



Sunny Giken Inc.

MicroPecker CAN アプリケーション開発ライブラリ
《 S810-MP-SDK2N 》 導入ご検討用詳細資料



目次

1.はじめに.....	1
1.1.用語.....	1
1.2.SDK できること.....	1
1.3.構成.....	2
2.開発環境.....	3
3.API 一覧.....	4
4.シーケンス.....	5
4.1.モニタリングシーケンス.....	5
4.2.シミュレーションシーケンス.....	6
5.機能比較.....	7

1. はじめに

本ドキュメントは MicroPecker CAN アプリケーション開発ライブラリ《S810-MP-SDK2N》(以下本 SDK)の導入をご検討されているお客様へ、その機能の詳細を記載した資料です。

尚本ドキュメントは CAN の基本的な知識がある事を前提として記述しています。

1.1. 用語

本ドキュメントで使用する用語の説明を以下に記します。

表 1:用語説明

用語	説明	備考
MicroPecker	本 SDK にて使用する CAN モニタリングを行うためのハードウェア。USB にて WinodowsPC と接続する。	本 SDK を使用するためには必須
モニタリング	CAN バス上の信号を取得、監視すること。	
CAN フレーム	CAN の通信単位	
スロット	送信する CAN フレームの情報をあらかじめ設定しておくためのバッファ。	

1.2. SDK でできること

- MicroPecker の接続/切断と LED 操作

PC に接続されている MicroPecker を認識、接続し MicroPecker 本体にある LED の点灯/消灯/点滅をすることができます。

- CAN のモニタリング

MicroPecker に接続されたターゲットから送信された CAN フレームをモニタリングし、そのデータを取得することができます。

- CAN のシミュレーション

MicroPecker の CAN 機能を用い、任意のタイミングで CAN フレームを送信することができます。



1.3. 構成

本 SDK を使用する場合のシステム構成は以下の通りです。

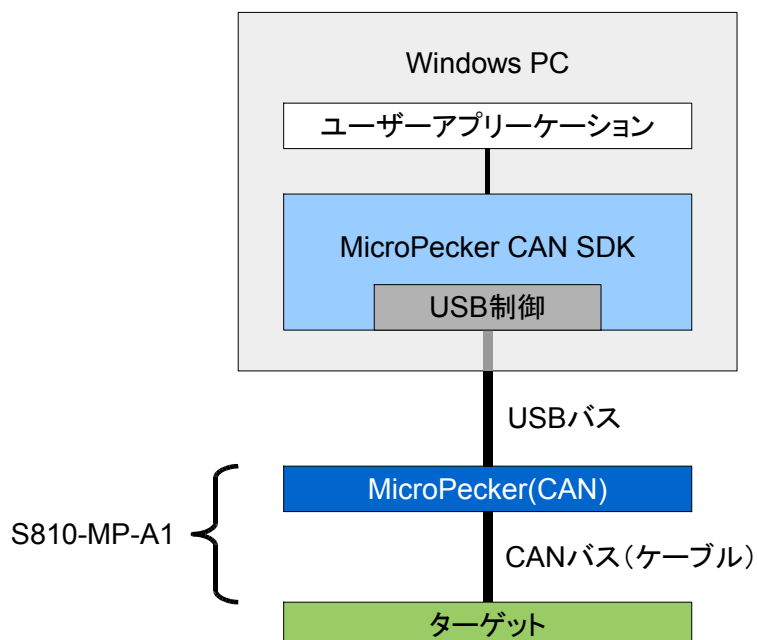


図 1: SDK 使用システム構成

ユーザーアプリケーションでは MicroPecker 本体の操作及び CAN 機能の設定、CAN 通信データの受取りを行うのみで USB (接続・切断・通信) 制御は SDK が担保します。

本 SDK を使用するには CAN のライセンスを保持した MicroPecker 本体並びに CAN ケーブルが必要となります (MicroPecker CAN Analyzer[S810-MP-A1]または MicroPecker CAN Analyzer Software License Kit[S810-MP-SA1]にて提供)。



2. 開発環境

本 SDK を使用することができる開発環境は以下の通りです。

表 2: 開発環境

PC	Microsoft Windows が動作する IBM PC/AT 互換機
OS (※1)(※2)	Microsoft Windows 10 (64bit) Microsoft Windows 8.1 (64bit) Microsoft Windows 7 (32bit/64bit) Microsoft Windows Vista (32bit)
開発環境	Microsoft Visual C++ (Ver.6.0 以降) Microsoft Visual C# (2005 以降)(※3) Microsoft Visual Basic (Ver.6.0) Microsoft Visual Basic .NET (2005 以降)(※3) Embarcadero C++ Builder (Ver.7 以降) Embarcadero Delphi (Ver.7.0 以降)
CPU	Celeron 1.6GHz 以上を推奨
ハードディスク	10Mbyte 以上の空き容量
メモリ	1Gbyte 以上を推奨
USB ポート	USB2.0 搭載 (Hi-Speed)
ログインユーザ	管理者権限を持つユーザ (USBドライバのインストール時に必要)

(※1) 64ビット OS の場合でも、32ビットアプリケーションのみ開発可能です。

(※2) Windows7 の XP モードなど、仮想環境での動作は非対応です。

(※3) 対応.NETバージョンは 2.0 および 4.0 です。

3. API 一覧

本 SDK にて提供される API は以下の通りです。

表 3: API 一覧

関数名	機能	説明
MPCANOpen	MicroPecker 接続	SDK の初期化と MicroPecker (CAN) との接続を行う。
MPCANClose	MicroPecker 切断	MicroPecker (CAN) の切断を行う。
MPCANSetLED	LED 操作	MicroPecker にある LED (赤および緑) を操作する。
MPCANGetStatus	ステータス取得	ステータスを取得する
MPCANSetParamSimple	簡易パラメータ設定	CAN の動作パラメータを設定する (動作モード、ボーレート、サンプルポイント、Ack 有無)。
MPCANSetParamDetail	詳細パラメータ設定	上記ボーレート及びサンプルポイント設定をより詳細に設定する (プリスケータ、セグメント等)
MPCANSetSlot	スロット設定	スロット情報を設定する。
MPCANMonitorStart	モニタリング開始	モニタリング (シミュレーション) を開始する。
MPCANMonitorStop	モニタリング停止	モニタリングを停止する。
MPCANGetTimeStamp	タイムスタンプ取得	現在のタイムスタンプを取得する。
MPCANChangeSlotData	スロットデータ変更	指定スロットの CAN フレームを変更する。
MPCANSendSlot	スロット送信	指定スロットの CAN フレームを送信する。
MPCANSendDirect	ダイレクト送信	CAN フレームをスロットを介さず直接送信する。
MPCANGetLog	ログ取得	モニタログを取得する。



4. シーケンス

本 SDK にてソフトウェアを開発する際の API コールシーケンスは以下の通りです。

実線枠 の項目は必須の処理、**点線枠** の項目は必須でない処理をそれぞれ表します。

4.1. モニタリングシーケンス

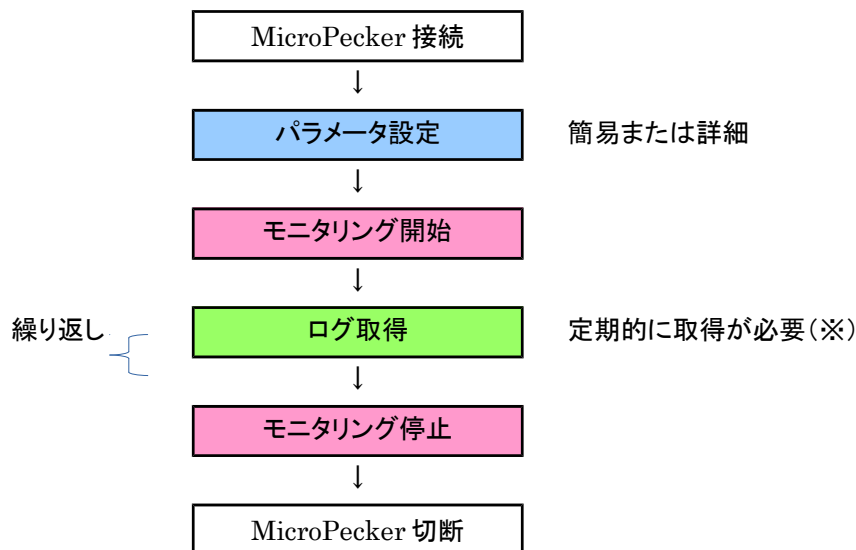


図 2: モニタリングシーケンス

※ 定期的に取得しない場合、MicroPecker 内部のバッファオーバーが発生し、以降のログが正しく取得できません(モニタリング停止→再開で再取得は可能です)。

4.2. シミュレーションシーケンス

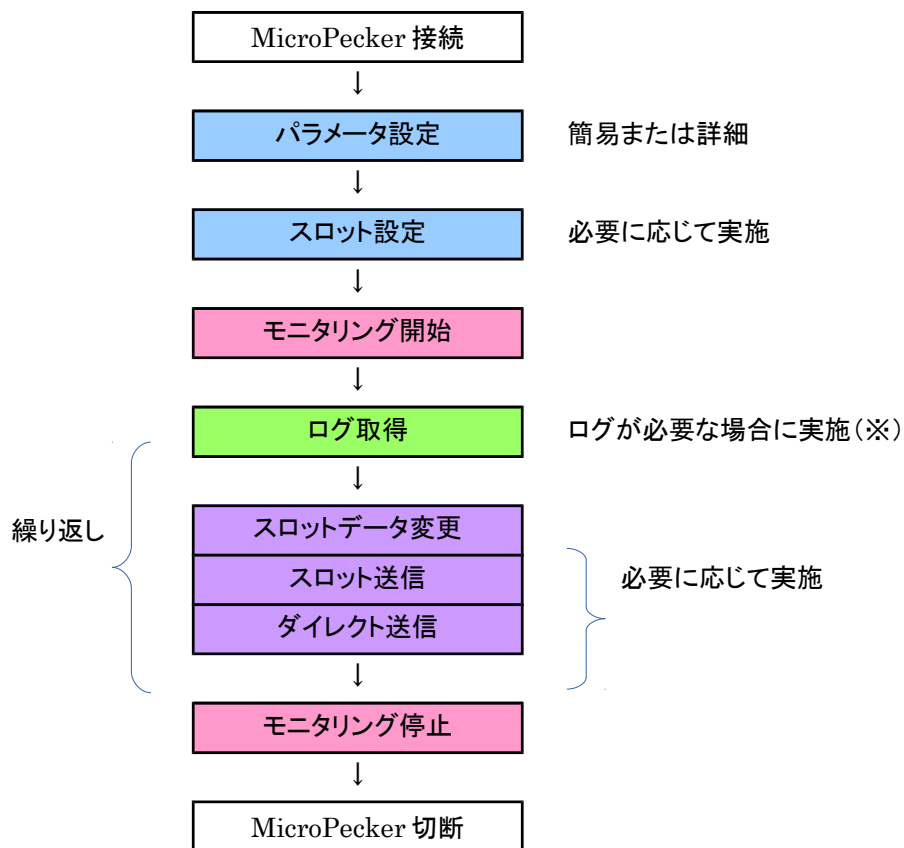


図 3: シミュレーションシーケンス

※ 定期的に取得しない場合、MicroPecker 内部のバッファオーバーが発生します。シミュレーション処理に問題はありませんが、バッファオーバー発生後のログは正しく取得できません(モニタ停止→再開で再取得可能)。



5. 機能比較

製品版 MicroPecker (CAN アナライザ) との機能差異は以下の通りです。

表 4: 機能比較

項目	製品版との差異	備考
複数 Ch 同時使用	MicroPecker Analyzer では最大 6Ch までの MicroPecker を同時に操作可能だが、SDK では 1Ch での使用のみ対応。	LIN SDK との同時使用は可能
ログ再生機能	MicroPecker Analyzer では CAN ログを読み込み、そのログを再生可能だが、SDK では複数の CAN フレームを一度に設定し送信させる API は非対応。	ダイレクト送信機能などを使い仮想的に行う事は可能。
スロット設定の一部	ワンショット送信設定および最低送信保証間隔の設定が SDK では非対応。	

※上記機能以外で SDK にて対応していない機能(ログトリガ、ログ解析機能等)については CAN アナライザ GUI にて対応している機能であり、MicroPecker の CAN 機能にて実現している機能ではありません。