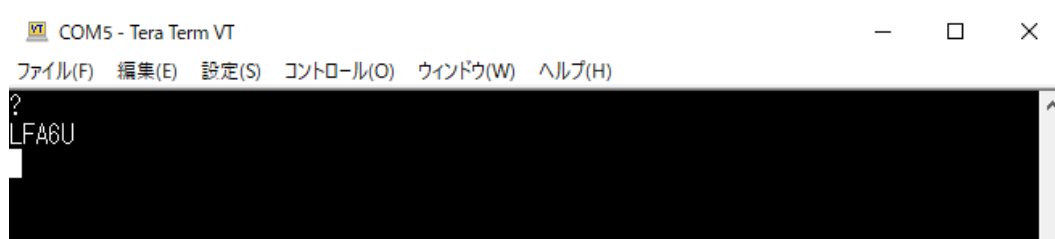
USB ポートは USB2.0/3.0 どちらでも使用できますが、万一認識しないなどのトラブルがあった場合は、他の USB2.0 ポート側に接続してみてください。PC が LFA6U を正常に認識すると、新たにシリアル COM ポートが生成されます。生成された COM 番号は、デバイスマネージャー>ポート(COM と LPT)の中に新たに増えた「USB シリアルデバイス(COM**)」で確認できます。この値はのちほど使いますのでどこかにメモをしておいてください。複数の COM ポートがあり特定できない場合は、一旦 LFA6U を引き抜き消滅したポートが該当シリアルポートです。

汎用ターミナルソフト(例では TeraTerm)を立ち上げ、該当のシリアルポートを開きます。USB ですのでボーレート設定は不要ですが、TeraTerm の場合は、端末の設定>改行コード>受信を「AUTO」にし、ローカルエコーにチェックを入れておくと見やすく表示されます。(写真は TeraTerm Version 4.97)

以降の説明は SLCAN のコマンド説明となりますが、メッセージは全てアスキー文字で、エンドコードは CR(0x0d) となります。コマンドやアンサーは""で囲み、エンドコードは明記しませんので、エンターキーを押すとコマンドの一番後ろにエンドコードが挿入されることを意識してください。

まず"?"コマンドを送信し、"LFA6U"とアンサーがある事を確認してください。アンサーが無い場合は、ターミナルが LFA6U に接続されていないので、再度 COM 番号など設定を見直してください。

図 5.9 Tera Term に“?”を入力した様子



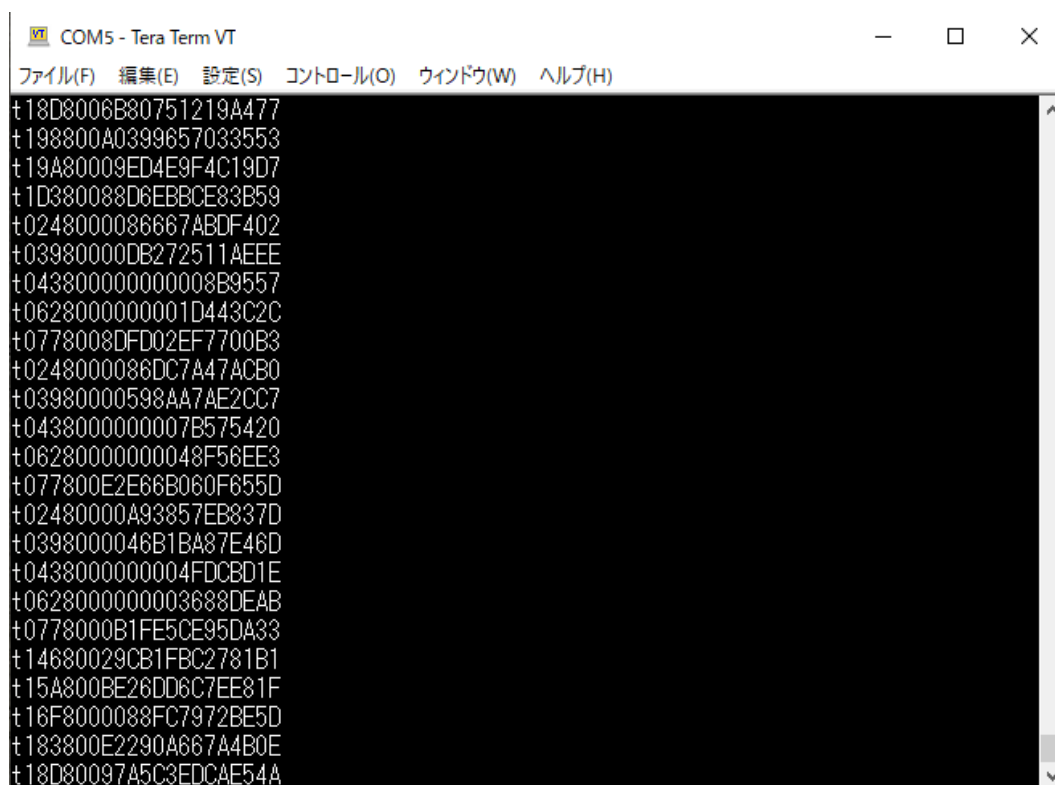
付属の LFA6U には、予め J/C コネクタ付きのケーブルが接続されていますので、それを空いている J/C に差し込みます。デフォルトでは、CGW は CAN データを全ての CAN バスに転送しますので、どこに挿入しても観測できる CAN データは同じです。

さて PASTA の電源を入れると、、、何も表示されません。始めは LAF6U の CAN ポートが閉じていますので"O"コマンドでオープンにします。通常はこの段階で CAN データが大量に表示されると思いますが、これが PASTA でやり取りされている全ての CAN データです。もし表示されない場合は、"S6"コマンドで一般的な自動車で使われている速度設定 500Kbit/s に設定してみてください。

※PASTA の CAN 通信仕様

PASTA の ECU の出荷時のファームウェアは、500Kbit/s の CAN-ID 長 11 ビット のデータフレームのみを取り扱っています。

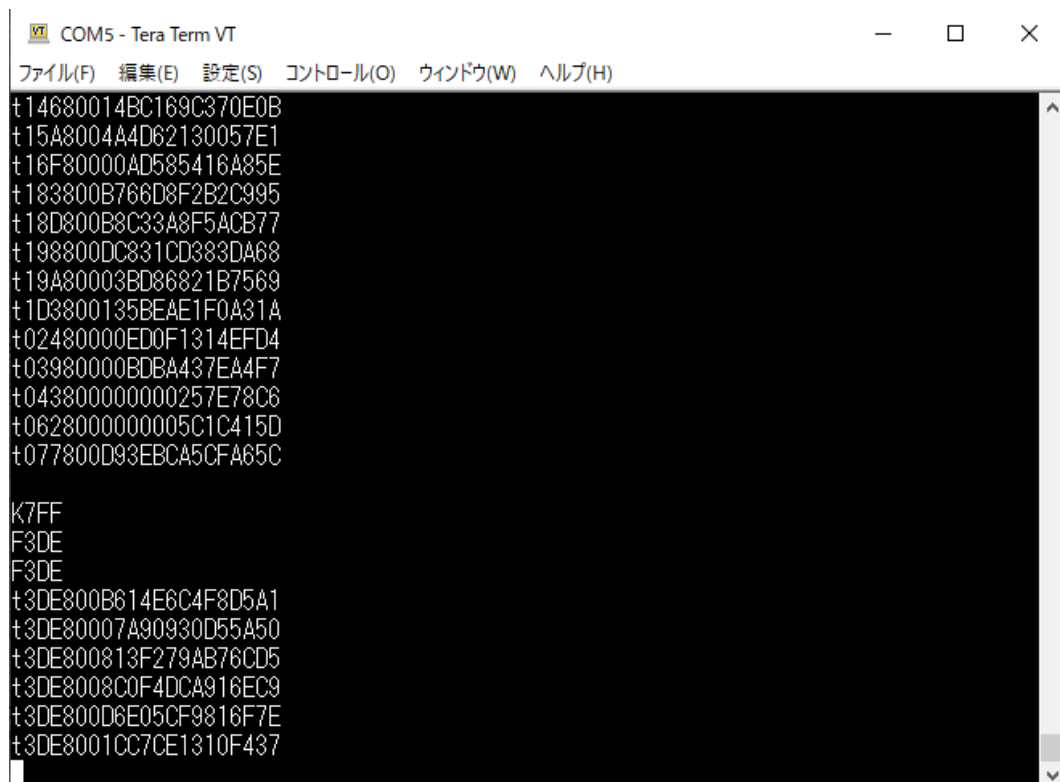
図 5.10 CAN データを観測できた場合の例



データのモニタには成功しましたが、このままでは何かデータが大量に流れている事しか分かりません。このような時はフィルタを掛け、自身が観測したい CAN ID のみを表示させます。例として、付録にある CAN-ID リストを参考に、500mSec 周期で流れているバッテリー警報のみを表示させてみましょう。

まずフィルタ範囲は CAN ID フル 11bit としますので、"K7FF"コマンドを送信します。この段階で、データの流りは止まったと思いますが、すでに F コマンドで設定された ID とマッチしていれば、その ID データが表示されます。今回表示したいのはバッテリー警報(0x3DE)ですから、"F3DE"コマンドを送信します。そうすると、CAN ID 3DE だけが 500mSec 周期で流れ出すと思います。

図 5.11 フィルタをした CAN データ



フィルタ範囲を変更する事で、複合表示も可能です。例えば、"K007"とすると下 3bit のみコンペアするようになりますので、**E と**6 の ID が見えるようになります。また任意のデータを CAN バスに送信する場合、例えば ID 01A に、データ 8 バイトが全て 0xFF のデータを送る場合は、"t01A8FFFFFFFFFFFFFFFF"コマンドを送信します。PASTA はデフォルトの状態ではセキ

セキュリティ技術が入っていません。そのため、予め CAN ID と データの意味が分かっているならば、異常な CAN メッセージを挿入することで車両を異常動作させる事が可能です。実車の場合はこんなに簡単にはいきませんが、CAN を使っている以上、セキュリティ上の脅威が潜んでいる可能性があります。ぜひ PASTA を使って、外部からの攻撃をいかに防ぐか考えてみてください。

表 5.3 LFA6U の SLCAN プロトコル仕様

コマンド	意味	
t	11 bit データフレームを送信	CAN-ID:0x012へ8バイト長のデータを送信 例) "t01280102030405060708"+CR
r	11 bit RTR フレームを送信	CAN-ID:0x321へリモートトランスミッションリクエスト送信 例) "r3210"+CR
T	29 bit データフレームを送信	CAN-ID:0x12345678へ8バイト長のデータを送信 例) "T1234567880102030405060708"+CR
R	29 bit RTR フレームを送信	-
S	CANのボーレート 設定	S0 = 10K S1 = 20K S2 = 50K S3 = 100K S4 = 125K S5 = 250K S6 = 500K S7 = 750K S8 = 1000K 例) 500Kbps : "S6"+CR
O	CANポート有効にする	例) "O"+CR
C	CANポート無効にする	例) "C"+CR
F	受信IDを設定	CAN-ID:0x123を受信対象とする 例) "F123"+CR
K	受信マスクを設定	下4ビットが異なるIDは全て受信する 例) "K7F0"+CR

付属の CAN ケーブルと J/C コネクタを使用して、 J/C の渡りケーブルを作ってみる

J/C ピンアサイン説明

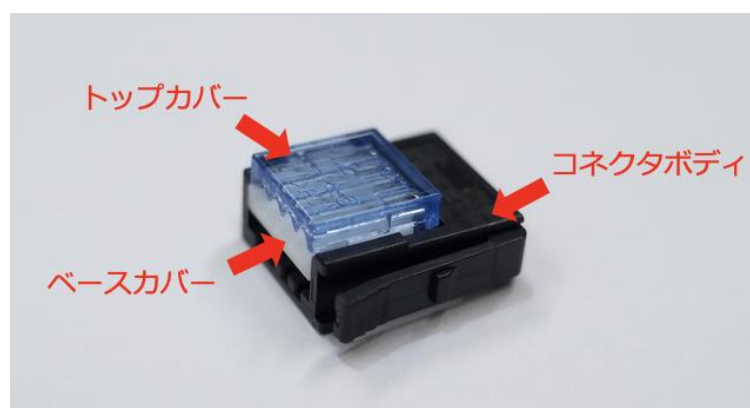
実車で使用されている CAN J/C コネクタは一般に入手が困難ですので、PASTA では入手性が良く特殊工具を必要としない e-CON を採用しています。PASTA J/C は e-CON で多く見られるピンアサインに準拠し、以下の通りとしています。

表 5.4 J/C ピンアサイン

PIN	DESCRIPTION	LINE COLOR (example)
1	+12V	red
2	CAN_L	white
3	GND	black
4	CAN_H	blue

実際の J/C に電源ラインはアサインされていませんが、ECU 増設時など配線の利便性を重視してこのようにしています。PASTA では e-CON を J/C コネクタ と表記します。

図 5.12 e-CON



PASTA には自作ケーブルが作れるように、J/C コネクタと CAN ケーブルが付属されていますので、実際に J/C 間を繋ぐ渡りケーブルを作ってみましょう。J/C コネクタと電線を接続するのはとても簡単で、電線の被覆むきすら必要ありませんが、トップカバーをコネクタボディに押し込む工具とケーブルをカットする工具は必要です。押し込み作業はペンチなどでもできますが、カバーに均一に圧力がかかるように、支点の移動でアゴ(ものを掴む部分)の大きさを調整できるプライヤ(コンビネーションプライヤ)の使用をお勧めします。

ケーブルのカットは小型のニッパをお勧めしますが、しっかりとしたハサミでも代用できます。ハサミの場合は、なるべく支点に近い場所でカットするのがコツです。

まず付属の CAN ケーブルを必要な長さにカットします。PASTA 内で使用するなら極短くて良いのですが、汎用的に使う事を考え 30cm 以上にしておく方が良いでしょう。今回は CAN 信号線 CAN-H/CAN-L のみ接続しますので、撚ってあるケーブル 1 対 2 本のみを使用します。次にカットしたケーブルを、J/C コネクタの 2 番ピンと 4 番ピンに、トップカバー（半透明部品）とベースカバー（白色部品）との間にある電線挿入口から、突き当たるまで挿入します。この時、電線色はなるべく PASTA に合わせておく事をお勧めします。

トップカバー上面から電線が奥まで挿入されていることを確認したら、プライヤのアゴをコネクタ厚になるよう調整して、コネクタ横方向から挟み、カバーをボディに押し込んでください。カバーがボディに対して水平になっていること、およびボディとカバーの間に隙間がないことをコネクタの横方向および後ろ方向から確認してください。

もしカバーが傾いていたり、押し込みが不十分でラッチがかかっていなかったりしたら、再度強く押しこんでください。これでケーブル片側の処理が完了しましたので、もう片側も同様に J/C コネクタを接続してください。この時、2 番ピンは 2 番ピン、4 番ピンは 4 番ピンにつながるように注意してください。当然ですが、同じピン同士を接続しないと通信できません。

図 5.13 プライヤによる押し込み

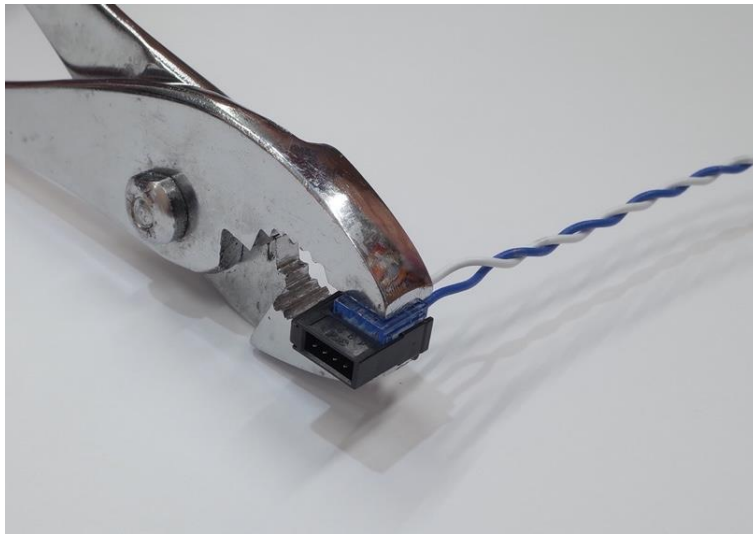


図 5.14 完成したケーブル



J/C(a) 2 ----- 2 J/C(b) CAN_L
J/C(a) 4 ----- 4 J/C(b) CAN_H

今一度ピン番号に間違いがないか目視確認してください。J/C コネクタ自体にピン番号が書かれていますが、見難い時は拡大鏡を使用すると確実です。万一 1 番ピン 12V が CAN 信号や GND に接続されると、故障の原因となりますので十分注意してください。

目視確認が正常でしたら、早速自作渡りケーブルが正常に動作するか確かめてみましょう。PASTA の出荷時設定では、3 台の ECU は全て J/C 経由で CGW に接続されており、CGW 経由で

通信をしています。この中の 1 台、例えば Powertrain につながっている J/C から、CGW につながっている J/C を引き抜きます。J/C コネクタはロック機構がありますので、抜く時は短辺にある爪を押しながら引き抜きます。

試しにこの状態で PASTA の電源を入れて、例えばステアリングやアクセルなどを操作してみましょう。Powertrain のアクチュエータは反応しませんよね？これは CGW から Powertrain への CAN バスが切断された為に、他の ECU からの情報が届かないので当然の結果です。一度電源を切り、先ほど作成した渡りケーブルを、Powertrain ECU の空いている J/C に差し込み、反対側を例えば Chassis ECU の J/C に差し込み、再度電源を入れます。

さて、結果はどうでしょうか、正常に動作しましたか？残念ながら正常に動作しなかった方は、ケーブル作りに失敗している可能性が高いですので、再度ケーブルのチェックを行ってください。

この状態では、Powertrain ECU と Chassis ECU の通信は、CGW を通さず、直接行いう事で成立するようになります。一見 CGW を通した時と挙動は変わりませんが、厳密に見るとメッセージが届く遅延時間は短くなっており、例えばステアリングを操作してから車両が実際に反応するまでのレスポンスは早くなっています。つまり CGW を導入する際はメリットデメリットを良く考え、その要求される反応速度も考慮しながら設計する必要があるという事です。

ちなみにどの位の差があるのかは、ECU のプログラムを修正して、Chassis ECU 送出タイミングと Powertrain ECU の受信タイミングを、アクチュエータコネクタの空きポートに出力し、オシロスコープで観測する事で CGW の処理遅延時間が観測できます。興味があれば、開発環境の項などを参考にチャレンジしてみてください。

※1.e-CON (Easy & Economy Connector) とは、主にセンサとターミナルの結線標準化を目的にした業界標準コネクタです。

※2.CAN (Controller Area Network) とは、ドイツの Bosch 社が開発したシリアル通信プロトコルで、1994 年に国際標準規格 (ISO11898/ISO11519) になりました。

クリッピング用分岐コネクタで信号を分岐させて CAN データをモニタしてみる

クリッピングポイントの使い方

図 5.15 分岐コネクタ



PASTA には、CAN ケーブルに割込みせて信号を引き出す為の、練習用のクリッピングポイントが用意されています。これは普段机上でセキュリティ理論の研究をされていて、CAN の配線を見たことが無い、触ったことが無い方を対象としており、実車の内張りを剥がさなくても、物理的なハッキング体験ができるよう考案された仕組みです。PASTA には空きの J/C があり、ここから CAN 信号を引き出す事はできるのですが、このクリッピングポイントを使用する事で、配線済みのケーブルにご自身で細工をして CAN 信号を取り出す体験が可能です。

ケーブルに細工を繰り返していると被覆に傷が付き交換が必要になりますが、クリッピングポイント部のケーブルは端子台に接続されており、容易に交換可能になっています。また交換用のケーブルも付属していますので、気兼ねなく何度でも練習できます。

付属している白い樹脂部品が分岐コネクタで、配線タップコネクタなどと呼ばれることもあります。この部品は汎用品ですので、添付品以外でも PASTA の CAN ケーブル(AVSS 0.3 mm²)に適合するものであれば使用できます。この分岐コネクタの金属が見える側を上から見ると溝があり、ここにケーブルを当て樹脂を折り曲げプライヤなどつぶすと、金属に電線が食い込み、導通される仕組みです。

では実際に配線分岐用のコネクタを使用して、CAN 信号を取り出してみましょう。まずは、分岐先に使用する CAN ケーブルを、J/C コネクタの説明の通り汎用的に使う事を考え 30cm 以上でカットします。このカットしたケーブルを分岐コネクタの外側の溝に嵌めるのですが、良く見ると片側の溝にはケーブルが貫通しないようストッパが付いています。切りっぱなしの分岐ケーブルを貫通させてしまうと、カット面に導線が露出していますのでショート危険があります。

ですから分岐側のケーブルを貫通させないよう、ストッパが付いている側の溝に、ケーブルカット面をストッパに押し当てる位置で挟みこみ、樹脂を折り曲げプライヤで掴みロックするまで握ります。もう片方のケーブルも同様に、別の分岐コネクタのストッパがある側に圧着させます。これで切りっぱなしケーブル端のそれぞれに、分岐コネクタが付いた状態になります。

今度は分岐元となる、クリッピングポイントのケーブルに、今圧着した分岐コネクタをカシメていきます。まず分岐コネクタの幅に合わせ、クリッピングポイントのケーブル中央付近の撚りを広げます。その状態で、ストッパが無い側の溝にケーブルを嵌め、分岐先と同様に圧着し、もう片方も同様に処理します。もしやり難ければ、端子台から一旦ケーブルを外し、圧着してから元に戻す方法もあります。この時ケーブル色を合わせておくと、結線間違いを減らす事ができお勧めです。ちなみに PASTA でのケーブル色は、CAN_L を白で統一しており、CAN_H 側は任意の色となっています。

さて皆さんはうまく作れましたか？これでクリッピングポイントから分岐するケーブルが用意できましたので、テストも兼ね早速使用してみたいと思います。ただその前に、J/C に接続できるよう分岐ケーブルの先には J/C コネクタを接続しておいてください。

次に Powertrain ECU の J/C から、CGW へ繋がっている CAN ケーブルを引き抜き、その代わりに分岐ケーブルの J/C コネクタを繋げます。J/C での実験では、CGW からのケーブルを引抜くと通信ができなくなり、その正常な動作をしなくなりましたが、この場合何事もなかったように正常動作を始めます。これは、OBDII に接続されている CAN バスがクリッピングポイントを経て CGW へつながっている為で、分岐ケーブルを Powertrain ECU の J/C に接続する事で、結果的に CGW に接続された状態になったからです。残念ながら正常に動作しなかった方は、おそらくケーブル作りに失敗している可能性が高いと思われます。再度ケーブルのチェックを行ってください。

付属ダウンロードケーブルを使って、ファームウェアを書き換えてみる

PASTA 開発環境

本製品には ECU/CGW 用の統合開発環境と、お手持ちのコンピュータと ECU/CGW を接続するためのケーブルが付属されていますので、すぐにファームウェアの開発が始められます。付属されている開発環境は、ECU/CGW で使用されているマイコンシリーズ RX63N/631 用で、C 言語コンパイラ、アセンブラ、リンカ、リモートデバッガ、フラッシュライターなどが、統合されています。

RX マイコンが採用されている理由

本製品では RH850 などの車載用マイコンを使用していません。また主に車載用マイコンで動作し機能安全規格 (ISO26262) に準拠した AUTOSAR (AUTomotive Open System ARchitecture) にも対応していません。そのため、自動車用途を期待して購入された方は不安になるかもしれません。RX マイコンが採用されているのには理由があります。それは、一般に車載用のマイコンはマイコンのベンダと秘密保持契約を結ばないとデータシートを入手することさえできません。PASTA が開発された目的は、多くの方がオープンな環境で自動車セキュリティの研究ができるプラットフォームとなることで、得られた研究成果を公表しやすくする事です。そのため、車載用のマイコンを採用しますと PASTA の本来持っていた目的を果たすことができなくなるのです。

RX マイコンは NDA が必要なく、容易に入手可能な汎用マイコンです。そのため、誰でもデータシートを入手する事ができ、また OS を搭載しないプレーンな C 言語を使用する事で、ハードウェアへのダイレクトアクセスが容易で、セキュリティ技術の実装に集中する事ができますので、車載セキュリティの教育・研究・評価を実現するのに適しています。

PASTA は出荷時から CAN の通信によって仮想車両をコントロールすることができます。出荷時に ECU に実装されているファームウェアは MIT ライセンスでオープンになっていますので、ユーザは公開されているファームウェアをベースに必要な部分のみを改変し、ご自身で考案したセキュリティ技術などを実装・評価しやすくなっています。

ファームウェアは製品付属の CD(*1)にも納められていますが、バージョンが古い可能性があります。最新版は GitHub からダウンロードする事をお薦めいたします。

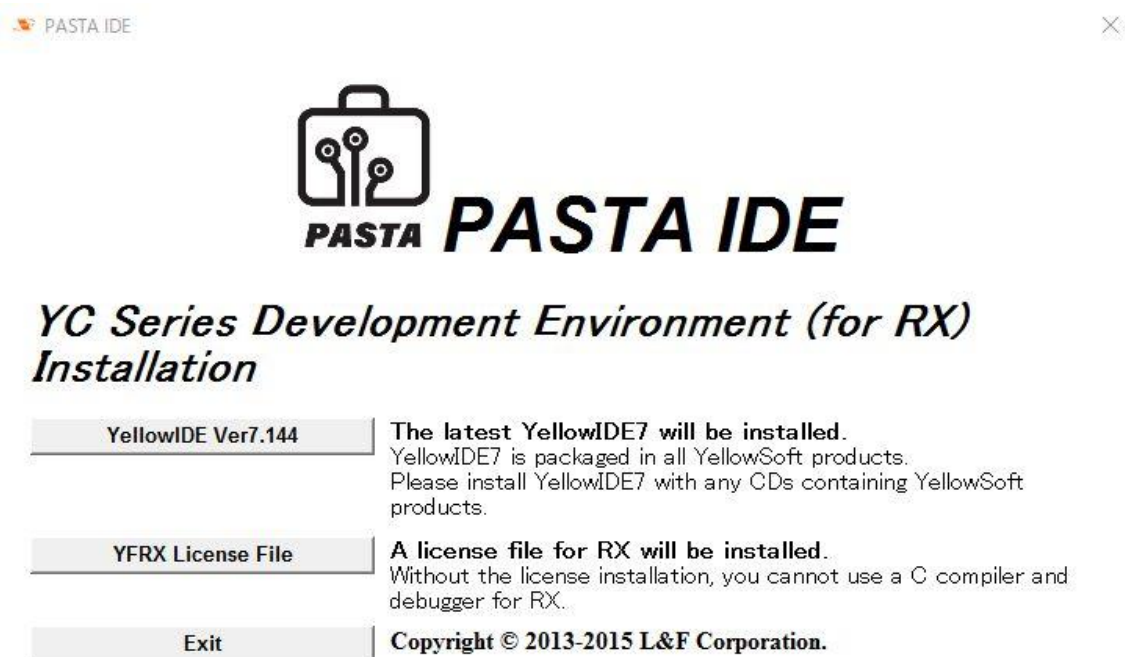
*1.CD 内の ECU/CGW ファームウェア・プロジェクトファイルの場所 : PASTA>SOFT>ECU_FW>

ECU のファームウェア書き換え

ここでは ECU のファームウェアの書き換え方法を、例を用いて具体的に説明します。

まずは、付属の CD から統合開発環境をインストールしてください。付属 CD の PASTA>SOFT>IDE>LFSelector.exe を実行すると、インストーラ選択画面が起動します。

図 5.16 IDE 立ち上げ画面

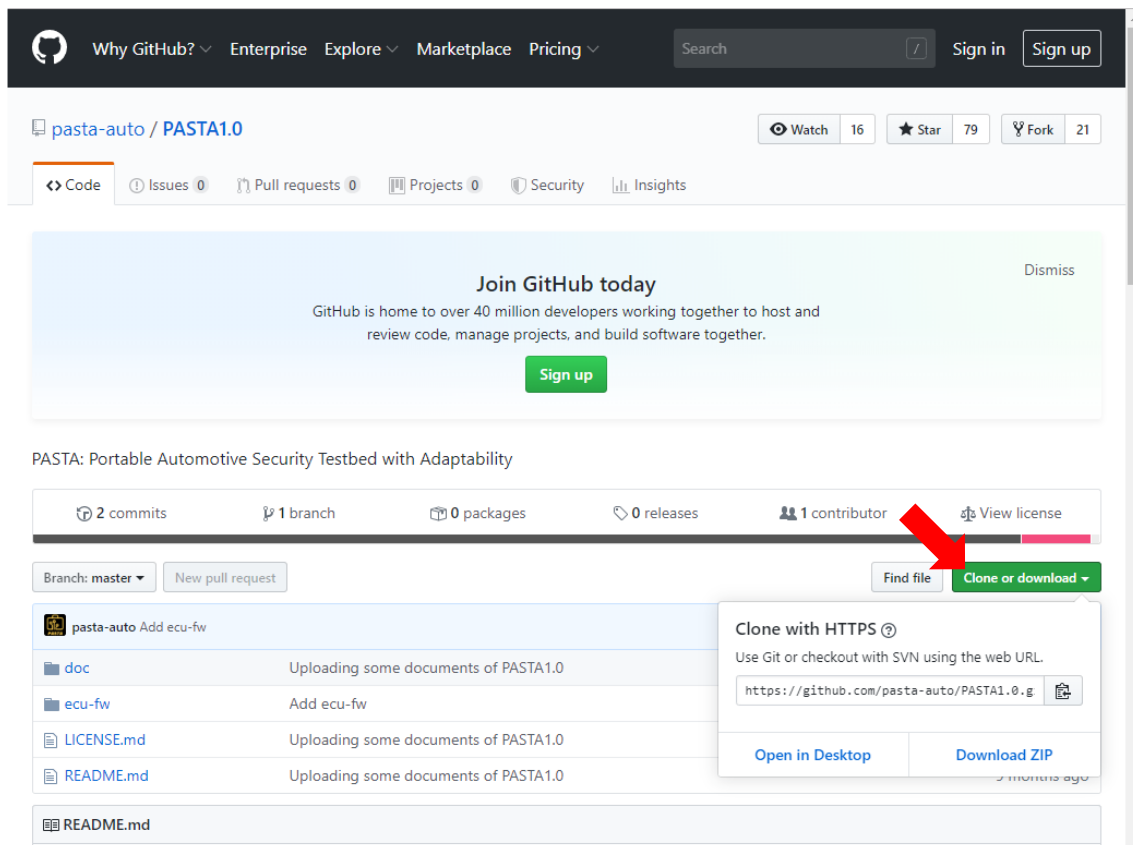


YellowIDE と書かれたボタンをクリックし、ウィザードに従い IDE のインストールを完了させます。次に LicenseFile と書かれたボタンをクリックし、ライセンス情報をインストールします。インストール自体に難しいところはありませんが、不明点は IDE のマニュアルを参照してください。

次に下記の GitHub の URL から、ECU/CGW のファームウェアプロジェクトファイル식을ダウンロードし、適当なフォルダに保存します。

<https://github.com/pasta-auto/PASTA1.0>

図 5.17 Github からのダウンロード

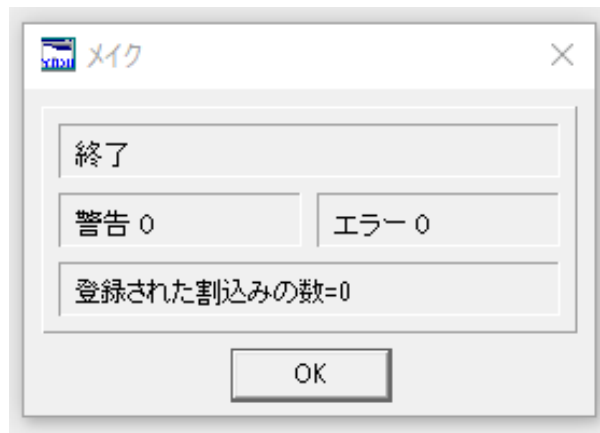


IDE を立ち上げ、Github からダウンロードしたプロジェクトファイル中の CAN2ECU.yip を開きます。

拡張子 yip が IDE に関連付けされていると、エクスプローラーなどから CAN2ECU.yip をダブルクリックするだけで IDE が起動し、プロジェクトが開かれます。プロジェクトが開いたら、まずは何も修正せずにメイクが通るか確認します。

プロジェクト>再構築 又は F10 で再構築が実行されます。

図 5.18 IDE メイクポップアップ画面

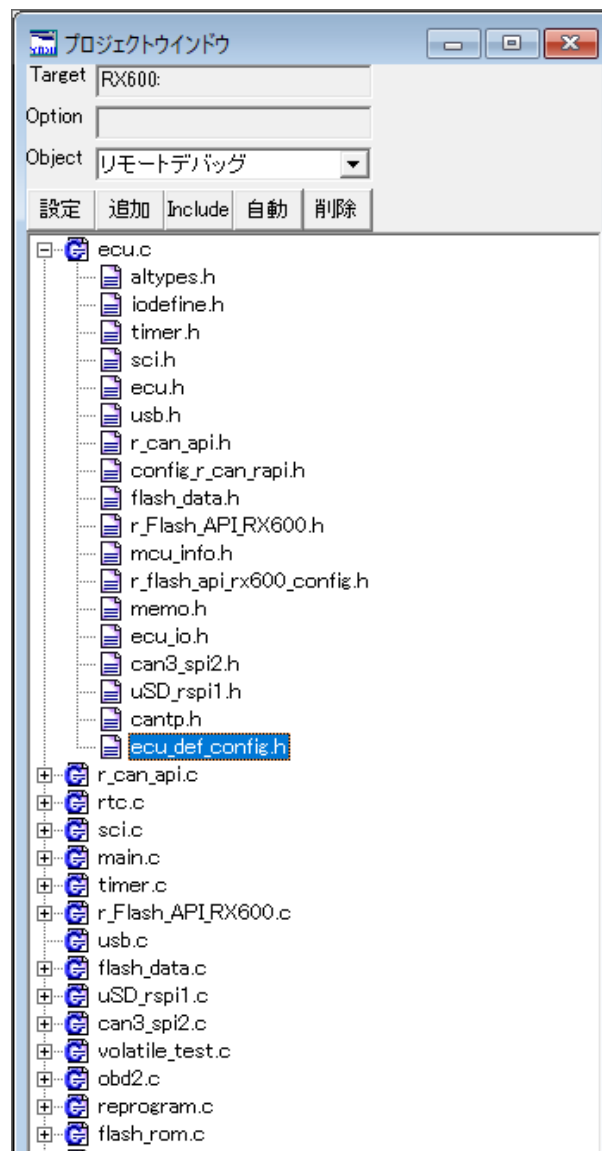


メイクのポップアップウィンドウが開くとメイク完了ですので、エラーや警告が 0 である事を確認します。この時点でエラーや警告が出る場合は、何かファイルが足りないか、ファイルが壊れている可能性がありますので、Github からのダウンロードをやり直してください。

正常にメイクができる事を確認したら、いよいよファームウェアの改変にトライしてみましょう。今回は分かりやすい例として、Chassis ECU に繋がる模擬アクチュエータの中から、ブレーキペダルの操作量に対する周期送信間隔を 20mSec から 500mSec へ遅くしてみます。送信周期とは、対象データを CAN バス上に送り出す、その間隔を指します。送信周期を遅くすると、データの変化がスムーズに伝わらなくなり、ぎくしゃくした動きになると予測できます。

ブレーキペダルの操作量は、Chassis ECU から CAN 経由で Powertrain ECU に伝わり、そこに繋がる模擬アクチュエータの車両を上から見た図のタイヤが、ブレーキ強度に比例して赤色が濃くなりますので、改修前後で変化が確認できるよう、改修前の状態で操作し動きを覚えておいてください。

図 5.19 IDE プロジェクトウィンドウ



さて修正すべきソースリストですが、ecu.c が参照するヘッダファイル ecu_def_config.h となります。ecu.c は IDE 内のプロジェクトウィンドウに表示されている筈ですが、ecu_def_config.h が見つからない場合は、ecu.c 左の+マークをクリックする事で参照ファイルが全て表示されます。ecu_def_config.h を見つけたら、プロジェクトウィンドウ内のファイル名をダブルクリックして、[図 5.20](#) のようにテキストエディタを立ち上げます。333 行目の ,20 が 20mSec の意味ですので、ここを 500 に書き換えます。

図 5.20 IDE ソースリストエディタ画面

```

322 //X add_cyceve_list(0, 0x3B3, 8, 1, 1,500,20); //P 有害排気ガス濃度・粒子状物質濃度↓
323 add_cyceve_list(0, 0x3BD, 8, 1, 1,500,21); //P エンジンオイル量↓
324 //X add_cyceve_list(0, 0x3C7, 8, 1, 1,500,22); //P 点火不良・点火タイミング異常↓
325 add_cyceve_list(0, 0x3D4, 8, 1, 1,500,23); //P エンジンスタータ故障↓
326 add_cyceve_list(0, 0x3DE, 8, 1, 1,500,24); //P バッテリー警報↓
327 //X add_cyceve_list(0, 0x42B, 8, 1, 1,500,25); //P サイドブレーキ警報↓
328 add_cyceve_list(0, 0x482, 8, 1, 1,500,26); //P エコドライブ判定↓
329 }↓
330 else↓
331 if(SELECT_ECU_UNIT == ECU_UNIT_CHASSIS)↓
332 {↓
333 add_cyceve_list(0, 0x01A, 8, 1, 1,500, 0); //C ブレーキ操作量↓
334 add_cyceve_list(0, 0x02F, 8, 1, 1, 20, 1); //C アクセル操作量↓
335 add_cyceve_list(0, 0x058, 8, 1, 1, 20, 2); //C ハンドル操作位置↓
336 add_cyceve_list(0, 0x06D, 8, 1, 1, 20, 3); //C シフトポジションスイッチ↓
337 add_cyceve_list(0, 0x083, 8, 1, 1, 20, 4); //C ウィンカー左右・ハザードスイッチ↓
338 add_cyceve_list(0, 0x098, 8, 1, 1, 20, 5); //C クラクションスイッチ↓
339 add_cyceve_list(0, 0x1A7, 8, 1, 1, 50, 6); //C ポジション・ヘッドライト・ハイビーム点
340 add_cyceve_list(0, 0x1B1, 8, 1, 1, 50, 7); //C バッシングスイッチ↓
341 add_cyceve_list(0, 0x1B8, 8, 1, 1, 50, 8); //C エンジンスタートボタン↓
342 add_cyceve_list(0, 0x1C9, 8, 1, 1, 50, 9); //C サイドブレーキ↓
343 add_cyceve_list(0, 0x25C, 8, 1, 1,100,10); //C フロントワイパー・間欠・LOW・HIGH・ウ
344 add_cyceve_list(0, 0x271, 8, 1, 1,100,11); //C リアワイパー・ウォッシャースイッチ↓
345 add_cyceve_list(0, 0x286, 8, 1, 1,100,12); //C ドアロックスイッチ・アンロックスイッチ↓
346 add_cyceve_list(0, 0x29C, 8, 1, 1,100,13); //C 右ドア・ウィンドウ昇降スイッチ↓
347 add_cyceve_list(0, 0x2B1, 8, 1, 1,100,14); //C 左ドア・ウィンドウ昇降スイッチ↓
348 }↓
349 else↓
350 if(SELECT_ECU_UNIT == ECU_UNIT_BODY)↓
351 {↓
352 add_cyceve_list(0, 0x08D, 8, 1, 1, 20, 0); //B ウィンカー左右点灯状態↓
353 add_cyceve_list(0, 0x0A2, 8, 1, 1, 20, 1); //B クラクション鳴動↓
354 add_cyceve_list(0, 0x0B4, 8, 1, 1, 20, 2); //B エアバッグ作動スイッチ↓
355 add_cyceve_list(0, 0x1BB, 8, 1, 1, 50, 3); //B ポジション・ヘッドライト・ハイビーム点
356 add_cyceve_list(0, 0x266, 8, 1, 1,100, 4); //B フロントワイパー・間欠・LOW・HIGH・ウ

```

add_cyceve_list(0, 0x01A, 8, 1, 1,500, 0); //C ブレーキ操作量



add_cyceve_list(0, 0x01A, 8, 1, 1,20, 0); //C ブレーキ操作量

テキストエディタ右上の×をクリックし、保存しますかで「はい」をクリックします。

より深い理解のために 1

IDE テキストエディタの文字コードは、シフト JIS 専用となっており UTF-8 には対応していません。もし普段 UTF-8 をお使いの場合、IDE エディタで開くと文字化けしますので、シフト JIS で保存し直すか、2 バイトコードは使わないようにしてください。UTF-8 のまま外部のエディタで編集したとしても、リモートデバッガも UTF-8 に対応しておりませんので、文字化けは回避できません。

その後、プロジェクト>メイク 又は F9 キー 又は ■メイクのスピードボタンでメイクを実行し、エラーや警告が 0 である事を確認します。初めに試した再構築ではありませんので注意してください。

より深い理解のために 2

再構築(F10)は、無条件に全てのソースリストをメイクしますが、メイク(F9)は、変更のあったソースリストのみメイクします。ちなみにメイクとは、コンパイル〜リンク〜実行ファイル生成までの一連の流れを指します。

図 5.21 ダウンロードケーブル



さてこれで書換えるべき実行ファイルが用意できましたので、次はコンピュータと ECU を専用のダウンロードケーブルで接続します。

図 5.22 ECU の基板



ダウンロードケーブルを接続するには ECU ケースを開け、基盤を剥き出しの状態にします。ECU ケースは上下を接続するために 6 点のツメで固定されていますので、コネクタの隙間などに指を掛け少し強く上方に引き上げます。

ケースが外しにくい場合はケーブルを外し、PASTA から一旦 ECU を取り外しますが、その際グレーのコネクタは爪を下に押し下げながら引き抜きますので注意してください。

ECU とパネルは 1 箇所ネジ止めされていますので、+ドライバーが必要となります。

図 5.23 ECU の基板へのダウンロードケーブルの接続



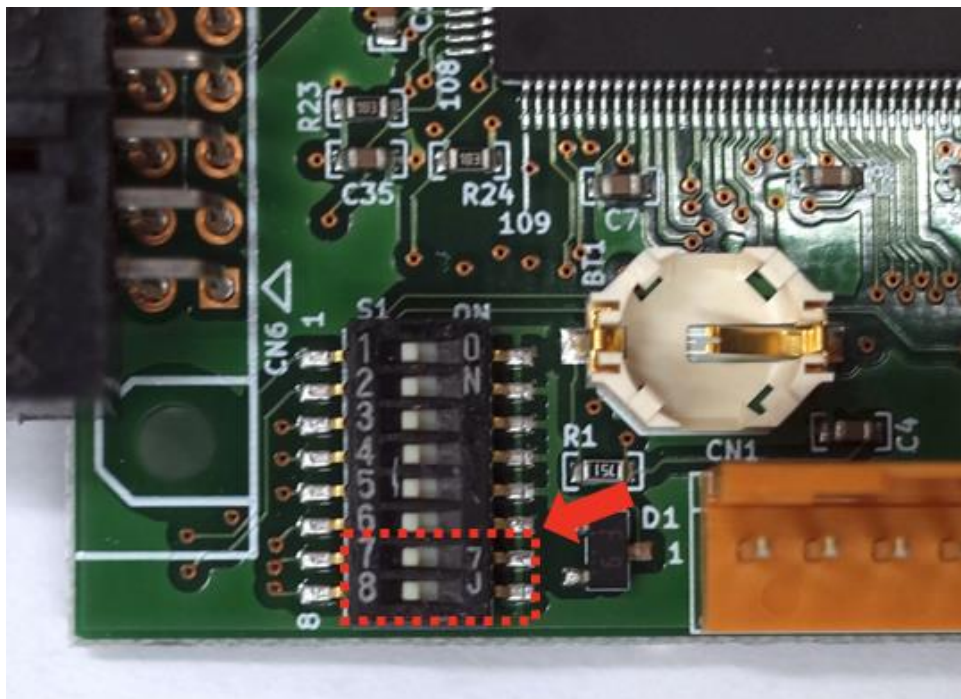
中の基板が見えたら、CN1 茶色コネクタとダウンロードケーブルの茶色コネクタ(IL-G6P)を接続します。

次にダウンロードケーブルの Dsub9s(メス)コネクタを PC の RS232C コネクタに接続するのですが、最近の PC には RS232C インタフェースは付いていませんので、付属の USB シリアル変換ケーブル(REX-USB60F)を経由して PC と USB で接続します。

USB シリアル変換ケーブルは初回接続時のみドライバのインストールが必要になりますので、USB シリアル変換ケーブルを PC に接続する前にセットアップをお願いします。

セットアップは、付属 CD の PASTA¥SOFT¥USB-Serial¥USB60F_Setup.exe を実行します。

図 5.24 ディップスイッチの 7 と 8



ダウンロードケーブルが接続できたら、S1 ディップスイッチの 7,8 を ON にします。このモードでは、生成した実行ファイルを IDE リモートデバッガからダウンロード実行する事で、ファームウェアの書き換えが実行されます。

ディップスイッチは精密ドライバやシャープペンシルの先端など、金属製の硬いもので操作をすると、樹脂部に傷が付き故障の原因となりますので、つまようじなど木製の柔らかいもので操作する事を薦めます。

より深い理解のために 3

ECU の書き換え方法にはいくつか方法があり、今回ご紹介の方法はブートローダーは消さずに、アプリケーションのみを書き換える方法となります。

他には、以下のような書き換え方法があります。

■主要コマンドと応答(デリミタ=CR) デバッガを使用しながら任意のタイミングで書き換える方法

S1 ディップスイッチの 8 のみを ON にします。

基本的に前記説明と同様ですが、このままではフラッシュの書き換えは行わず、RAM 上で動作しています。CN6 の 26 ピンコネクタに接続した、模擬アクチュエータ専用ケーブルをターミナルソフトに接続し、そこからコマンドを送信する事で、フラッシュメモリへ実行ファイルを書き込みます。

主要コマンドと応答(デリミタ=CR)

BOOTCLEAR 対象フラッシュメモリ領域を消去 > BOOTCLEAR OK
BOOTCOPY 対象フラッシュメモリ領域に書込 > BOOTCOPY OK

■ブートローダー自体も書き換える方法

フラッシュメモリライター FWRITE2 を立ち上げ、リモートデバッグ用のモニタソフトを書き込みます。通常はブートローダーの書き換えは実施しませんが、誤って消してしまった、自作のブートローダーを使用したりしたいなどでは必要となります。

さてここまでできたらいいよ電源を入れますが、ECU への給電はグレーのコネクタから行っている、PASTA のアタッシュケースから ECU を取り外した場合はケーブルを元に戻してください。

また先程の作業で IDE は起動したままだと思いますが、終了してしまった方は再度 IDE を起動し、対象のプロジェクトを開いてください。

PASTA の電源を入れた状態で、プロジェクト>実行 又は F5 キー 又は ▶実行スピードボタンで、リモートデバッガ(YScope)が起動します。起動に成功すると YScope ウィンドウが開きますので、デバッグ>実行 又は ▶実行スピードボタンで、ECU へ実行ファイルを転送後すぐに実行を開始します。

実行を開始するとプログラムフラッシュへの書き込みが始まり、書き換えた実行ファイルが起動し D2 L E D が点滅を始めれば、書き換え完了です。

なお書き換え時間はほぼ 1 秒以内です。もしそれ以上かかるようならフラッシュメモリの劣化が疑われます。フラッシュメモリのメーカー書き換え保証回数は 1000 回ですが、保持時間の保証が無くなるだけでこれ以上の書き換えも可能です。最後に S1 ディップスイッチの 7,8 を OFF にするのを忘れずに、ECU ケースを元に戻せば一連の作業は完了です。

Yscope が起動しない場合、表示されるポップアップ画面より原因を特定して対策してください。

Yscope の起動に失敗した場合の代表的な警告メッセージと対策

メッセージ :

「通信ポートのオープンに失敗しました。」

対策 :

USB シリアル変換ケーブルに割り当てられたシリアル COM 番号が異なるか、別のアプリですでに使われています。デバイスマネージャーから 製造元 : RATOC の USB Serial Port を見つけ、COM 番号を控えます。YScope ウィンドウの 設定>システム設定>通信ポートタブ の通信ポートを合わせてください。またその右側ポート設定ボタンから「ビット/秒:38000bps、データビット:8、パリティ:なし、ストップビット:1、フロー制御:なし」になっている事を確認します。

メッセージ：

「通信ポートがタイムアウトしました。ケーブル、設定を確認してください。」

対策：

ECU の電源が入っていない可能性

ECU とのダウンロードケーブル接続が正しくない可能性

ECU のディップスイッチ設定が正しくない可能性

ECU にブートルoader(モニタ)が書かれていない可能性

ECU のブートルoader(モニタ)と通信設定が不一致の可能性

などが考えられますので、再度確認してください。

さて改変後の動きはどのように変わったでしょうか？

ブレーキを操作して、書き換え前との動作比較をしてみてください。

6. 付録

PASTA 基本仕様

図 6.1 PASTA を閉じた外観



消費電流	約 1.6A
外形寸法	W470 x H355 x D150 mm（突起含む）
重量	約 8.0kg
梱包外形寸法	約 W480 x H380 x D175 mm
梱包重量	約 8.6kg
梱包主材質	ダンボール

ECU 基本仕様

消費電流	約 110mA (CAN 未接続、アイドリング時)
入力電圧	DC12V (8~55V)
外形寸法	W100 x D65 x H25 mm
重量	約 94 g
基板外形寸法	85.3 x 57.6 mm (突起含まず)
基板重量	約 40 g
主要部品	
CPU	RX63N/RX631 シリーズ (Renesas Electronics Corporation)
形式	R5F563NFDDFB ※Ethernet 非搭載品 R5F5631FDDFB の場合もあります
CLK	100 MHz ※最大動作クロック
Program Flash	2 MByte
Data Flash	32 KByte
SRAM	256 KByte
CAN transceiver	TJA1050 (NXP) ※旧社名 PHILIPS
CMC (Common Mode Chokes)	ACT45B-510-2P (TDK) ※互換品 B82787C0513H002(EPCOS)の場合もあります。
RS232C transceiver	SN65C3223EDBR (TI) ※互換品の場合もあります。

AC アダプタ仕様

図 6.2 AC アダプタ



電源容量	60W（無負荷時の電力消費 < 0.15W）
入力電圧	AC90 ~ 264V
入力形状	3 極インレット（IEC320/C14）
出力電圧	DC12VA
出力電流	5A
出力形状	ケーブル直出し先端 DC プラグ (外形 φ5.5mm/内径 φ2.5mm/センター+)
外形寸法	L115 x W53 x H38mm
効率基準	EU CoC ティア 2、US DoE レベル VI NRCan、ErP 2011 フェーズ II 適合
安全規格	UL/cUL、GS、PSE、CE、FCC IEC62368-1、IEC60950-1 準拠 制限電源（LPS : Limited Power Source）定格 過電圧および短絡に対する保護機能を搭載
メーカー形式	SDI65-12-U-P6（CUI Global Inc.）

※AC アダプタの仕様は、製造メーカーのモデルチェンジなどで予告なく変更される場合があります。

LFA6U 仕様

Figure 6.3 LFA6U-PASTA



基板外形寸法	62 x 20 mm (突起含まず)
基板重量	約 10 g
CPU	RX631 シリーズ (Renesas Electronics Corporation)
形式	R5F5631MDDFM
CLK	100 MHz ※最大動作クロック
Program Flash	256 KByte
Data Flash	32 KByte
SRAM	64 KByte
CAN transceiver	TJA1050 (NXP) ※旧社名 PHILIPS

CAN ID リスト

Chassis Powertrain Body ※送信ECUを基準とした色分け

送信	受信	ID	HEX	名称	内容	適用	周期 (ms)	データ 長	OFF/最小	ON/最大	単位
Chassis	Powertrain	26	01A	ブレーキ操作量	指令	操作	20	2	0	1023	%
Powertrain	Chassis	36	024	ブレーキ出力量	報告	出力	20	2	0	1023	%
Powertrain	Chassis	326	146	ブレーキオイル量	報告	ダミー	50	2	0	1023	%
Powertrain	Chassis	346	15A	アンチロックブレーキ作動	報告	ダミー	50	1	00	01	ON/OFF
Chassis	Powertrain	47	02F	アクセル操作量	指令	操作	20	2	0	1023	%
Powertrain	Chassis	57	039	スロットル位置	報告	出力	20	2	0	1023	%
Powertrain	Chassis	367	16F	スロットル調整	報告	出力	50	2	0	1023	%
Powertrain	Chassis	67	043	エンジン回転数・速度	報告	計測	20	4	0/-32767	65535/32767	rpm,km/h
Powertrain	Chassis	387	183	エンジン冷却水温度	報告	ダミー	50	1	0	255	°C
Powertrain	Chassis	397	18D	エンジン故障	警報	ダミー	50	1	00	FF	ON/OFF
Chassis	Powertrain	88	058	ハンドル操作位置	指令	操作	20	2	-511	511	%
Powertrain	Chassis	98	062	パワステ出力量	報告	出力	20	2	0	100	%
Powertrain	Chassis	408	198	パワステ故障	警告	ダミー	50	1	00	01	ON/OFF
Chassis	Powertrain	109	06D	シフトポジションスイッチ	指令	操作	20	1	00	10,20	ON/OFF
Powertrain	Chassis	119	077	シフトポジション位置	報告	出力	20	1	00	01,02,04,08,10	ON/OFF
Chassis	Powertrain	440	1B8	イグニッションボタン	指令	操作	50	1	00	01	ON/OFF
Powertrain	Chassis	410	19A	エンジン運転中	報告	出力	50	1	00	01	ON/OFF
Powertrain	Chassis	980	3D4	エンジンスタータ故障	警告	ダミー	500	1	00	01	ON/OFF
Powertrain	Chassis	990	3DE	バッテリー警報	警告	ダミー	500	1	00	01	ON/OFF
Chassis	Body	131	083	ウイinker左右・ハザードスイッチ	指令	操作	20	1	00	01,02,04	ON/OFF
Body	Chassis	141	08D	ウイinker左右点灯状態	報告	出力	20	1	00	01,02,03	ON/OFF
Chassis	Body	152	098	クラクションスイッチ	指令	操作	20	1	00	01	ON/OFF
Body	Chassis	162	0A2	クラクション鳴動	報告	出力	20	1	00	01	ON/OFF
Chassis	Body	423	1A7	ポジション・ヘッドライト・ハイビームスイッチ	指令	操作	50	1	00	01,02,04	ON/OFF
Chassis	Body	433	1B1	バッシングスイッチ	指令	操作	50	1	00	08	ON/OFF
Body	Chassis	443	1BB	ポジション・ヘッドライト・ハイビーム点灯状態	報告	出力	50	1	00	01,02,04,08	ON/OFF
Chassis	Body	604	25C	フロントワイパー 間欠・LOW・HIGH・ウォッシャースイッチ	指令	操作	100	1	00	01,02,04,10	ON/OFF
Body	Chassis	614	266	フロントワイパー 間欠・LOW・HIGH・ウォッシャー動作状態	報告	出力	100	1	00	01,02,04,10	ON/OFF
Chassis	Body	625	271	リアワイパー ウォッシャースイッチ	指令	操作	100	1	00	01,02,03	ON/OFF
Body	Chassis	635	27B	リアワイパー ウォッシャー動作状態	報告	出力	100	1	00	01,02,03	ON/OFF
Chassis	Body	646	286	ドアロックスイッチ・アンロックスイッチ	指令	操作	100	1	00	01,02,03	ON/OFF
Body	Chassis	656	290	ドア開閉・施錠状態	報告	出力	100	2	00/00	01,02,03/01,02,03	ON/OFF
Body	Chassis	1056	420	ドアロック駆動装置故障	警告	ダミー	500	1	00	01,02,03	ON/OFF
Chassis	Powertrain	457	1C9	サイドブレーキ	指令	操作	50	1	00	01	ON/OFF
Powertrain	Chassis	467	1D3	サイドブレーキ動作状態	報告	出力	50	1	00	01	ON/OFF
Chassis	Body	668	29C	右ドア・ウィンドウ昇降スイッチ	指令	操作	100	1	00	01,02	ON/OFF
Body	Chassis	678	2A6	右ドア・ウィンドウ位置・リミットスイッチ状態	報告	出力	100	2	00/0	01,02/255	ON/OFF,%
Chassis	Body	689	2B1	左ドア・ウィンドウ昇降スイッチ	指令	操作	100	1	00	01,02	ON/OFF
Body	Chassis	699	2BB	左ドア・ウィンドウ位置・リミットスイッチ状態	報告	出力	100	2	00/0	01,02/255	ON/OFF,%
Body	Chassis	180	0B4	エアバッグ作動スイッチ	報告	ダミー	20	1	00	01	ON/OFF
Body	Chassis	1111	457	シートベルトセンサー	報告	ダミー	500	1	00	01	ON/OFF
Body	Chassis	1121	461	シートベルト警報	警告	ダミー	500	1	00	01	ON/OFF
Body	Chassis	1132	46C	ボンネット開閉スイッチ	報告	ダミー	500	1	00	01	ON/OFF
Body	Chassis	1143	477	トランク開閉スイッチ	報告	ダミー	500	1	00	01	ON/OFF
Powertrain	Chassis	1154	482	エコドライブ判定	報告	ダミー	500	1	00	01	ON/OFF

※データ配置はビッグエンディアン

CARLA 公開版

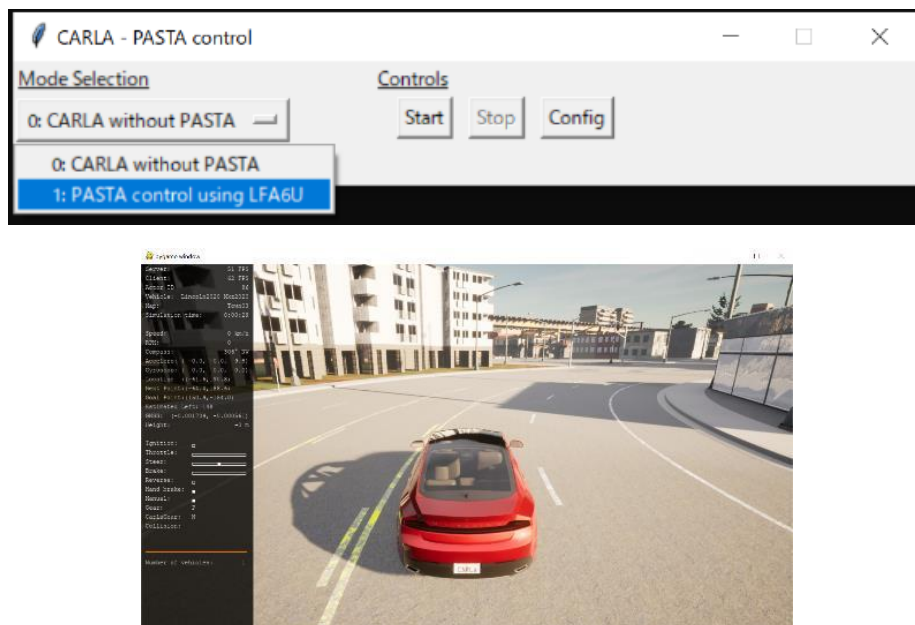
現在、PASTA1.0 に対応した自動車シミュレータ CARLA Co-Simulator(以後：CARLA)が公開されています。

<https://github.com/pasta-auto/CARLA-PASTA>

※インストール手順は、「PASTA-CARLA CoSim Installation Guide」を参照してください。

PASTA1.0 との接続は LAF6U -PASTA(CARLA がインストールされた PC の USB ポートに接続)を使用します。

PASTA1.0 と CARLA をリンクさせるためには、CARLA 起動時に表示されます「CARLA-PASTA control」の Mode Selection メニューより[1:PASTA control using LFA6U]を選択します。
次に Controls メニューの[Start]をクリックしリンクさせます。



CARLA の起動画面

※PASTA1.0 に PASTA-ハンドルコントローラを接続しているときステアリング(ハンドル)がガタつく場合は、「CARLA-PASTA control」の Controls メニューより、一度ストップ[Stop]し再スタート[Start]を実行してください。

7. お問い合わせ先

販売元



株式会社チップワンストップ
(Chip One Stop, Inc.)

PASTA 特設サイト

<https://www.chip1stop.com/sp/products/toyota-pasta>

PASTA に関するお問い合わせはこちら

<https://www.chip1stop.com/sp/products/toyota-pasta-form>

製造元



株式会社エル・アンド・エフ (LandF Co.,Ltd.)

<https://www.l-and-f.co.jp>

開発元

トヨタ自動車株式会社 (TOYOTA MOTOR CORPORATION)