

PEAK-SYSTEM TECHNIK GMBH

PCAN - PARAMETERS

PCAN-Basic Parameters Description

索引

はじめに	4
サポートされている PCAN-Parameter	5
パラメーターグループ	6
初期化済パラメーター	7
ハードウェアの識別	8
PCAN_CHANNEL_CONDITION	8
PCAN_CHANNEL_IDENTIFYING	9
PCAN_DEVICE_ID	11
PCAN_HARDWARE_NAME	12
PCAN_CONTROLLER_NUMBER	15
PCAN_IP_ADDRESS	16
PCAN_ATTACHED_CHANNELS	18
情報パラメーターの使用	20
PCAN_API_VERSION	20
PCAN_CHANNEL_VERSION	21
PCAN_CHANNEL_FEATURES	22
PCAN_BITRATE_INFO	23
PCAN_BITRATE_INFO_FD	24
PCAN_BUSSPEED_NOMINAL	25
PCAN_BUSSPEED_DATA	26
PCAN_LAN_SERVICE_STATUS	28
PCAN_FIRMWARE_VERSION	29
PCAN_ATTACHED_CHANNELS_COUNT	30
特別な動作の使用	32
PCAN_5VOLTS_POWER	32
PCAN_BUSOFF_AUTORESET	33
PCAN_LISTEN_ONLY	34
PCAN_BITRATE_ADAPTING	36
PCAN_INTERFRAME_DELAY	37
データフローの制御	39
PCAN_RECEIVE_EVENT	39
PCAN_MESSAGE_FILTER	41
PCAN_RECEIVE_STATUS	42
PCAN_ALLOW_STATUS_FRAMES	44
PCAN_ALLOW_RTR_FRAMES	45
PCAN_ALLOW_ERROR_FRAMES	46
PCAN_ACCEPTANCE_FILTER_11BIT	47
PCAN_ACCEPTANCE_FILTER_29BIT	49
ロギングパラメータの使用	52
PCAN_LOG_LOCATION	52

PCAN_LOG_STATUS.....	53
PCAN_LOG_CONFIGURE	55
PCAN_LOG_TEXT	56
トレースパラメータの使用	58
PCAN_TRACE_LOCATION.....	58
PCAN_TRACE_STATUS	59
PCAN_TRACE_SIZE.....	61
PCAN_TRACE_CONFIGURE	62
電子回路パラメータの使用	65
PCAN_IO_DIGITAL_CONFIGURATION	65
PCAN_IO_DIGITAL_VALUE.....	66
PCAN_IO_DIGITAL_SET	67
PCAN_IO_DIGITAL_CLEAR.....	68
PCAN_IO_ANALOG_VALUE	69
付録 A : レジストリのデバッグログ	70
Log Session のアクティブ化.....	70
Log Session の非アクティブ化	70
非常に重要な注意	70
付録 B : PCAN Trace Format 1.1	71
例	71
説明	71
付録 C : PCAN Trace Format 2.0	73
例	73
説明	73
付録 D : アクセプタンス・コードとアクセプタンス・マスクの計算.....	75
コード	75
マスク	75

はじめに

最近、PCAN-Basic 内で設定可能なパラメーターの数が増えてきました。特定のパラメーターをいつ使用する必要があるのか、どのように動作するのかを把握するのが難しい場合があります。さらに、あらかじめ初期化された動作をサポートするパラメーターもあります。それらのパラメーターはどのような意図で作られているのでしょうか？このドキュメントでは、このような質問などにお答えしていきます。

このドキュメントは、PCAN-Basic API バージョン 4.4.0 に基づいていることを考慮してください。API のバージョンを確認し、必要に応じて更新してください。

注意：このドキュメントに記載されているすべてのパラメーターが、PCAN-Basic で使用できるすべての Peak-Device に適用できるわけではありません。

API の普遍的な性質により、一部のパラメーターは、当社の製品ラインの特定のアイテムでのみ使用できます。パラメーターが参照する機能がサポートされているかどうかを確認するには、デバイスのユーザーマニュアルを参照してください。

API が最初のリリース以降の変更履歴は、下記の Web サイトで参照できます。

<https://www.peak-system.com/PCAN-Basic.126.0.html>

無料の PCAN-Basic (API) の新しいリリースなど、当社の製品に関する最新情報を簡単に入手したい場合は、RSS フィードを購読するか、サポート Web サイト (<https://www.peak-system.com/Support.55.0.html>) にアクセスしてください。

サポートされている PCAN-Parameter

PCAN-Basic は現在、関数 CAN_GetValue / CAN_SetValue を使用して read/configured できる 28 個のパラメーターをサポートしています。一部のパラメーターは **read-only** パラメーターであるため、すべてのパラメーターをコンフィグレーションできるわけではありません。以下に、パラメーターとそれに関連する値のリストがあります：

• <u>PCAN_DEVICE_ID</u>	1
• <u>PCAN_5VOLTS_POWER</u>	2
• <u>PCAN_RECEIVE_EVENT</u>	3
• <u>PCAN_MESSAGE_FILTER</u>	4
• <u>PCAN_API_VERSION</u>	5
• <u>PCAN_CHANNEL_VERSION</u>	6
• <u>PCAN_BUSOFF_AUTORESET</u>	7
• <u>PCAN_LISTEN_ONLY</u>	8
• <u>PCAN_LOG_LOCATION</u>	9
• <u>PCAN_LOG_STATUS</u>	10
• <u>PCAN_LOG_CONFIGURE</u>	11
• <u>PCAN_LOG_TEXT</u>	12
• <u>PCAN_CHANNEL_CONDITION</u>	13
• <u>PCAN_HARDWARE_NAME</u>	14
• <u>PCAN_RECEIVE_STATUS</u>	15
• <u>PCAN_CONTROLLER_NUMBER</u>	16
• <u>PCAN_TRACE_LOCATION</u>	17
• <u>PCAN_TRACE_STATUS</u>	18
• <u>PCAN_TRACE_SIZE</u>	19
• <u>PCAN_TRACE_CONFIGURE</u>	20
• <u>PCAN_CHANNEL_IDENTIFYING</u>	21
• <u>PCAN_CHANNEL_FEATURES</u>	22
• <u>PCAN_BITRATE_ADAPTING</u>	23
• <u>PCAN_BITRATE_INFO</u>	24
• <u>PCAN_BITRATE_INFO_FD</u>	25
• <u>PCAN_BUSSPEED_NOMINAL</u>	26
• <u>PCAN_BUSSPEED_DATA</u>	27
• <u>PCAN_IP_ADDRESS</u>	28
• <u>PCAN_LAN_SERVICE_STATUS</u>	29
• <u>PCAN_ALLOW_STATUS_FRAMES</u>	30
• <u>PCAN_ALLOW_RTR_FRAMES</u>	31
• <u>PCAN_ALLOW_ERROR_FRAMES</u>	32
• <u>PCAN_INTERFRAME_DELAY</u>	33
• <u>PCAN_ACCEPTANCE_FILTER_11BIT</u>	34
• <u>PCAN_ACCEPTANCE_FILTER_29BIT</u>	35
• <u>PCAN_IO_DIGITAL_CONFIGURATION</u>	36
• <u>PCAN_IO_DIGITAL_VALUE</u>	37
• <u>PCAN_IO_DIGITAL_SET</u>	38

• PCAN_IO_DIGITAL_CLEAR	39
• PCAN_IO_ANALOG_VALUE	40
• PCAN_FIRMWARE_VERSION	41
• PCAN_AVAILABLE_CHANNELS_COUNT	42
• PCAN_AVAILABLE_CHANNELS	43

パラメーターグループ

さまざまなパラメーターの目的を明確にするために、次のように 5 つのグループに配置されています：

「ハードウェア識別」のパラメーター：

- [PCAN_CHANNEL_CONDITION](#)
- [PCAN_DEVICE_ID](#)
- [PCAN_HARDWARE_NAME](#)
- [PCAN_CONTROLLER_NUMBER](#)
- [PCAN_CHANNEL_IDENTIFYING](#)
- [PCAN_IP_ADDRESS](#)
- [PCAN_AVAILABLE_CHANNELS](#)

「情報提供」を目的としたパラメーター：

- [PCAN_API_VERSION](#)
- [PCAN_CHANNEL_VERSION](#)
- [PCAN_CHANNEL_FEATURES](#)
- [PCAN_BITRATE_INFO](#)
- [PCAN_BITRATE_INFO_FD](#)
- [PCAN_BUSSPEED_NOMINAL](#)
- [PCAN_BUSSPEED_DATA](#)
- [PCAN_LAN_SERVICE_STATUS](#)
- [PCAN_FIRMWARE_VERSION](#)
- [PCAN_AVAILABLE_CHANNELS_COUNT](#)

「動作に影響を与える」ためのパラメーター：

- [PCAN_5VOLTS_POWER](#)
- [PCAN_BUSOFF_AUTORESET](#)
- [PCAN_LISTEN_ONLY](#)
- [PCAN_BITRATE_ADAPTING](#)
- [PCAN_INTERFRAME_DELAY](#)

「データリードとフロー制御」のパラメーター：

- [PCAN_RECEIVE_EVENT](#)
- [PCAN_MESSAGE_FILTER](#)
- [PCAN_RECEIVE_STATUS](#)
- [PCAN_ALLOW_STATUS_FRAMES](#)

- [PCAN_ALLOW_RTR_FRAMES](#)
- [PCAN_ALLOW_ERROR_FRAMES](#)
- [PCAN_ACCEPTANCE_FILTER_11BIT](#)
- [PCAN_ACCEPTANCE_FILTER_29BIT](#)

「ロギングとデバッグ」のパラメーター：

- [PCAN_LOG_LOCATION](#)
- [PCAN_LOG_STATUS](#)
- [PCAN_LOG_CONFIGURE](#)
- [PCAN_LOG_TEXT](#)

「CANデータ記録（トレース）」のパラメーター：

- [PCAN_LOG_LOCATION](#)
- [PCAN_LOG_STATUS](#)
- [PCAN_LOG_CONFIGURE](#)
- [PCAN_LOG_TEXT](#)

「電子回路（I/Oピン）」のパラメーター：

- [PCAN_IO_DIGITAL_CONFIGURATION](#)
- [PCAN_IO_DIGITAL_VALUE](#)
- [PCAN_IO_DIGITAL_SET](#)
- [PCAN_IO_DIGITAL_CLEAR](#)
- [PCAN_IO_ANALOG_VALUE](#)

初期化済パラメーター

PCAN-Basic API 内のパラメーター設定は、“Logging and Debugging” としてグループ化されたパラメーター（これらは特にチャンネルに関連付けられていません）を除き、チャンネルが正常に**初期化された後**に許可されます。しかし、チャンネルの初期化前に設定を行う必要がある場合もあります。以下のパラメーターは、チャンネルの**初期化前**に設定することができます：

- [PCAN_RECEIVE_STATUS](#)
- [PCAN_LISTEN_ONLY](#)
- [PCAN_BITRATE_ADAPTING](#)

ハードウェアの識別

まず、最初の識別は、使用する PCAN-Channel を選択する際に行われると考えてください。チャンネルの名前から、使用するバスが特定されます。

PCAN_USB BUS1

上記の名前は、接続する PCAN-Hardware、使用するバスの種類 (**USB**)、およびシステムに登録された **first (1)** ハードウェアであることを API に通知します。PCAN-Basic では、次のインターフェイスに接続できます：

- USB : Universal Serial Bus。最大 16 チャンネル。
- PCI : Peripheral Component Interconnect (ExpressCard ハードウェアを含む)。最大 16 チャンネル。
- PCC : PC カード (PCMCIA)、Personal Computer Memory Card。最大 2 チャンネル。
- LAN : Virtual PCAN-Gateway 接続。最大 16 チャンネル。
- DNG : Parallel port Dongle。最大 1 チャンネル。
- ISA : Industry Standard Architecture。最大 8 チャンネル。

ハードウェアをシステムに登録する方法は、コントローラードライバーとシステム自体に依存することに**注意してください**。同じ種類の複数のデバイス (USB など) がシステムにインストールされている場合、デフォルトでは、システムの再起動後に PCAN_USB BUS1 に接続しても同じハードウェアに接続できるとは限りません。

したがって、パラメーターは適切なハードウェアの検出に役立つように使用されます。以下のパラメーターは、接続する物理ハードウェアを識別するために使用されます。たとえば、複数のデバイスが接続される場合などです。

PCAN_CHANNEL_CONDITION

このパラメーターは、フラグ値を返すことにより、PCAN-Channel の使用状態を識別するために使用されます。たとえば、PCAN-Channel が使用可能な場合にのみ接続が可能である、ということになります。つまり、次のことを意味します：

- 有効： PCAN-Channel は、以下の “Supported by” セクションにリストされているもののいずれかである。
- 接続可能： PCAN-Channel が初期化されていないか、現在 PCAN-View アプリケーションで使用されています。

Availability

バージョン 1.0.0 以降で使用できます。バグ修正により、バージョン 1.0.4 以降は使いやすさが大幅に向上しました。このパラメーターの動作は、バージョン 4.0.0 で変更されました。

Supported by

PCAN-ISA (チャンネル PCAN_ISABUS1 から PCAN_ISABUS8)
 PCAN-DNG (チャンネル PCAN_DNGBUS1)
 PCAN-PCI (チャンネル PCAN_PCIBUS1 から PCANPCIBUS16)
 PCAN-USB (チャンネル PCAN_USB BUS1 から PCAN_USB BUS16)
 PCAN-PCC (チャンネル PCAN_PCCBUS1 から PCAN_PCCBUS2)
 PCAN-LAN (チャンネル PCAN_LANBUS1 から PCAN_LANBUS16)

Access Mode

このパラメーターは read のみ可能です。変更することはできません。

Possible Values

PCAN-Channel の条件は、次の定義済みの値のいずれかになります。

Defined Value	Description
PCAN_CHANNEL_UNAVAILABLE	チャンネルが無効であるか、存在しません
PCAN_CHANNEL_AVAILABLE	チャンネルを使用できます。
PCAN_CHANNEL_OCCUPIED	チャンネルはすでに初期化されています。
PCAN_CHANNEL_PCANVIEW	チャンネルは PCAN-View によって使用されていますが、初期化できます。

最後の値は、PCAN-Basic バージョン 4.0.0 で導入されたことに注意してください。

この値は、PCAN_CHANNEL_AVAILABLE と PCAN_CHANNEL_OCCUPIED の間の OR 演算です。このため、性 Availability（結果が PCAN_CHANNEL_AVAILABLE に等しい）のみをチェックするすべてのソフトウェアは、PCAN-View アプリケーションによって接続されているチャンネルを認識できません。

Default Value

適用されません。

Initialization Status

このパラメーターは PCAN-Channel の現在のステータスを確認するために使用されるため、関連性はありません。

When to Use

これは、特定の時点でシステムに登録されているチャンネルの Availability ステータスを知る必要がある場合に使用できます。

Application – Example of Use

PCAN-PCI デバイスに接続するテストアプリケーションを作成するとします。ユーザーがデータ送信に使用する PCAN-Channel を決定できるようにするには、使用可能なすべての PCAN-PCI チャンネルをリストする必要があります。このパラメーターを使用して、占有または使用不可のチャンネルをフィルタリングできます。

```
Repeat From PCAN_PCIBUS1 To PCAN_PCIBUS16
{
    Get the value CHANNEL_CONDITION on Channel-X (PCAN_PCIBUSX)
    If "CHANNEL_CONDITION Contains PCAN_CHANNEL_AVAILABLE" Then
    {
        Include Channel-X to the Available-Channels-List
    }
}
Show The PCAN-Channels available for connection are:
Print List Available-Channels-List
```

PCAN_CHANNEL_IDENTIFYING

このパラメーターは、使用中の USB ベースの PCAN-Channel を物理的に識別するために使用されます。識別は、USB デバイスのステータス LED を使用して行われます。現在、PEAK-System は 3 つの異なる世代の USB デバイスを提供しています。

- 第 1 世代 : PCAN-USB、PCAN-Hub
- 第 2 世代 : PCAN-USB Pro、PCAN-USB2
- 第 3 世代 : PCAN-USB Classic、PCAN-USB FD、PCAN-USB Pro FD、

使用するハードウェアに応じて、LED の点滅は色と点滅速度が異なります：

- 第 1 世代：点滅色は**赤色**で、点滅速度は約 300 ms です。
- 第 2 世代：点滅色は**オレンジ色**で、点滅速度は約 250 ms です。
- 第 3 世代：点滅色は**オレンジ色**で、点滅速度は約 250 ms です。

Availability

バージョン 1.3.0 以降で使用できます。

Supported by

PCAN-USB (チャネル PCAN_USBBUS1 から PCAN_USBBUS16)

Access Mode

このパラメーターは read / write です。設定して読み取ることができます。

Possible Values

このパラメーターは、アクティブ化 または 非アクティブ化できる識別に使用される手順を表します。

Defined Value	Description
PCAN_PARAMETER_OFF	識別手順は OFF に設定されています。
PCAN_PARAMETER_ON	識別手順は ON に設定されています。

一度にアクティブにできるチャネルは 1 つだけであることに**注意してください**。別のチャネルで識別手順を ON にするには、最初に前のチャネルを OFF にする必要があります。

Default Value

この識別手順のデフォルト状態は OFF です (PCAN_PARAMETER_OFF)。電源を入れた後、USB デバイスの LED は、明示的に OFF になるまで点滅し続けます。

Initialization Status

このパラメーターは、初期化された PCAN-Channel と初期化されていない PCAN-Channel の両方で使用できます。なお、この識別手順をアクティブにしても、識別中のデバイスで発生する通信には影響しないことに**注意してください**。

When to Use

これは、アプリケーションが複数の USB デバイスに接続でき、チャネルへの接続を確立する前など、決められた時間内にどの（物理）チャネルを使用する必要があるかが明確でない場合に使用できます。また、複数の USB デバイスと同時に長時間通信するアプリケーション（または複数の人が使用するアプリケーション）で、決められた時間内にチャネルが使用されているかどうかを確認するのに役立ちます。

Application – Example of Use

複数の USB デバイス（たとえば 5 つ）と通信するアプリケーションがあるとします。このアプリケーションは、各 PCAN-Channel を表すデバイスの順序が異なる可能性のあるコンピュータで実行されています（コンピュータは特定の期間内に自動的に再起動され、物理 CAN ネットワークは最終的にスワップされるなど）。これで、アプリケーションを使用していて、デバイスを切断する必要がありますが、どの PCAN-Channel がアプリケーションに関連付けられているかわからないため、他のチャネルを妨害したくありません。特定のチャネルの識別手順を ON にするだけの小さなアプリケーションを作成して、探しているデバイスがどれであることを確認できます。

```

Repeat From PCAN_USBBUS1 To PCAN_USBBUS5
{
    Set PCAN_CHANNEL_IDENTIFYING on Channel-X (PCAN_USBBUSX) to PCANON
    If "Identifying Procedure of Channel-X was activated" Then
    {
        Show Channel- X is being identified. Click OK to continue...
        Set PCAN_CHANNEL_IDENTIFYING on Channel-X (PCAN_USBBUSX) to OFF
    }
}

```

PCAN_DEVICE_ID

このパラメーターは、コンピューターに同時に接続されている同じ種類の 2 つ以上のデバイスを区別するために使用されます。デバイス識別子は、各デバイスのフラッシュメモリに保存されている永続的な値です。つまり、ハードウェアを切断した後、値が失われることはありません。

デバイスは同じ識別子を持つことができることに**注意してください**。使用されているデバイスが異なる識別子でコンフィグレーションされていることを保証するのはユーザーの責任であり、この値による差別化が機能します。

このパラメーターは、以前は PCAN_DEVICE_NUMBER と呼ばれていました。PCAN-Basic バージョン 4.4.0 以降、名前が PCAN_DEVICE_ID に変更されました。PCAN_DEVICE_NUMBER は、下位互換性の理由から引き続き存在しますが、**非推奨とされています**。ユーザーは代わりに PCAN_DEVICE_ID を使用する必要があります。

Availability

バージョン 1.0.0 以降で使用できます。 PCAN_DEVICE_NUMBER として。

バージョン 4.4.0 以降、初期化せずに読み取ることができます。

Supported by

PCAN-PCI (チャネル PCAN_PCIBUS1 から PCANPCIBUS16)

PCAN-USB (チャネル PCAN_USBBUS1 から PCAN_USBBUS16)

PCAN-LAN (チャネル PCAN_LANBUS1 から PCAN_LANBUS16)

Notes :

PCAN-PCI : FPGA ベースのデバイスのみ。4.2.0 以上のデバイスドライババージョンが必要です。

PCAN-LAN : ファームウェアバージョンが 2.8.2 以上のデバイスのみ。4.2.0 以上のデバイスドライババージョンが必要です。

Access Mode

このパラメーターは read / write です。設定して読み取ることができます。

Possible Values

PCAN-USB デバイスのファームウェアバージョンによると、この値の解像度は 1 バイト (範囲 [0 ... 255]) またはダブルワード (範囲 [0 ... 4294967295]) になります。

Default Value

このパラメーターが以前に設定されたことがない場合、値は使用される解像度で可能な最大値である 255 (FFh) または 429496729 (FFFFFFFFh) です。

Initialization Status

Get : 初期化されたまたは初期化されていない PCAN-Channel で読み取ることができます。

Set : 初期化された PCAN-Channel にのみ設定できます。

When to Use

同じシステムに同時に接続されている PCAN-USB デバイスを区別する必要がある場合に使用できます。

Application – Example of Use

1つの CAN-BUS からデータを読み取り、それを 2 番目の CAN-BUS に応答するアプリケーション(別名 Gateway アプリケーション)を作成するとします。このためには、各 CAN-BUS に接続された 1 つの PCAN-USB が必要です。両方の PCAN-USB のデバイス番号を設定して、書き込みに使用されるバス(たとえば、**number 1** for the “to write to” bus)と読み取りに使用されるバス(たとえば、**number 2** for the “to read from” bus)が分かるようにすることができます。このパラメーターを使用すると、両方のチャンネルが使用可能かどうか、送信に使用されているデバイスと書き込みに使用されているデバイスを知ることができます。

```
Repeat from PCAN_USBBUS1 To PCAN_USBBUS16
{
  Get "DEVICE_ID" from Channel-X
  If "DEVICE_ID" Equals 1" Then
  {
    Mark Channel-X as: WRITE_BUS
  }
  If "DEVICE_ID" Equals 2" Then
  {
    Mark Channel-X as: READ_BUS
  }
}
If "READ_BUS was found" AND "WRITE_BUS was found" Then
{
  Show "Both Channels were found. Starting..."
  Start working
}
Else
{
  Show "Error! Not all Channels found. Terminating..."
  Terminate
}
```

PCAN_HARDWARE_NAME

このパラメーターは、PCAN-Channel で表されるハードウェアから説明テキストを取得するために使用されます。このテキストにより、USB などの同じインターフェイスを使用するデバイスのモデルを認識できます。通常の PCAN-USB アダプターは "PCAN-USB" を返しますが、新しいデュアル CAN / CAN FD / LIN チャンネルアダプタは "PCAN-USB Pro FD" を返します。

Availability

バージョン 1.0.6 以降で使用できます。

バージョン 4.4.0 以降、初期化せずに読み取ることができます。

Supported by

PCAN-ISA (チャンネル PCAN_ISABUS1 から PCAN_ISABUS8)

PCAN-DNG (チャンネル PCAN_DNGBUS1)

PCAN-PCI (チャンネル PCAN_PCIBUS1 から PCANPCIBUS16)

PCAN-USB (チャンネル PCAN_USBBUS1 から PCAN_USBBUS16)

PCAN-PCC (チャンネル PCAN_PCCBUS1 から PCAN_PCCBUS2)

PCAN-LAN (チャンネル PCAN_LANBUS1 から PCAN_LANBUS16)

Access Mode

このパラメーターは read のみ可能です。変更することはできません。

Possible Values

値は、指定された PCAN-Channel によって指定されたハードウェアの名前を含む NULL 文字で終端された文字列です。この文字列の最大長は 32 バイトです (null-termination character)。

現在の PCAN-Channel で表されるハードウェアモデルに応じて、以下のテキストが返されます：

Hardware Name Value	Interface	Hardware Description
PEAK ISA-CAN	PCAN-ISA	PCAN-ISA, PCAN-PC/104
PEAK ISA-CAN SJA	PCAN-ISA	PCAN-ISA, PCAN-PC/104 with a SJA1000
PEAK Dongle-CAN	PCAN-DNG	PCAN-Dongle with an 82C200
PEAK Dongle-CAN EPP	PCAN-DNG	PCAN-Dongle with an 82C200, using EPP mode
PEAK Dongle-CAN SJA	PCAN-DNG	PCAN-Dongle with a SJA1000
PEAK Dongle-CAN SJA EPP	PCAN-DNG	PCAN-Dongle with a SJA1000, using EPP mode
PEAK Dongle-Pro	PCAN-DNG	PCAN-Dongle Pro
PEAK Dongle-Pro EPP	PCAN-DNG	PCAN-Dongle Pro in EPP mode
PCAN-PCI	PCAN-PCI	CAN Interface for PCI
PCAN-PCI Express	PCAN-PCI	CAN Interface for PCI Express
PCAN-PCI Express FD	PCAN-PCI	CAN and CAN FD Interface for PCI Express
PCAN-cPCI	PCAN-PCI	CAN Interface for CompactPCI
PCAN-MiniPCI	PCAN-PCI	CAN Interface for Mini PCI
PCAN-miniPCle	PCAN-PCI	CAN Interface for PCI Express Mini (PCle)
PCAN-miniPCle FD	PCAN-PCI	CAN and CAN FD Interface for PCI Express Mini (PCle)
PCAN-M.2	PCAN-PCI	CAN and CAN FD Interface for M.2 (PCle)
PCAN-Chip PCle FD	PCAN-PCI	Chip Solutions for the CAN FD Connection to PCI Express
PCAN-PCI/104-Plus	PCAN-PCI	CAN Interface for PC/104-Plus
PCAN-PCI/104-Plus Quad	PCAN-PCI	Four-Channel CAN Interface for PC/104-Plus
PCAN-PCI/104-Express	PCAN-PCI	CAN Interface for PCI/104-Express
PCAN-PC/104-Express FD	PCAN-PCI	CAN and CAN FD Interface for PCI/104-Express
PCAN-ExpressCard	PCAN-PCI	PCAN-ExpressCard
PCAN-ExpressCard 34	PCAN-PCI	PCAN-ExpressCard 34
PCAN-USB	PCAN-USB	PCAN-USB Adapter, PCAN-USB Hub
PCAN-USB FD	PCAN-USB	PCAN-USB FD Adapter, PCAN-USB
PCAN-USB Pro	PCAN-USB	PCAN-USB Pro dual CAN/LIN
PCAN-USB Pro FD	PCAN-USB	PCAN-USB Pro FD dual CAN/LIN FD
PCAN-USB Hub	PCAN-USB	All-in-one USB Adapter for communication through USB, CAN and RS-232

PCAN-USB X6	PCAN-USB	6-Channel CAN and CAN FD Interface for High-Speed USB 2.0
PCAN-Chip USB	PCAN-USB	Stamp Module for the Implementation of CAN FD to USB Connections
PCAN-PCCARD-CAN	PCAN-PCC	PCAN-PC Card
PCAN-Ethernet Gateway DR	PCAN-LAN	PCAN-Gateway wired for mounting on a DIN rail
PCAN-Wireless Gateway DR	PCAN-LAN	PCAN-Gateway wireless for mounting on a DIN rail
PCAN-Wireless Gateway	PCAN-LAN	PCAN-Gateway wireless with D-Sub connector
PCAN-Wireless Automotive Gateway	PCAN-LAN	PCAN-Gateway wireless with automotive connector

Default Value

適用されません。

Initialization Status

Get : 初期化されたまたは初期化されていない PCAN-Channel で読み取ることができます。

When to Use

同じインターフェイス（PCAN-PCI、PCAN-ExpressCard など）を使用して複数のハードウェアモデルを区別する必要がある場合に使用できます。

Application – Example of Use

次のようなシナリオを考えてみましょう。データ転送に通常の PCAN-USB デバイスを使用する診断アプリケーションを開発したいとします。このプログラムは、デフォルトで PCAN-USB Pro が接続されているコンピュータ上で実行されます。PCAN-USB Pro は、他のプログラムから使用されることを意図しており（ECU 制御、ゲートウェイ設定など）、したがって占有されるべきではありません。これは、システムに 3 つの PCAN-Channel（PCAN_USBBUS1 から PCAN_USBBUS3）が登録されることを意味します。診断ネットワークは常に PCAN-USB にプラグインされますので、アプリケーションは PCAN-USB Pro チャンネルの 1 つではなく、シングルチャンネルを接続するようにしなければなりません。このパラメーターを使用すると、どの PCAN-Channel が PCAN-USB を表し、どの PCAN-USB Pro を表すかを識別することができます。

```

Repeat From PCAN_USBBUS1 To PCAN_USBBUS3
{
    Initialize the current Channel-X (PCAN_USBBUSX)
    If "Channel-X was initialized" Then
    {
        Get the value HARDWARE_NAME
        If "HARDWARE_NAME Equals PCAN-USB" then
        {
            Mark Channel-X as: DEBUG_BUS
            Exit Repeat
        }
        Uninitialize Channel-X
    }
}
If "DEBUG_BUS was found" Then
{
    Show DEBUG_BUS found, connected, and ready to work...
    Start working
}
Else
{
    Show Error! Single PCAN-USB Channel was not found. Terminating...
    Terminate
}

```

PCAN_CONTROLLER_NUMBER

このパラメーターは、マルチチャネル CAN ハードウェア（PCAN-PCI、PCAN-USB Pro、PCAN-LAN など）の物理 CAN チャネルインデックスを識別するために使用されます。このインデックスはゼロベースであるため、デバイスの最初のチャネルは 0、2 番目のチャネルは 1 というように続きます。

Availability

バージョン 1.2.0 以降で使用できます。

バージョン 4.4.0 以降、初期化せずに読み取ることができます。

Supported by

PCAN-DNG（チャネル PCAN_DNGBUS1）

PCAN-ISA（チャネル PCAN_ISABUS1 から PCAN_ISABUS8）

PCAN-PCI（チャネル PCAN_PCIBUS1 から PCANPCIBUS16）

PCAN-USB（チャネル PCAN_USBBUS1 から PCAN_USBBUS16）

PCAN-PCC（チャネル PCAN_PCCBUS1 から PCAN_PCCBUS2）

PCAN-LAN（チャネル PCAN_LANBUS1 から PCAN_LANBUS16）

Access Mode

このパラメーターは read のみ可能です。変更することはできません。

Possible Values

[0…n-1] の範囲の数値。ここで、n は使用されているデバイス上の物理チャネルの数です。インデックス番号と、ハードウェアエチケット上の CAN チャネルの記述との対応関係は次のとおりです：

Channel Index	Channel Label
0	CAN 1
1	CAN 2
n-1	CAN n

Default Value

適用されません。

Initialization Status

Get：初期化されたまたは初期化されていない PCAN-Channel で読み取ることができます。

Set：初期化された PCAN-Channel にのみ設定できます。

When to Use

マルチチャネル PCAN デバイスのどの物理チャネルを接続する必要があるかを判断するために使用できます。

Application – Example of Use

簡単なケース：例えば、PCAN-USB デバイスの第 2 チャネルでのみ動作するアプリケーションを作成するとします。探しているチャネルが見つかるまで、使用可能な各 USB チャネルの PCAN_CONTROLLER_NUMBER を要求することができます。

```

Repeat From PCAN_USBBUS1 To PCAN_USBBUS16
{
    Initialize the current Channel-X (PCAN_USBBUSX)
    If "Channel-X was initialized" Then
    {
        Get the value CONTROLLER_NUMBER
        If "CONTROLLER_NUMBER Equals 1" then
        {
            Mark Channel-X as: CHANNEL_CAN2
            Exit Repeat
        }
        Uninitialize Channel-X
    }
}
If "CHANNEL_CAN2 was found" Then
{
    Show CAN-Channel Two was found.
    Start working
}
Else
{
    Show Error! CAN-Channel Two not found.
    Terminate
}

```

複雑なケース：特定の PCAN-USB Pro ハードウェアの 2 番目のチャネル（デバイス number 7 など）を使用した場合、一度に複数のマルチチャネルデバイスをコンピューターに接続されている可能性があります。パラメーター **PCAN_HARDWARE_NAME** を使用すると、接続されている PCAN-USB Pro を見つけることができます。パラメーター **PCAN_DEVICE_ID** を使用すると、適切なデバイス（number 7）を見つけることができます。最後に、**PCAN_CONTROLLER_NUMBER** を使用すると、使用する適切な CAN チャネルを見つけることができます：

```

Repeat From PCAN_USBBUS1 To PCAN_USBBUS16
{
    Initialize the current Channel-X (PCAN_USBBUSX)
    If "Channel-X was initialized" Then
    {
        Get the value HARDWARE_NAME
        If "HARDWARE_NAME Equals PCAN-USB Pro" then
        {
            Get the value DEVICE_ID
            If "DEVICE_ID Equals 7" Then
            {
                Get the value CONTROLLER_NUMBER
                If "CONTROLLER_NUMBER Equals 1" Then
                {
                    Mark Channel-X as: DEV7_CAN2
                    Exit Repeat
                }
            }
        }
        Uninitialize Channel-X
    }
}
If "DEV7_CAN2 was found" Then
{
    Show PCAN-USB Pro (Device Nr. 7 / CAN-Channel Two) was found.
    Start working
}
Else
{
    Show Error! PCAN-USB Pro (Device Nr. 7 / CAN-Channel Two) not found.
    Terminate
}

```

PCAN_IP_ADDRESS

このパラメーターは、PCAN-LAN タイプのハードウェアにのみ適用されます。これは、コンピューターに同時に接続されているこの種の 2 つ以上のハードウェアを区別するために使用されます。IP アドレスは、PCAN-Gateway デバイスに設定されたネットワークアドレスです。つまり、ネットワーク（LAN / WAN）を介して PCAN-Gateway

デバイスと通信するために使用されるアドレスです。

IP アドレスは、ネットワーク内で同じ IP アドレスを 2 つ持つことは許可されていないため、デバイスを効果的に識別します（アドレスの競合）。

Availability

バージョン 4.0.0 以降で使用できます。

Supported by

PCAN-LAN（チャネル PCAN_LANBUS1 から PCAN_LANBUS16）

Access Mode

このパラメーターは read のみ可能です。変更することはできません。

Possible Values

IP アドレスに使用されるフォーマットは IPv4 であるため、Possible Values は、[0~255] の範囲にある ‘.’ で区切られた 4 つの数値セクションを表す文字列です。IP アドレスの例は “192.168.0.1” です。

Default Value

適用されません。

Initialization Status

このパラメーターを使用する前に、PCAN-Channel を初期化する必要があります。

When to Use

特定の時間に同じシステムに接続されている PCAN-LAN デバイスを区別する必要がある場合や、リモートの PCAN-Gateway デバイスの詳細情報を得るために IP アドレスを使用する場合に使用することができます。

Application – Example of Use

接続に使用できる複数の PCAN-LAN チャネルがあり、それぞれが異なる PCAN-Gateway デバイスを表しているとします。リモートアドレス 192.168.1.95 の CAN データを監視したいのですが、PCAN-Basic にチャネルの Availability を要求すると、“PCAN_LANBUS1、PCAN_LANBUS2、PCAN_LANBUS3、…” などのチャネルのリストのみが返されます。各チャネルの IP アドレスを尋ねると、目的のデバイスを見つけるのに役立ちます。

```
Repeat From PCAN_USBBUS1 To PCAN_USBBUS16
{
    Initialize the current Channel-X (PCAN_USBBUSX)
    If "Channel-X was initialized" Then
    {
        Get the value CONTROLLER_NUMBER
        If "CONTROLLER_NUMBER Equals 1" then
        {
            Mark Channel-X as: CHANNEL_CAN2
            Exit Repeat
        }
        Uninitialize Channel-X
    }
}
If "CHANNEL_CAN2 was found" Then
{
    Show CAN-Channel Two was found.
    Start working
}
Else
{
    Show Error! CAN-Channel Two not found.
    Terminate
}
```

PCAN_ATTACHED_CHANNELS

このパラメーターは、現在の Availability に関係なく、システム上のすべての既存の PCAN-Channel に関する情報を 1 回の呼び出しで取得するために使用されます (**CHANNEL_CONDITION** を参照)。

このパラメーターは、別の **PCAN_ATTACHED_CHANNELS_COUNT** と密接に関連しています。これは、パラメーター **PCAN_ATTACHED_CHANNELS** を使用するとき、関数 **CAN_GetValue** に渡す必要がある、バッファのサイズ計算にとって重要な既存のチャンネルの数を返します。

このバッファのバイト単位のサイズは、**PCAN_ATTACHED_CHANNELS_COUNT** に構造体 **TPCANChannelInformation** のサイズを乗じた結果を使用して計算されます。

Python を使用する場合は、バッファのサイズを計算する必要はありません。Python で **PCANBasic.GetValue** を呼び出すと、結果としてタプルが返されるため、関数は、チャンネルの情報を格納および返すのに十分な大きさのバッファを内部的に定義しています。

利用可能なチャンネルごとに、次の情報が提供されます。

- *channel_handle*: API 呼び出しに使用される PCAN-Channel 識別ハンドルが含まれます (例: **PCAN_USBBUS1**、**PCAN_PCIBUS2** など)。
- *device_type*: PCAN-Channel が属するデバイスのタイプを示します (**PCAN_USB**、**PCAN_PCI** など)。
- *controller_number*: PCAN-Channel に関連付けられた物理 CAN チャンネルインデックス (ゼロベース) を示します。この値は、パラメーター [PCAN_CONTROLLER_NUMBER](#) を指定して **CAN_GetValue** を呼び出したときに返される値と同じです。
- *device_features*: PCAN-Channel に関連付けられた特別なプロパティに関する情報が含まれています。この値は、パラメーター [PCAN_CHANNEL_FEATURES](#) を指定して **CAN_GetValue** を呼び出したときに返される値と同じです。
- *device_name*: PCAN-Channel が属するデバイスからの説明テキストが含まれます。この値は、パラメーター [PCAN_HARDWARE_NAME](#) を指定して **CAN_GetValue** を呼び出したときに返される値と同じです。
- *device_id*: PCAN-Channel が属するデバイスのフラッシュメモリに保存されている識別値を表します。この値は、パラメーター [PCAN_DEVICE_ID](#) (以前は **PCAN_DEVICE_NUMBER** と呼ばれていました) を使用して **CAN_GetValue** を呼び出したときに返される値と同じです。
- *channel_condition*: PCAN-Channel の使用状態を表します。この値は、パラメーター [PCAN_CHANNEL_CONDITION](#) を指定して **CAN_GetValue** を呼び出したときに返される値と同じです。

Availability

バージョン 4.4.0 以降で使用できます。

Supported by

PCAN-NONEBUS：使用可能なチャンネルの数はどのチャンネルにも関連付けられていません。つまり、このクエリに特定のチャンネルを使用することはできません。

Access Mode

このパラメーターは read のみ可能です。変更することはできません。

Possible Values

戻り値は、TPCANChannelInformation 要素の配列です。

この配列には、パラメーター [PCAN ATTACHED CHANNELS COUNT](#) を指定して CAN_GetValue を呼び出したときに返される値と同じ数の要素が含まれています。

Default Value

適用されません。

Initialization Status

このパラメーターはチャンネルに依存しないため、関連性はありません。

When to Use

これは、1 回の関数呼び出しで、特定の時間に PC 内のすべての既存の PCAN-Channel を列挙するために使用できます。

Application – Example of Use

一般的に、アプリケーションは特定の作業を開始する前に、まずアプリケーションが持つ接続の可能性を示します。これは、接続可能なチャンネル、それらの名前、機能、および使用するデバイスの選択に役立つ可能性のあるその他の特性をシステムで検索することを意味します。パラメーター PCAN_ATTACHED_CHANNELS は、1 回の関数呼び出しでこのすべての情報を取得するために使用されます。

```

Get the PCAN_ATTACHED_CHANNELS_COUNT as ChannelsCount
Calculate the size of TPCANChannelInformation as StructSize
Create a Buffer with size ChannelsCount * StructSize

```

```
Get PCAN_ATTACHED_CHANNELS in Buffer
```

```

Repeat From i: 1 To ChannelsCount
{
    Show channels fields of Buffer[i]
}

```

情報パラメーターの使用

これらのパラメーターは、API 自体、およびハードウェア（デバイスドライバのバージョンなど）に関するバージョン情報を提供することを目的としています。使用されているバージョンに応じて異なる機能を使用できる場合とできない場合があるため、これは重要です。

PCAN-Basic ソフトウェアが特定のハードウェアで正しく動作することを確認するには、最初に（接続後）バージョンパラメータを確認することをお勧めします。こうすることで、ソフトウェアが開発時に機能していたのと同じようにユーザーに対して機能することを保証できます。

PCAN-Parameter と API ドライバー／ファームウェアバージョンとの間に依存関係が発生した場合は、PCAN-Basic のオンラインヘルプおよび PEAK-System の Web サイト（フォーラムなど）で通知およびコメントされることに**注意**してください。

PCAN_API_VERSION

このパラメーターは、API 実装バージョンを取得するために使用されます。

Availability

バージョン 1.0.0 以降で使用できます。

Supported by

すべてのチャネル：API 構造により、関数 `CAN_GetValue` を使用するときには PCAN-Parameter を取得するにはチャネル値が必要です。ただし、API バージョンは特定のチャネルに依存していないため、`PCAN_NONEBUS` を含む任意のチャネル値を使用できます。

Access Mode

このパラメーターは read のみ可能です。変更することはできません。

Possible Values

API バージョンの値は、"a.b.c.d" というフォーマットの文字列として表されます：

- a：メジャーバージョン番号を表します。
- b：マイナーバージョン番号を表します。
- c：リリースバージョン番号を表します。
- d：ビルド番号を表します。

4 つの値すべての最大サイズは 16 ビットであり、それぞれに 65535 の値を許可します。戻り値は、最大長が 24 バイトの NULL で終了する文字列です。いずれの場合も成功を保証するために、大きなバッファを使用することをお勧めします。

Default Value

適用されません。

Initialization Status

このパラメーターはチャネルに依存しないため、関連性はありません。

When to Use

使用する機能が使用可能かどうかを判断するためや、アプリケーションでの有益な出力として使用することもできます。

Application – Example of Use

例えば、アプリケーションから、使用されている API やライブラリのリストとそのバージョンを表示し、何か問題が発生した場合には、ユーザーがバージョン情報を確認できるようできます。

```
Get the value PCAN_API_VERSION on PCAN_NONEBUS
Show The PCAN-Basic version used is:
Print PCAN_API_VERSION
```

PCAN_CHANNEL_VERSION

このパラメーターは、使用されている PCAN デバイスの基になるデバイスドライバーに関する情報や著作権情報を取得するために使用されます。

Availability

バージョン 1.0.0 以降で使用できます。

Supported by

PCAN-ISA (チャンネル PCAN_ISABUS1 から PCAN_ISABUS8)
 PCAN-DNG (チャンネル PCAN_DNGBUS1)
 PCAN-PCI (チャンネル PCAN_PCIBUS1 から PCANPCIBUS16)
 PCAN-USB (チャンネル PCAN_USBBUS1 から PCAN_USBBUS16)
 PCAN-PCC (チャンネル PCAN_PCCBUS1 から PCAN_PCCBUS2)
 PCAN-LAN (チャンネル PCAN_LANBUS1 から PCAN_LANBUS16)

Access Mode

このパラメーターは read のみ可能です。変更することはできません。

Possible Values

ドライバーのバージョンと著作権に関する情報は、以下の情報を各行に含む複数行の文字列 (4 行) で表されます。

- 1) デバイスドライバー名とドライバーバージョン
- 2) ドライバーに実装されているアーキテクチャと対象プラットフォーム
- 3) 著作権の発行年
- 4) 会社名と本社所在地

このフォーマットは、デバイスドライバーバージョン 3.x 以降で使用できることに**注意してください**。戻り値は、最大長が 256 バイトの NULL で終了する文字列です (NULL 終了を含む)。いずれの場合も成功を保証するために、大きなバッファを使用することをお勧めします。

Default Value

適用されません。

Initialization Status

このパラメーターは特定のチャンネルに使用されるデバイスドライバーを参照するため、関係ありません。デバイスドライバーは Windows の起動時にロードされ、Windows のシャットダウン時に再度アンロードされます。

When to Use

このパラメーターは、アプリケーションで使用されているドライバーに関する情報出力として使用することができます。

Application – Example of Use

アプリケーションがハードウェアなしで配布されているため、ユーザーがテストしていないバージョンのデバイスを使用する可能性があります。このパラメーターを使用すると、実際にはソフトウェアではなく、間違っただライバーまたは古いドライバーの使用によって引き起こされたエラーを探すことによって、時間を失うことを回避できます。

```
If "an unexpected error occurred on Channel-USED" Then
{
  Get the value PCAN_API_VERSION on Channel-USED
  Get the value PCAN_CHANNEL_VERSION on Channel-USED
  Show Unknown error while working with Channel-USED
  Show Contact our support indicating the following data:
  Print PCAN_API_VERSION
  Print PCAN_CHANNEL_VERSION
  Terminate
}
```

PCAN_CHANNEL_FEATURES

このパラメーターは、使用されている PCAN デバイスの基になるデバイスドライバーに関する情報や著作権情報を取得するために使用されます。

Availability

バージョン 4.0.0 以降で使用できます。

Supported by

PCAN-ISA (チャンネル PCAN_ISABUS1 から PCAN_ISABUS8)

PCAN-DNG (チャンネル PCAN_DNGBUS1)

PCAN-PCI (チャンネル PCAN_PCIBUS1 から PCANPCIBUS16)

PCAN-USB (チャンネル PCAN_USBBUS1 から PCAN_USBBUS16)

PCAN-PCC (チャンネル PCAN_PCCBUS1 から PCAN_PCCBUS2)

PCAN-LAN (チャンネル PCAN_LANBUS1 から PCAN_LANBUS16)

Access Mode

このパラメーターは read のみ可能です。 変更することはできません。

Possible Values

特別な機能に関する情報は、“flag” 値として返されます。このドキュメントが作成された時点では、次のフラグのみが定義されています。

1) **FD_CAPABLE** : チャンネルがフレキシブルデータレート通信をサポートしていることを示します。

注 : 新しい CAN-FD 仕様を使用して通信するには、チャンネルが FD 対応であり、関数 CAN_InitializeFD で初期化する必要があります。初期化が成功すると、CAN 通信は関数 CAN_ReadFD および CAN_WriteFD によって実行されます。FD 対応チャンネルと FD 機能は、非 FD 通信にも使用できることに注意してください。つまり、標準 ISO 11898 (CAN 2.0 A / B) で指定されている CAN データです。

2) **DELAY_CAPABLE** : チャンネルが送信フレーム間の遅延の構成をマイクロ秒の解像度でサポートすることを示します。

注 : ファームウェアバージョンが 2.4.0 以上の FPGA ベースのデバイスのみがこの機能をサポートします。このドキュメントが作成された時点では、FPGA ベースの USB デバイスのみが遅延構成をサポートできました。

3) **IO_CAPABLE** : チャンネルで表されるハードウェアに I/O ピンが装備されており、それらを構成できることを示します。

注 : 現在、I/O パラメーターの使用をサポートしているのは PCAN-ChipUSB デバイスのみです。

Default Value

適用されません。

Initialization Status

このパラメーターは、初期化された PCAN-Channel と初期化されていない PCAN-Channel の両方で使用できます。

When to Use

機能に応じて、PCAN-Channel の初期化モードを決定するために使用できます。

Application – Example of Use

USB FD ハードウェアの使用をサポートするようにアプリケーションが更新されたとします。つまり、アプリケーションを FD として初期化するには、接続されている USB ハードウェアが FD 対応かどうかをユーザーに通知する機能が必要になります。このパラメーターを使用して、FD 対応ハードウェアのリストをユーザーに表示することができます：

```
Repeat From PCAN_USBBUS1 To PCAN_USBBUS16
{
    Get the value CHANNEL_FEATURES of the Channel-X (PCAN_USBBUSX)
    If "CHANNEL_FEATURES contain FD_CAPABLE" Then
    {
        Add Channel-X to: FD_LIST
    }
}
If "FD_LIST is not empty" Then
{
    Show FD Devices found:
    Print FD_LIST
}
Else
{
    Show No FD Channels available
}
```

PCAN_BITRATE_INFO

このパラメーターは、関数 CAN_Initialize を使用してチャンネルが初期化されたときに、使用されているビットレートに関する情報を取得するために使用されます。

Availability

バージョン 4.0.0 以降で使用できます。

バージョン 4.4.0 以降、初期化せずに読み取ることができます。

Supported by

PCAN-ISA (チャンネル PCAN_ISABUS1 から PCAN_ISABUS8)

PCAN-DNG (チャンネル PCAN_DNGBUS1)

PCAN-PCI (チャンネル PCAN_PCIBUS1 から PCANPCIBUS16)

PCAN-USB (チャンネル PCAN_USBBUS1 から PCAN_USBBUS16)

PCAN-PCC (チャンネル PCAN_PCCBUS1 から PCAN_PCCBUS2)

PCAN-LAN (チャンネル PCAN_LANBUS1 から PCAN_LANBUS16)

Access Mode

このパラメーターは read のみ可能です。変更することはできません。

Possible Values

この値の解像度は Word（範囲 [0 ... 65535]）で、CAN コントローラー SJA1000 のビットレートレジスタ（BTR0-BTR1）を表します。

Default Value

適用されません。

Initialization Status

Get：初期化済、または未初期化の PCAN-Channel で読み込まれます。

When to Use

使用されているビットレートを表す BTR0BTR1 値を取得するために使用できます。

Application – Example of Use

パラメーター `PCAN_BITRATE_ADAPTING` を使用して、チャンネル（`PCAN_USBBUS1`）を接続したとします。接続した後、使用されているビットレートが指定されたものとは異なることに気付きます。このパラメーターは、使用されているビットレートを通知するため、通信に使用されている実際のビットレート値についてユーザーに通知できます。

```
Set the value BITRATE_ADAPTING on Channel PCAN_USBBUS1 To ON
Initialize the Channel PCAN_USBBUS1 at 0x001C (500 kBit/s)
If "Initialize result equals PCAN_ERROR_OK" Then
{
    Show Channel successfully initialized with BTR0BTR1 0x001C
}
Else
{
    If "Initialize result equals PCAN_ERROR_WARNING" Then
    {
        Get the value PCAN_BITRATE_INFO as: NEW_BTR0BTR1
        Show Channel successfully initialized but with different BTR0BTR1:
        Print NEW_BTR0BTR1
    }
    Else
    {
        Show Error! Channel couldn't be initialized!
        Terminate
    }
}
}
```

PCAN_BITRATE_INFO_FD

このパラメーターは、関数 `CAN_InitializeFD` を使用してチャンネルが初期化されたときに、使用されているビットレートに関する情報を取得するために使用されます。

Availability

バージョン 4.0.0 以降で使用できます。

バージョン 4.4.0 以降、初期化せずに読み取ることができます。

Supported by

PCAN-ISA（チャンネル `PCAN_ISABUS1` から `PCAN_ISABUS8`）

PCAN-DNG（チャンネル `PCAN_DNGBUS1`）

PCAN-PCC（チャンネル `PCAN_PCCBUS1` から `PCAN_PCCBUS2`）

PCAN-PCI（チャンネル `PCAN_PCIBUS1` から `PCAN_PCIBUS16`）

PCAN-USB（チャンネル `PCAN_USBBUS1` から `PCAN_USBBUS16`）

PCAN-LAN（チャンネル `PCAN_LANBUS1` から `PCAN_LANBUS16`）

Access Mode

このパラメーターは read のみ可能です。変更することはできません。

Possible Values

設定 Possible Values は、FD 対応ハードウェアで使用される nominal ビットレートと Data ビットレート (PCAN-Basic のオンラインヘルプの TPCANBitrateFD の章を参照) を表す文字列です。

Default Value

適用されません。

Initialization Status

Get : 初期化されたまたは初期化されていない PCAN-Channel で読み取ることができます。v

When to Use

使用されているビットレートを表す TPCANBitrateFD 値を取得するために使用できます。

Application – Example of Use

パラメーター PCAN_BITRATE_ADAPTING を使用して、チャンネル (PCAN_USBBUS1) を接続したとします。接続後、使用されているビットレートが指定されたものとは異なることに気が付きます。このパラメーターを問い合わせることで、使用されているビットレートを知ることができるので、通信に使用されている実際のビットレート値をユーザーに通知できます。

```
Set the value BITRATE_ADAPTING on Channel PCAN_USBBUS1 To ON
Initialize the Channel PCAN_USBBUS1 as FD using this bit rate:
f_clock=20000000, nom_brp=5, nom_tseg1=2, nom_tseg2=1, nom_sjw=1
(1 Mbit/s for both, nominal and data bit rates)
If "InitializeFD result equals PCAN_ERROR_OK" Then
{
    Show Channel successfully initialized with bit rate parameters:
    Print f_clock=20000000, nom_brp=5, nom_tseg1=2, nom_tseg2=1, nom_sjw=1
}
Else
{
    If "InitializeFD result equals PCAN_ERROR_WARNING" Then
    {
        Get the value PCAN_BITRATE_INFO_FD as: NEW_BITRATE_FD
        Show Channel successfully initialized but with different bit rate:
        Print NEW_BITRATE_FD
    }
    Else
    {
        Show Error! Channel couldn't be initialized!
        Terminate
    }
}
}
```

PCAN_BUSSPEED_NOMINAL

このパラメーターは、現在使用されている nominal CAN Bus speed に関する情報を Bit / s で取得するために使用されます。

Availability

バージョン 4.0.0 以降で使用できます。

Supported by

PCAN-ISA (チャンネル PCAN_ISABUS1 から PCAN_ISABUS8)

PCAN-DNG (チャンネル PCAN_DNGBUS1)

PCAN-PCC (チャンネル PCAN_PCCBUS1 から PCAN_PCCBUS2)

PCAN-PCI (チャンネル PCAN_PCIBUS1 から PCANPCIBUS16)

PCAN-USB (チャンネル PCAN_USBBUS1 から PCAN_USBBUS16)

PCAN-LAN (チャンネル PCAN_LANBUS1 から PCAN_LANBUS16)

Access Mode

このパラメーターは read のみ可能です。変更することはできません。

Possible Values

この値の解像度はダブルワード (範囲 [0 ... 4294967295]) です。

Default Value

適用されません。

Initialization Status

このパラメーターを使用する前に、PCAN-Channel を初期化する必要があります。

When to Use

これは、どのようなユーザーにもわかりやすく、かつ迅速に理解できる、親しみやすいビットレート値を示すために使用できます。

Application – Example of Use

パラメーター PCAN_BITRATE_ADAPTING を使用して、チャンネル (PCAN_USBBUS1) を接続したとします。接続後、使用されているビットレートが指定されたものとは異なることに気がきます。設定されたビットレートは不明な BTR0~BTR1 値に基づいている可能性があるため、これを自分でデコードできない可能性があります。このパラメーターを使用すると、この “decoded” 値を求めることができるため、コード化されたビットレート値 (ビットなど) の代わりに、使用されている bits/s、Kbits/s、Mbit/s などのビットレート単位で表示することができます (ビットレートレジスタのように) :

```
Set the value BITRATE_ADAPTING on Channel PCAN_USBBUS1 To ON
Initialize the Channel PCAN_USBBUS1 at 0x001C (500 kBit/s)
If "Initialize result equals PCAN_ERROR_OK" Then
{
    Show Channel successfully initialized at 500 kBit/s
}
Else
{
    If "Initialize result equals PCAN_ERROR_WARNING" Then
    {
        Get the value PCAN_BUSSPEED_NOMINAL as: BUSSPEED
        Show Channel successfully initialized but with different bit rate:
        Print (BUSSPEED / 1000) kBit/s
    }
    Else
    {
        Show Error! Channel couldn't be initialized!
        Terminate
    }
}
}
```

PCAN_BUSSPEED_DATA

このパラメーターは、現在使用されている CAN data speed (Bit rate Switch) に関する情報を bit / s で取得するために使用されます。

Availability

バージョン 4.0.0 以降で使用できます。

Supported by

PCAN-ISA (チャンネル PCAN_ISABUS1 から PCAN_ISABUS8)
 PCAN-DNG (チャンネル PCAN_DNGBUS1)
 PCAN-PCC (チャンネル PCAN_PCCBUS1 から PCAN_PCCBUS2)
 PCAN-PCI (チャンネル PCAN_PCIBUS1 から PCANPCIBUS16)
 PCAN-USB (チャンネル PCAN_USBBUS1 から PCAN_USBBUS16)
 PCAN-LAN (チャンネル PCAN_LANBUS1 から PCAN_LANBUS16)

Access Mode

このパラメーターは read のみ可能です。変更することはできません。

Possible Values

この値の解像度はダブルワード (範囲[0 ... 4294967295]) です。

Default Value

適用されません。

Initialization Status

このパラメーターを使用する前に、PCAN-Channel を初期化する必要があります。

When to Use

これは、どのようなユーザーにもわかりやすく、かつ迅速に理解できる、親しみやすいビットレート値を示すために使用できます。

Application – Example of Use

パラメーター PCAN_BITRATE_ADAPTING を使用して、チャンネル (PCAN_USBBUS1) を接続したとします。接続後、使用されているビットレートが指定されたものとは異なることに気がきます。設定されたビットレートは不明なビットレート値に基づいている可能性があるため、これを自分でデコードできない可能性があります。このパラメーターを使用すると、この “decoded” (デコードされた) 値を求めることができるため、直感的ではないコード化されたビットレート値の代わりに、ビット/秒、K ビット/秒、M ビット/秒などで使用されるビットレート (クロック周波数、サンプルジャンプなど) を表示できます :

```
Set the value BITRATE_ADAPTING on Channel PCAN_USBBUS1 To ON
Initialize the Channel PCAN_USBBUS1 as FD using this bit rate:
f_clock=20000000, nom_brp=5, nom_tseg1=2, nom_tseg2=1, nom_sjw=1
(1 Mbit/s for both, nominal and data bit rates)
If “InitializeFD result equals PCAN_ERROR_OK” Then
{
  Show Channel successfully initialized at 1 Mbit/s | 1 Mbit/s:
}
Else
{
  If “InitializeFD result equals PCAN_ERROR_WARNING” Then
  {
    Get the value PCAN_BUSSPEED_NOMINAL as: BUSSPEED_N
    Get the value PCAN_BUSSPEED_DATA as: BUSSPEED_D
    Show Channel successfully initialized but with different bit rate:
    Print (BUSSPEED_N / 1000) kBit/s | (BUSSPEED_D / 1000) kBit/s
  }
  Else
  {
    Show Error! Channel couldn't be initialized!
    Terminate
  }
}
```

PCAN_LAN_SERVICE_STATUS

このパラメーターは、Virtual PCAN-Gateway ソリューションの一部であるシステムサービスの実行ステータスを取得するために使用されます。このサービスは、デバイスドライバ PCAN-LAN と連携して動作します。どちらも、Windows システムで PCAN 環境を使用することにより、PCAN-LAN ハードウェア (PCAN-Gateway Ethernet / Wireless) との相互作用を可能にします。

Availability

バージョン 4.1.0 以降で使用できます。

Supported by

PCAN_NONEBUS : サービスのステータスはどのチャンネル接続にも関連付けられていません。つまり、このクエリに特定のチャンネルを使用することはできません。

Access Mode

このパラメーターは read のみ可能です。変更することはできません。

Possible Values

Virtual PCAN-Gateway サービスのステータスは、次の定義済みの値のいずれかになります：

Defined Value	Description
SERVICE_STATUS_STOPPED	サービスが'running'されていない、つまり停止しているか、'running'とは異なる状態になっています。
SERVICE_STATUS_RUNNING	サービスが'running'されています。

Default Value

適用されません。

Initialization Status

このパラメーターはチャンネルに依存しないため、関連性はありません。

When to Use

Virtual PCAN-Gateway の通信が機能していることを確認するために使用することができます。

Application – Example of Use

コンピュータで検出された最初の PCAN-LAN チャンネルを常に接続するアプリケーションを作成し、Windows の起動時にアプリケーションが自動的に起動するとします。アプリケーションは、PCAN-Gateway デバイスとサービスの間の接続を確立するのに十分な時間である 20 秒間、チャンネルの接続を試みます (サービスとデバイスの両方がすでに初期化されています)。接続が発生しない場合、アプリケーションは終了します。ここで、何らかの理由で、サービスが 30 秒の遅延で開始されたとしましょう。この場合、アプリケーションは 20 秒後に終了するため、チャンネルに接続できなくなります。この場合、サービスが実行されているかどうかをチェックすることができるので、サービスの開始後にのみタイムアウトのチェックを開始します。このようにして、アプリケーションは実際には 30 秒 (またはサービスの初期化に必要な時間) 待機し、その時点で、サービスが 'running' になると、最大 20 秒が経過するまで、または接続が行われるまで接続のチェックを開始します。

```

..... At_Application_Start
{
    Repeat From 1 To 20 // Check time about twenty seconds
    {
        Get the value PCAN_LAN_SERVICE_STATUS as: LAN_STATUS
        If "LAN_STATUS equals SERVICE_STATUS_RUNNING" Then
        {
            Set "Initialize (PCAN_LANBUS1)" as: CONNECTION_STATUS
            Exit Repeat
        }
    }
    If "CONNECTION_STATUS equals PCAN_ERROR_OK" Then
    {
        Show Channel connected! Starting to work...
    }
    Else
    {
        If "LAN_STATUS equals SERVICE_STATUS_STOPPED" Then
        {
            Show Error! PCAN-LAN Service is not running!
        }
        Else
        {
            Show Error! Channel couldn't be initialized!
        }
        Terminate
    }
}
}

```

PCAN_FIRMWARE_VERSION

このパラメーターは、PCAN-Channel に関連付けられている PCAN デバイスのファームウェアバージョンを取得するために使用します。

Availability

バージョン 4.4.0 以降で使用できます。

Supported by

PCAN-ISA (チャネル PCAN_ISABUS1 から PCAN_ISABUS8)

PCAN-DNG (チャネル PCAN_DNGBUS1)。

PCAN-PCI (チャネル PCAN_PCIBUS1 から PCANPCIBUS16)

PCAN-USB (チャネル PCAN_USBBUS1 から PCAN_USBBUS16)

PCAN-PCC (チャネル PCAN_PCCBUS1 から PCAN_PCCBUS2)

PCAN-LAN (チャネル PCAN_LANBUS1 から PCAN_LANBUS16)

Access Mode

このパラメーターは read のみ可能です。変更することはできません。

Possible Values

API バージョン値は、"a.b.c" というフォーマットの文字列として表されます：

a : メジャーバージョン番号を表します。

b : マイナーバージョン番号を表します。

c : リリースバージョン番号を表します。

3つの値の最大サイズは 16 ビットであり、それぞれに 65535 の値を許可します。戻り値は、最大長が 18 バイトの NULL 文字で終端する文字列です。いずれの場合も成功を保証するために、大きなバッファを使用することをお勧めします。

Default Value

適用されません。

Initialization Status

このパラメーターを使用する前に、PCAN-Channel を初期化する必要があります。

When to Use

これは、デバイスが最新であるかどうかを判断するために使用することもできますし、アプリケーションの情報出力として使用することもできます。

Application – Example of Use

アプリケーションで使用されている PCAN-Hardware に関する情報を表示して、物理的な問題が発生した場合にユーザーがバージョン情報を得ることができるようにしたいとします。これにより、ユーザーの問題の原因が古いファームウェアに起因するものであるかどうかを確認することができます。

```
Get the value PCAN_FIRMWARE_VERSION on Channel-USED
Show The device is using the Firmware version:
Print PCAN_FIRMWARE_VERSION
```

PCAN_ATTACHED_CHANNELS_COUNT

このパラメーターは、現在の Availability にかかわらず、システム上のすべての既存の PCAN-Channel に関する情報を 1 回の呼び出しで取得するために使用されます ([CHANNEL_CONDITION](#) を参照)。

このパラメーターは、別の [PCAN_ATTACHED_CHANNELS](#) と非常に密接に関連しています。チャンネルデータを含む、“TPCANChannelInformation” タイプの構造のバッファを返します。このバッファのバイト単位のサイズは、PCAN_ATTACHED_CHANNELS_COUNT の結果に構造体 TPCANChannelInformation のサイズを掛けたものを使用して計算されます。

Availability

バージョン 4.4.0 以降で使用できます。

Supported by

PCAN-NONEBUS：使用可能なチャンネルの数はどのチャンネルにも関連付けられていません。つまり、このクエリに特定のチャンネルを使用することはできません。

Access Mode

このパラメーターは read のみ可能です。変更することはできません。

Possible Values

[0…n]の範囲の数値で、n はデバイスごとにサポートされる最大チャンネルの合計です。このドキュメントを書く時点では、最大 59 のチャンネルを同時に処理できます。1 つの PCAN-Dongle、2 つの PCAN-PCC、8 つの PCAN-ISA、16 の PCAN-PCI、16 の PCAN-USB、および 16 の PCAN-LAN デバイスです。

Default Value

適用されません。

Initialization Status

このパラメーターは、特定のチャンネルに依存しないため、関連性はありません。

When to Use

システム上に現在存在するチャンネルがあるかどうかを判断したり、必要に応じてそれらのチャンネルに関する情報を取得するためのバッファのサイズを計算するために使用できます。

Application – Example of Use

一般的に、アプリケーションは、特定の作業を開始する前に、まずアプリケーションが持つ接続の可能性を示します。これは、接続可能なチャンネル、それらの名前、機能、および使用するデバイスの選択に役立つ可能性のあるその他の特性をシステムで検索することを意味します。使用するプログラミング言語に応じて、これらの既存のチャンネルに関する情報を格納するのに十分な大きさのバッファを生成する必要があります。このパラメーターは、そのバッファのサイズを計算するために使用されます。

```
Get the PCAN_AVAILABLE_CHANNELS_COUNT as ChannelsCount  
Calculate the size of TPCANChannelInformation as StructSize  
Create a Buffer with size ChannelsCount * StructSize
```

```
Get PCAN_AVAILABLE_CHANNELS in Buffer
```

```
Repeat From i: 1 To ChannelsCount
```

```
{  
    Show channels fields of Buffer[i]  
}
```

特別な動作の使用

これらのパラメーターは、使用されているデバイスでいくつかのモードをアクティブにして、それらのデバイスが例外的な方法で反応または動作するようにすることを目的としています。

すべてのモードがすべての種類のデバイスでサポートされているわけではないことに**注意してください**。

PCAN_5VOLTS_POWER

このパラメーターは、PCAN-Device の D-Sub コネクタの外部 5V を切り替えるために使用されます。これは、外部バスコンバータモジュールをカード (AU5790 / TJA1054) に接続するときに役立ちます。

Availability

バージョン 1.0.0 以降で使用できます。

バージョン 4.4.0 以降、初期化せずに読み取ることができます。

Supported by

PCAN-PCC (チャンネル PCAN_PCCBUS1 から PCAN_PCCBUS2)

PCAN-USB (チャンネル PCAN_USBBUS1 から PCAN_USBBUS16)

Notes :

PCAN-USB : “PCAN-USB Hub” タイプのデバイスのみがこのパラメーターをサポートできます。

Access Mode

このパラメーターは read / write です。設定して読み取ることができます。

Possible Values

このパラメーターは、アクティブ化 または、非アクティブ化 できる追加の電圧を表します。

Defined Value	Description
PCAN_PARAMETER_OFF	D-Sub コネクタの外部 5V は inactive (非アクティブ) です。
PCAN_PARAMETER_ON	D-Sub コネクタの外部 5V が active (アクティブ) です。

Default Value

追加電圧のデフォルト状態は inactive (非アクティブ) です (PCAN_PARAMETER_OFF)。active (アクティブ) 化した後、追加の 5V は、明示的に非アクティブ化されるか、デバイスが再初期化されるまで (プラグアウトして再度プラグインするか、PC で再起動するまで)、D-Sub に留まります。

Initialization Status

Get : 初期化されたまたは初期化されていない PCAN-Channel で読み取ることができます。

Set : 初期化された PCAN-Channel にのみ設定できます。

When to Use

外部バスコンバータモジュールをデバイスに接続して、デバイスにも電源を供給する場合に使用します

Application – Example of Use

アプリケーションが PC カードのチャンネルを使用して Single-Wired (シングルワイヤード) CAN ネットワークに接続されているとしましょう。バスコンバーター (High-speed から Single-Wire CAN など) も使用されるチャンネルに接続されます。これは、ソフトウェアまたは診断データを転送する特別な場合にのみ使用されます。アダプターを機能させるには、PCAN_5VOLTS_POWER を使用する必要があります。

```
Set the value PCAN_5VOLT_POWER of Channel-USED To ON
If "PCAN_5VOLT_POWER equals PCAN_PARAMETER_ON" Then
{
    Show Channel-USED has now 5V power in D-Sub
    Do needed work/communication
    Set the value PCAN_5VOLT_POWER of Channel-USED To OFF
    If "PCAN_5VOLT_POWER equals PCAN_PARAMETER_OFF" Then
    {
        Show The 5V power on Channel-USED is now deactivated
    }
    Else
    {
        Show Warning: the 5V power couldn't be disabled
        Show ....Risk of damage if short circuit....
    }
}
Else
{
    Show 5V power couldn't be enabled
}
```

PCAN_BUSOFF_AUTORESET

このパラメーターは、バスオフ状態が検出されたときに PCAN-Channel の CAN コントローラーを自動的にリセットするように PCAN ドライバーに指示します。

Availability

バージョン 1.0.0 以降で使用できます。

Supported by

PCAN-ISA (チャンネル PCAN_ISABUS1 から PCAN_ISABUS8)

PCAN-DNG (チャンネル PCAN_DNGBUS1)

PCAN-PCI (チャンネル PCAN_PCIBUS1 から PCANPCIBUS16)

PCAN-USB (チャンネル PCAN_USBBUS1 から PCAN_USBBUS16)

PCAN-PCC (チャンネル PCAN_PCCBUS1 から PCAN_PCCBUS2)

PCAN-LAN (チャンネル PCAN_LANBUS1 から PCAN_LANBUS16)

Access Mode

このパラメーターは read / write です。設定して読み取ることができます。

Possible Values

このパラメーターは、アクティブ化 または 非アクティブ化 できます。

Defined Value	Description
PCAN_PARAMETER_OFF	ハードウェアの自動リセットはオフです。
PCAN_PARAMETER_ON	ハードウェアの自動リセットはオンです。

Default Value

バスオフ時の自動リセットのデフォルト状態は非アクティブです (PCAN_PARAMETER_OFF)。それをアクティブにした後、自動リセットは、明示的に非アクティブになるか、チャンネルが切断されるまで（たとえば、関数 CAN_Uninitialize を使用して）アクティブのままになります。

Initialization Status

このパラメーターを使用する前に、PCAN-Channel を初期化する必要があります。

When to Use

例えば、アプリケーションが無人で動作し、バスオフエラーが発生することが分かっている場合など、バスオフ状態になった後にデバイスを手動でリセットすることを避けるために使用できます。

Application – Example of Use

アプリケーションが車の電子制御ユニット (ECU : Electronic Control Unit) でいくつかの診断を実行していて、この ECU がバッテリー駆動 (車のスイッチのオンとオフ) であるとします。アプリケーションが同じ CAN ネットワークと通信し、ECU のオンとオフを切り替えると、PCAN-Channel (ハードウェア、CAN コントローラー) がオフ状態になる可能性があります。オフ状態が解除されるまで通信できません。車のスイッチをオンまたはオフにするたびにアプリケーション/PCAN-Channel を手動でリセットする必要をなくすために、このパラメーターを使用してこれを自動的に行うことができます：

```
Set the value PCAN_BUSOFF_AUTORESET of Channel-USED To ON
If "PCAN_BUSOFF_AUTORESET equals PCAN_PARAMETER_ON" Then
{
    Show Channel-USED will reset itself automatically on Bus-OFF
    Do needed work/communication
}
Else
{
    Show Auto-reset on Bus-OFF couldn't be enabled
}
```

PCAN_LISTEN_ONLY

このパラメーターを使用すると、ユーザーは、PCAN-Channel で表される CAN デバイスを Listen-Only に設定できます。このモードが設定されている場合、CAN コントローラーはアクティブイベント (CAN メッセージの送信など) には参加せず、パッシブモード (CAN モニター) に留まり、PCAN-Channel で使用される CAN バス上のトラフィックを分析できます。フィリップスのデータシート "SJA1000 Stand-alone CAN controller" も参照してください。

このパラメーターは、いわゆる **“pre-initialized”** パラメーターです。つまり、パラメーターをできるだけ早く activate / deactivate するために、PCAN-Channel が初期化される前に設定できます。これにより、機密性の高い操作で発生する可能性のある問題を回避できます。

Availability

バージョン 1.0.0 以降で使用できます。

Supported by

PCAN-ISA (チャンネル PCAN_ISABUS1 から PCAN_ISABUS8)

PCAN-DNG (チャンネル PCAN_DNGBUS1)

PCAN-PCI (チャンネル PCAN_PCIBUS1 から PCANPCIBUS16)

PCAN-USB (チャンネル PCAN_USBBUS1 から PCAN_USBBUS16)

PCAN-PCC (チャンネル PCAN_PCCBUS1 から PCAN_PCCBUS2)

PCAN-LAN (チャンネル PCAN_LANBUS1 から PCAN_LANBUS16)

Access Mode

このパラメーターは read / write です。設定して読み取ることができます。

Possible Values

このパラメーターは、アクティブ化 または 非アクティブ化 できます。

Defined Value	Description
PCAN_PARAMETER_OFF	Listen-only モードはオフです。
PCAN_PARAMETER_ON	Listen-only モードはオンです。

Default Value

Listen-Only モードのデフォルト状態は無効になっています (PCAN_PARAMETER_OFF)。アクティブ化した後、Listen-Only モードは、明示的に非アクティブ化されるか、チャンネルが切断される (たとえば、関数 CAN_Uninitialize を使用するなど) までアクティブのままになります。

Initialization Status

このパラメーターは、初期化されたチャンネルでも、初期化されていないチャンネルでも使用できます。

When to Use

これは、アプリケーションが CAN ネットワーク内で転送されるデータを、妨害することなくパッシブに検査したい場合に使用できます。

Application – Example of Use

例えば、4 つの異なるビットレートのみが使用されている環境でアプリケーションを動作させる必要があるとします。4 種類のビットレートは既知であるため、接続時に CAN ネットワークで現在設定されているビットレートを自動検出する機能を提供したいとします。このパラメーターを使用すると、異なるビットレートを使用しているネットワークにパッシブに接続することができ、誤ったビットレートで接続するとエラーが発生します。この方法では、アプリケーションが使用されているビットレートを認識し、この手順が実行されている間は通信に影響を与えません：

```

Mark BAUDRATE_FOUND as: Empty
Repeat From Baudrate1 To BaudRate4
{
  Set PCAN_LISTEN_ONLY on Channel-USED to ON
  If "PCAN_LISTEN_ONLY equals PCAN_PARAMETER_ON" Then
  {
    Initialize the Channel-USED with BaudRate-X
    If "Channel-USED was initialized" Then
    {
      If "Any message received" Then
      {
        Mark BAUDRATE_FOUND as: BaudRate-X
        Uninitialize the Channel-USED
        Exit Repeat
      }
      Uninitialize the Channel-USED
    }
    Else
    {
      Show Channel cannot be initialized. Terminating...
      Terminate
    }
  }
}
If "BAUDRATE_FOUND is not Empty" then
{
  Show Baud rate found:
  Print BaudRateFound
}
Else
{
  Show Baud rate couldn't be found.
}

```

PCAN_BITRATE_ADAPTING

このパラメーターは、使用されているビットレートが不明な場合に、ユーザーがアクティブな PCAN-Channel に接続することを可能にします。このモードが設定されている場合、PCAN-Basic は最初に初期化プロセスでパラメーターとして指定されたビットレートの使用を最初に試みます。チャンネルに異なるビットレートが設定されている場合、新しい接続は設定されたビットレートを使用し、初期化関数は使用されたビットレートが指定されたビットレートと異なることを示す警告値を返します。

このパラメーターは、いわゆる **“pre-initialized only”** のパラメーターで、PCAN-Channel が初期化される前にしか設定できないことを意味しています。

Availability

バージョン 4.0.0 以降で使用できます。

Supported by

PCAN-ISA（チャンネル PCAN_ISABUS1 から PCAN_ISABUS8）

PCAN-DNG（チャンネル PCAN_DNGBUS1）

PCAN-PCI（チャンネル PCAN_PCIBUS1 から PCANPCIBUS16）

PCAN-USB（チャンネル PCAN_USBBUS1 から PCAN_USBBUS16）

PCAN-PCC（チャンネル PCAN_PCCBUS1 から PCAN_PCCBUS2）

PCAN-LAN（チャンネル PCAN_LANBUS1 から PCAN_LANBUS16）

Access Mode

このパラメーターは read / write です。設定して読み取ることができます。

Possible Values

このパラメーターは、アクティブ化 または 非アクティブ化 できます。

Defined Value	Description
PCAN_PARAMETER_OFF	Bitrate-Adapting 機能はオフです。
PCAN_PARAMETER_ON	Bitrate-Adapting 機能はオンです。

Default Value

Bitrate-Adapting モードのデフォルト状態は非アクティブ化されています（PCAN_PARAMETER_OFF）。このパラメーターは、初期化時にのみ有効です。チャンネルをアクティブにした後で設定することはできません。initialize / InitializeFD 関数を呼び出すと、このパラメーターは Default Value に戻ります。

Initialization Status

このパラメーターは、初期化されていないチャンネルでのみ使用できます。

When to Use

これは、アプリケーションがチャンネルに接続したいときに、そのチャンネルが異なるまたは未知のビットレートで使用されている（PCAN-View）か、どうかにかかわらず、使用することができます。

Application – Example of Use

アプリケーションがリモート LAN チャンネル (PCAN-Gateway virtual channels) で動作し、それらの1つ、いくつか、またはすべてで設定されたビットレートがわからないとします。LAN チャンネルのビットレートは PCAN-Basic API を使用して変更することができないため、間違ったビットレートを使用すると、初期化は失敗します。initialize を呼び出す前にこのパラメーターをアクティブにすると、アプリケーションは渡されたビットレートをテストし、一致しない場合は無視することができます。このようにして、初期化は常に成功します。

```
Set the value BITRATE_ADAPTING on Channel PCAN_LANBUS1 To ON
Initialize the Channel PCAN_LANBUS1 at any bit rate, e.g. 0x001C (500 kBit/s)
If "Initialize result equals PCAN_ERROR_OK" Then
{
    Show LAN-Channel successfully initialized. Bit rate: 500 kBit/s
}
Else
{
    If "Initialize result equals PCAN_ERROR_WARNING" Then
    {
        Get the value PCAN_BUSSPEED_NOMINAL as: BUSSPEED
        Show LAN-Channel successfully initialized. Bit rate:
        Print (BUSSPEED / 1000) kBit/s
    }
    Else
    {
        Show Error! Channel couldn't be initialized!
        Terminate
    }
}
}
```

PCAN_INTERFRAME_DELAY

このパラメーターは、ユーザーが PCAN-Channel 内で送信される CAN フレーム間に μ s の解像度で pause / delay をコンフィグレーションするのに役立ちます。同じ PCAN-Hardware を使用している他のアプリケーション（例えば、PCAN-View）は、delay を設定しても影響を受けません。

注: この機能は、ファームウェアバージョンが 2.4.0 以上の FPGA ベースのデバイスでのみサポートされます。このドキュメントの執筆時点では、FPGA ベースの PCAN-USB デバイス (PCAN-USB FD、PCAN-USB Pro FD、PCAN-Chip USB) のみが PCAN_INTERFRAME_DELAY をサポートしています。

Availability

バージョン 4.2.0 以降で使用できます。

Supported by

PCAN-PCI (チャンネル PCAN_PCIBUS1 から PCANPCIBUS16)

PCAN-USB (チャンネル PCAN_USBBUS1 から PCAN_USBBUS16)

Access Mode

このパラメーターは read / write です。設定して読み取ることができます。

Possible Values

この値は、[0 ... 1023] マイクロ秒の範囲内である必要があります。設定する値がファームウェアでサポートされている解像度よりも大きい場合、値は切り捨てられます。

Default Value

PCAN_INTERFRAME_DELAY のデフォルト値は 0 です。これは、遅延が非アクティブ化されていることを意味します。0 より大きな値を設定すると、PCAN_INTERFRAME_DELAY は、明示的に非アクティブ化されるか（0 に設定される）か、またはチャンネルが切断される（たとえば、関数 CAN_Uninitialize を使用するなど）まで使用されます。

Initialization Status

このパラメーターは、初期化されたチャンネルでのみ使用できます。

When to Use

連続して送信される CAN フレームの分離時間をインクリメントしたいアプリケーションで使用できます。

Application – Example of Use

例えば、PCAN-USB Pro FD のような FPGA ベースのデバイスを使用して、いくつかの ECU をフラッシュするアプリケーションがあるとして。あなたの ECU は分散しており、ゲートウェイを使用して接続されているため、小さな伝送遅延が発生する可能性があります。FPGA デバイスは最大 100% のバス負荷をサポートすることができるため、アプリケーションがデータを高速に送信し、ISO-TP や UDS のようなクライアント/サーバーモデルに依存している場合、使用されるフラッシングプロトコルに問題が発生する可能性があります。パッケージ間にわずかな遅延を設定することで、バス負荷の最大値に達しないようにし、プロトコルが障害なく動作するようにすることができます。

```

Initialize the Channel PCAN_USBBUS1 at any bit rate, e.g. 0x001C (500 kBit/s)
If "Initialize result equals PCAN_ERROR_OK" Then
{
    Set the value PCAN_INTERFRAME_DELAY on Channel PCAN_USBBUS1 To 10
    If "PCAN_INTERFRAME_DELAY result equals PCAN_ERROR_OK" Then
    {
        Show Interframe delay set to 10 microseconds
        Do work
    }
    Else
    {
        Show Error! Interframe delay couldn't be set / is not supported!
    }
}
Else
{
    Show Error! Channel couldn't be initialized!
    Terminate
}

```

データフローの制御

これらのパラメーターは、PCAN-Channel を介して受信されるデータ、その受信方法、さらにはアプリケーションが新しい受信データをチェックする方法／タイミングを制御することを目的としています。CAN ネットワーク内で送信される情報の量に応じて、アプリケーションが受け入れるデータを制限することは、作業を容易にするために合理的です。

多くのデータを受信し、その一部を処理するだけでは、メモリや CPU の処理が無駄になり、システムの動作が遅くなります。同じように、入力されたデータを読み取るための反応時間も、入力された情報をうまく処理するための鍵となります。

PCAN_RECEIVE_EVENT

このパラメーターは、イベントハンドル ([Windows Event Objects](#)) を基になる API に渡します。このイベントは、CAN データが PCAN-Channel の受信キューに入るとトリガーされます（その状態は “signaled” に設定されます）。

イベントは通常、アプリケーションが異なる実行スレッドで処理を分離するときに使用されます。スレッドでは、イベントの発生を待機しても、アプリケーションの通常の実行には影響しません。

メッセージがキューに入るたびにイベントがトリガーされるのではなく、“not signaled” の状態でデータが受信された場合にのみイベントが通知されることに**注意してください**。イベントが通知されたら、空になるまでキューを読み取り、最終的にイベントをリセットする必要があります（手動リセットイベントを使用している場合）。

Availability

バージョン 1.0.0 以降で使用できます。

Supported by

PCAN-ISA（チャネル PCAN_ISABUS1 から PCAN_ISABUS8）

PCAN-DNG（チャネル PCAN_DNGBUS1）

PCAN-PCI（チャネル PCAN_PCIBUS1 から PCANPCIBUS16）

PCAN-USB（チャネル PCAN_USBBUS1 から PCAN_USBBUS16）

PCAN-PCC（チャネル PCAN_PCCBUS1 から PCAN_PCCBUS2）

PCAN-LAN（チャネル PCAN_LANBUS1 から PCAN_LANBUS16）

Access Mode

このパラメーターは read / write です。設定して読み取ることができます。

Possible Values

このパラメーターは、アクティブ化 または 非アクティブ化 できます。

Status	Value needed
ENABLED	Windows 関数 CreateEvent によって返される有効なイベントオブジェクトハンドル。
DISABLED	0、または NULL、または IntPtr.Zero（管理対象環境）。

Default Value

デフォルトの状態は無効(0)です。このパラメーターを有効にした後(イベントハンドルを設定することにより)、PCAN-Basic API は、ハンドルが無効になるまで(ハンドルに値 0 を設定することにより)、またはチャンネルが切断されるまで(たとえば、関数 CAN_Uninitialize を使用して)ハンドルに信号を送信しようとします。

PCAN-Channel を再初期化する必要がある場合、イベントは初期化後に Default Value の 0 に戻るため、チャンネルの初期化後に毎回イベントを再設定する必要があることに**注意してください**。API が無効なハンドルを設定しようとする可能性があり、望ましくない動作を引き起こす可能性があるため、**PCAN-Channel の初期化を解除した後は、([CloseHandle](#) を使用して) ハンドルを閉じることが強く推奨されることにも注意してください**。

Initialization Status

このパラメーターを使用する前に、PCAN-Channel を初期化する必要があります。

When to Use

タイムアウトの回避：アプリケーションが可能な限り迅速に情報に反応し処理したい場合に使用できます。不要なデータポーリングを避けるために使用することができます。アプリケーションが、めったに受信されない、またはいつ到着するかわからない特定のメッセージをチェックする必要がある場合です。

Application – Example of Use

デバイス Electronic Control Unit (電子制御ユニットなど) のデータ更新に使用される診断アプリケーションを作成したとします。アプリケーションは、デバイスが初期化されるまで待機してから、デバイスをメンテナンスモードに設定するためのメッセージを送信する必要があります。デバイスは、メンテナンスメッセージを受信してから最初の 10 ms 以内に応答する必要があります。そうしないと、目的のモードに入ることができません。このためには、リクエストを送信するスレッドを開始し、レスポンスを待ちます：

```

InitializeFunction
{
    Initialize the Channel-USED
    Create AutoResetEvent using the function "CreateEvent"
    Mark AppMode: Normal-Mode
    Start ThreadFunction
}

ThreadFunction
{
    Set PCAN_RECEIVE_EVENT on Channel-USED to AutoResetEvent
    If "PCAN_RECEIVE_EVENT" result equals PCAN_ERROR_OK Then
    {
        Send Diagnostic-Message
        If "Diagnostic-Message was sent" then
        {
            Wait until AutoResetEvent is signaled or timeout(10)...
            Read Message
            If "Message is maintenance-confirmation" Then
            {
                Mark AppMode: Maintenance-Mode
            }
            Set PCAN_RECEIVE_EVENT on Channel-USED to 0
        }
    }
}

MainFunction
{
    If "AppMode equals Maintenance-Mode" Then
    {
        Show Application is in MAINTENANCE mode
    }
    Else
    {
        Show Application is in NORMAL mode
    }
}

```


PCAN_MESSAGE_FILTER

このパラメーターは、CAN チップのアクセプタンス・マスクとアクセプタンス・コードを変更することにより、メッセージを受信するかどうかを PCAN-Channel に指示します。

アクセプタンス・マスクとアクセプタンス・コードを変更する必要がある場合は、内部ハードウェア・リセットが実行されることに**注意してください**。他のアプリケーションが同じデバイスを使用している場合、そのアプリケーションの通信に影響を与える可能性があります。

Availability

バージョン 1.0.0 以降で使用できます。

Supported by

PCAN-ISA (チャンネル PCAN_ISABUS1 から PCAN_ISABUS8)

PCAN-DNG (チャンネル PCAN_DNGBUS1)

PCAN-PCI (チャンネル PCAN_PCIBUS1 から PCANPCIBUS16)

PCAN-USB (チャンネル PCAN_USBBUS1 から PCAN_USBBUS16)

PCAN-PCC (チャンネル PCAN_PCCBUS1 から PCAN_PCCBUS2)

PCAN-LAN (チャンネル PCAN_LANBUS1 から PCAN_LANBUS16)

Access Mode

このパラメーターは read / write です。設定して読み取ることができます。

Possible Values

設定操作では、このパラメーターを opened あるいは closed に設定することができます。

Defined Value	Description
PCAN_FILTER_OPEN	CAN フィルタは、すべてのメッセージの通過を許可します。
PCAN_FILTER_CLOSE	CAN フィルタはすべてのメッセージを破棄します。

取得操作では、3 番目の値を受け取ることができます。

Defined Value	Description
PCAN_FILTER_CUSTOM	CAN フィルタを使用すると、カスタム範囲のメッセージを渡すことができます。

Default Value

フィルタのデフォルトの状態は、すべてのメッセージ (PCAN_FILTER_OPEN) を受信することです。PCAN-Channel は、チャンネルが初期化された直後に、CAN ネットワークで送信されているあらゆるメッセージの受信を開始することに**注意してください**。また、関数 CAN_FilterMessages を使用すると、関数を呼び出す前のフィルタの状態が PCAN_FILTER_OPEN であった場合、目的のメッセージ範囲を登録する前にフィルタが自動的に閉じられてしまうことにも**注意してください**。

Initialization Status

このパラメーターを使用する前に、PCAN-Channel を初期化する必要があります。

When to Use

これは、たとえば、定義された期間中に不要なメッセージを受信しないようにするなど、特定の時間内のメッセージの受け入れを切り替えるために使用することができます。

Application – Example of Use

CAN ネットワークからかなりの量の情報を読み取って解釈し、それをいくつかのビジュアルコントロールに表示するアプリケーションがあるとします。データの変動が速すぎるため、いつかデータの一般的なステータスを確認する必要がありますが、ネットワーク内で送信される情報をフリーズすることはできません。CAN フィルタをしばらく閉じて、最後に受信した情報がビジュアルコントロールに表示させ、それを確認するのに十分な時間を確保することができます：

```

Set the value PCAN_MESSAGE_FILTER on Channel-USED to PCAN_FILTER_CLOSE
If "PCAN_MESSAGE_FILTER result equals PCAN_ERROR_OK" Then
{
    Show Filter is closed
    Do needed checking
    Show Check is finished. Enabling communication again...
    Set the value PCAN_MESSAGE_FILTER on Channel-USED to PCAN_FILTER_OPEN
    If "PCAN_MESSAGE_FILTER result equals PCAN_ERROR_OK" Then
    {
        Show Filter is open
    }
    Else
    {
        Show Error: Filter couldn't be reestablished
    }
}
Else
{
    Show Error: Filter couldn't be closed
}

```

PCAN_RECEIVE_STATUS

このパラメーターは、受信フィルタの値に関係なく、ユーザーが PCAN-Channel 内のメッセージ (Data : データ、Status : ステータス、および Error : エラー フレーム) の受信を allow / disallow するのに役立ちます。PCAN-Channel の受信フィルタは変更されません (同じ PCAN-Hardware で動作する他のアプリケーションは妨害されません)。

このパラメーターは、いわゆる **“pre-initialized”** パラメーターです。つまり、PCAN-Channel が初期化される前に設定することで、パラメーターの activate / deactivate を可能な限り迅速に行い、機密性の高い操作で発生する問題を回避することができます。

Availability

バージョン 1.1.0 以降で使用できます。

Supported by

PCAN-ISA (チャネル PCAN_ISABUS1 から PCAN_ISABUS8)

PCAN-DNG (チャネル PCAN_DNGBUS1)

PCAN-PCI (チャネル PCAN_PCIBUS1 から PCANPCIBUS16)

PCAN-USB (チャネル PCAN_USBBUS1 から PCAN_USBBUS16)

PCAN-PCC (チャネル PCAN_PCCBUS1 から PCAN_PCCBUS2)

PCAN-LAN (チャネル PCAN_LANBUS1 から PCAN_LANBUS16)

Access Mode

このパラメーターは read / write です。設定して読み取ることができます。

Possible Values

このパラメーターは、アクティブ化 または 非アクティブ化 できます。

Defined Value	Description
PCAN_PARAMETER_OFF	メッセージ受信状態は OFF です。
PCAN_PARAMETER_ON	メッセージ受信状態は ON です。

Default Value

受信ステータスの Default Value が有効化されます (PCAN_PARAMETER_ON)。非アクティブ化した後、受信ステータスは、明示的に再アクティブ化されるか、チャンネルが切断されるまで（たとえば、関数 CAN_Uninitialize を使用して）非アクティブのままになります。

Initialization Status

このパラメーターは、初期化されたチャンネルまたは初期化されていないチャンネルで使用できます。

When to Use

これは、メッセージフィルタを変更することなく、しばらくの間メッセージを破棄したいアプリケーションで使用でき、使用されているデバイス内の障害を回避することができます。

Application – Example of Use

例えば、12 種類のメッセージレンジなど、複雑なフィルタを使用するアプリケーションがあるとしてします。ある時点で、フィルタを再設定することなく、しばらくの間メッセージの受信を停止する必要があります。これにより、CAN コントローラーのリセット（フィルタの再設定が必要な場合に発生）を回避できます：

```

Set the value PCAN_RECEIVE_STATUS on Channel-USED to PCAN_PARAMETER_OFF
If "PCAN_RECEIVE_STATUS" result equals PCAN_ERROR_OK Then
{
  Show Message receiving is disabled
  Do needed operations
  Set the value PCAN_RECEIVE_STATUS on Channel-USED to PCAN_PARAMETER_ON
  If "PCAN_RECEIVE_STATUS" result equals PCAN_ERROR_OK Then
  {
    Show Normal operation reestablished. Message receiving enabled
  }
  Else
  {
    Show Error: Receiving status couldn't be reestablished
  }
}
Else
{
  Show Error: Receiving status couldn't be changed
}

```

PCAN_ALLOW_STATUS_FRAMES

このパラメーターは、PCAN-Channel 内での Status フレームの受信を allow / disallow するためのものです。このパラメーターは、PCAN-Channel のアクセプタンス・フィルタには影響しません。さらに、同じ PCAN-Hardware で動作する他のアプリケーションは引き続き Status フレームを受信します。

PCAN_RECEIVE_STATUS パラメーターを無効にすると、Status フレームの受信も抑制されることに**注意してください**。

Availability

バージョン 4.2.0 以降で使用できます。

Supported by

PCAN-ISA (チャンネル PCAN_ISABUS1 から PCAN_ISABUS8)

PCAN-DNG (チャンネル PCAN_DNGBUS1)

PCAN-PCI (チャンネル PCAN_PCIBUS1 から PCANPCIBUS16)

PCAN-USB (チャンネル PCAN_USBBUS1 から PCAN_USBBUS16)

PCAN-PCC (チャンネル PCAN_PCCBUS1 から PCAN_PCCBUS2)

PCAN-LAN (チャンネル PCAN_LANBUS1 から PCAN_LANBUS16)

Access Mode

このパラメーターは read / write です。設定して読み取ることができます。

Possible Values

このパラメーターは、アクティブ化 または 非アクティブ化 できます。

Defined Value	Description
PCAN_PARAMETER_OFF	Status フレームの受信はオフです。
PCAN_PARAMETER_ON	Status フレームの受信はオンです。

Default Value

Status フレーム受信のデフォルト値はアクティブになります (PCAN_PARAMETER_ON)。非アクティブ化した後、Status フレームの受信は、明示的に再アクティブ化されるか、チャンネルが切断されるまで（たとえば、関数 CAN_Uninitialize を使用して）非アクティブのままになります。

Initialization Status

このパラメーターは、初期化されたチャンネルでのみ使用できます。

When to Use

アクセプタンス・フィルタを使用してこれを行うことはできないため、Status フレームを allow / discard するアプリケーションで使用できます。

Application – Example of Use

メッセージを送信してデバイスをウェイクアップする必要があるアプリケーションがあるとします。デバイスがスリープモードになっているため、ウェイクアップメッセージを送信すると、バスに何らかの障害が発生する可能性があります。デバイスが起動して実行されるまで、しばらくの間 Status フレームの受信を無効にすることができます：

```

Set the value PCAN_ALLOW_STATUS_FRAMES on Channel-USED to PCAN_PARAMETER_OFF
If "PCAN_ALLOW_STATUS_FRAMES" result equals PCAN_ERROR_OK" Then
{
  Show The reception of Status Frames is disabled
  Do needed operations
  Set the value PCAN_ALLOW_STATUS_FRAMES on Channel-USED to PCAN_PARAMETER_ON
  If "PCAN_ALLOW_STATUS_FRAMES" result equals PCAN_ERROR_OK" Then
  {
    Show Normal operation reestablished. Status Frames enabled
  }
  Else
  {
    Show Error: reception of Status Frames couldn't be reestablished
  }
}
Else
{
  Show Error: Reception of Status Frames couldn't be changed
}

```

PCAN_ALLOW_RTR_FRAMES

このパラメーターは、ユーザーが PCAN-Channel 内の RTR フレームの受信を allow / disallow するためのものです。このパラメーターは、PCAN-Channel のアクセプタンス・フィルタには影響しません。さらに、同じ PCAN-Hardware で動作する他のアプリケーションは引き続き RTR フレームを受信します。

PCAN_RECEIVE_STATUS パラメーターを無効にすると、RTR フレームの受信も抑制されることに**注意してください**。

Availability

バージョン 4.2.0 以降で使用できます。

Supported by

PCAN-ISA (チャネル PCAN_ISABUS1 から PCAN_ISABUS8)
 PCAN-DNG (チャネル PCAN_DNGBUS1)
 PCAN-PCI (チャネル PCAN_PCIBUS1 から PCANPCIBUS16)
 PCAN-USB (チャネル PCAN_USBBUS1 から PCAN_USBBUS16)
 PCAN-PCC (チャネル PCAN_PCCBUS1 から PCAN_PCCBUS2)
 PCAN-LAN (チャネル PCAN_LANBUS1 から PCAN_LANBUS16)

Access Mode

このパラメーターは read / write です。設定して読み取ることができます。

Possible Values

このパラメーターは、アクティブ化 または 非アクティブ化 できます。

Defined Value	Description
PCAN_PARAMETER_OFF	RTR フレームの受信はオフです。
PCAN_PARAMETER_ON	RTR フレームの受信はオンです。

Default Value

RTR フレーム受信のデフォルト値はアクティブになります (PCAN_PARAMETER_ON)。非アクティブ化した後、RTR フレームの受信は、明示的に再アクティブ化されるか、チャンネルが切断されるまで（たとえば、関数 CAN_Uninitialize を使用して）非アクティブのままになります。

Initialization Status

このパラメーターは、初期化されたチャンネルでのみ使用できます。

When to Use

これは、RTR フレームをしばらく allow / discard したいアプリケーションで使用することができ、メッセージフィルタに手を加えることなく、使用するデバイス内の障害を回避することができます。

Application – Example of Use

変化する可能性のある情報で RTR フレームに応答するアプリケーションがあるとします。(例えば、ユーザーが設定できる。) ユーザーがこの情報を更新している間、RTR を処理するコードを停止または無効にすることなく、RTR メッセージの受信（およびその処理）を非アクティブ化にすることができます：

```
Set the value PCAN_ALLOW_RTR_FRAMES on Channel-USED to PCAN_PARAMETER_OFF
If "PCAN_ALLOW_RTR_FRAMES" result equals PCAN_ERROR_OK Then
{
    Show The reception of RTR Frames is disabled
    Do needed operations
    Set the value PCAN_ALLOW_RTR_FRAMES on Channel-USED to PCAN_PARAMETER_ON
    If "PCAN_ALLOW_RTR_FRAMES" result equals PCAN_ERROR_OK Then
    {
        Show Normal operation reestablished. RTR Frames enabled
    }
    Else
    {
        Show Error: reception of RTR Frames couldn't be reestablished
    }
}
Else
{
    Show Error: Reception of RTR Frames couldn't be changed
}
```

PCAN_ALLOW_ERROR_FRAMES

このパラメーターは、ユーザーが PCAN-Channel 内の CAN Error フレームの受信を allow / disallow するためのものです。このパラメーターは、PCAN-Channel のアクセプタンス・フィルタには影響しません。さらに、同じ PCAN-Hardware で動作する他のアプリケーションは引き続きエラーフレームを受信します。

PCAN_RECEIVE_STATUS パラメーターを無効にすると、Error フレームの受信も抑制されることに**注意してください**。

Availability

バージョン 4.2.0 以降で使用できます。

Supported by

PCAN-ISA (チャンネル PCAN_ISABUS1 から PCAN_ISABUS8)

PCAN-DNG (チャンネル PCAN_DNGBUS1)

PCAN-PCI (チャンネル PCAN_PCIBUS1 から PCANPCIBUS16)

PCAN-USB (チャンネル PCAN_USBBUS1 から PCAN_USBBUS16)

PCAN-PCC (チャンネル PCAN_PCCBUS1 から PCAN_PCCBUS2)

PCAN-LAN (チャンネル PCAN_LANBUS1 から PCAN_LANBUS16)

Access Mode

このパラメーターは read / write です。設定して読み取ることができます。

Possible Values

このパラメーターは、アクティブ化 または 非アクティブ化 できます。

Defined Value	Description
PCAN_PARAMETER_OFF	CAN Error フレームの受信は OFF です。
PCAN_PARAMETER_ON	CAN Error フレームの受信は ON です。

Default Value

CAN Error フレーム受信の Default Value は無効になっています (PCAN_PARAMETER_OFF)。それをアクティブにした後、CAN Error フレームの受信は、明示的に非アクティブになるか、チャンネルが切断されるまで（たとえば、関数 CAN_Uninitialize を使用して）アクティブのままになります。

Initialization Status

このパラメーターは、初期化されたチャンネルでのみ使用できます。

When to Use

アクセプタンス・フィルタを使用してこれを行うことはできないため、CAN Error フレームを allow / discard するアプリケーションで使用することができます。

Application – Example of Use

CAN 通信に関して期待される動作をしないアプリケーションがあるとします。エラーフレームを有効にし、CAN バスに障害が発生しているかどうかを確認し、考えられる原因についての詳細な情報を得ることができます：

```

Set the value PCAN_ALLOW_ERROR_FRAMES on Channel-USED to PCAN_PARAMETER_ON
If "PCAN_ALLOW_ERROR_FRAMES" result equals PCAN_ERROR_OK Then
{
    Show The reception of RTR Frames is enabled
    Do needed operations
    Set the value PCAN_ALLOW_ERROR_FRAMES on Channel-USED to PCAN_PARAMETER_OFF
    If "PCAN_ALLOW_ERROR_FRAMES" result equals PCAN_ERROR_OK Then
    {
        Show Normal operation reestablished. Error Frames disabled
    }
    Else
    {
        Show Error: reception of Error Frames couldn't be disabled
    }
}
Else
{
    Show Error: Reception of Error Frames couldn't be changed
}

```

PCAN_ACCEPTANCE_FILTER_11BIT

このパラメーターは、SJA1000 CAN コントローラーのアクセプタンス・フィルタに指定されているように、ユーザーが特定の 11 ビットのアクセプタンス・コードとアクセプタンス・マスクを使用して PCAN-Channel のアクセプタンス・フィルタを構成するのに役立ちます。

このパラメーターを使用すると、複雑なフィルターパターンを構成でき、CAN の知識が豊富なユーザーを対象としています。マスクとコードパターンの計算は簡単なことではないことに注意してください。ほとんどのアプリケーションでは、メッセージの受信範囲を設定するために関数 CAN_FilterMessages を使用する方が適切です。コードとマスクの計算については、付録 D に簡単な例があります。

Notes:

- アクセプタンス・コードとアクセプタンス・マスクは 64 ビット値で一緒にコード化され、それぞれが 4 バイトを使用します。**アクセプタンス・コードは最上位**バイトに格納されます。64 ビットの符号なし整数変数との間で値をコーディングおよびデコードするには、ビット単位のシフト演算が必要です。
- 29 ビット CAN ID を示すアクセプタンス・コードとアクセプタンス・マスクを設定するには、パラメーター PCAN_ACCEPTANCE_FILTER_29BIT を使用する必要があります。
- SJA1000CAN コントローラーには、standard (11 ビット) ID と extended (29 ビット) ID の両方に対して 1 つのアクセプタンス・フィルタしかありません。11 ビット ID の設定を行う場合、適応策としてアクセプタンス・マスクとアクセプタンス・コードが内部的に左にシフトされます。これにより、不要なメッセージが受信される可能性もあります。このため、11 ビットと 29 ビットのフィルタを混在させることは好ましくありません。
- 内部ハードウェア・リセットは、アクセプタンス・フィルタが変更されるたびに実行されます。他のアプリケーションが同じデバイスを使用している場合、そのアプリケーションの通信に影響を与える可能性があります。

Availability

バージョン 4.2.0 以降で使用できます。

Supported by

PCAN-ISA (チャネル PCAN_ISABUS1 から PCAN_ISABUS8)
 PCAN-DNG (チャネル PCAN_DNGBUS1)
 PCAN-PCI (チャネル PCAN_PCIBUS1 から PCANPCIBUS16)
 PCAN-USB (チャネル PCAN_USBBUS1 から PCAN_USBBUS16)
 PCAN-PCC (チャネル PCAN_PCCBUS1 から PCAN_PCCBUS2)
 PCAN-LAN (チャネル PCAN_LANBUS1 から PCAN_LANBUS16)

Access Mode

このパラメーターは read / write です。設定して読み取ることができます。

Possible Values

このパラメーターの解像度は 4 ワードです。実際には、アクセプタンス・**コード**とアクセプタンス・**マスク**を表す 2 つのダブルワード値が含まれているため、このパラメーターが許容する最大値の範囲は、それらの内部値の限界である [0..16838] の範囲となります。つまり、64 ビット値としてのこのパラメーターの最大値は 70364449226751、つまり 16 進数で **00003FFF00003FFFh** となります。

Default Value

アクセプタンス・フィルタのデフォルトの状態は、すべてのメッセージを受信することです (PCAN_FILTER_OPEN)。これは、デフォルトのアクセプタンス・**コード 0h** とアクセプタンス・**マスク 7FFh** (**00000000000007FFh**) を表しています。このパラメーターを使用する前のフィルタの状態が PCAN_FILTER_OPEN であった場合、目的のコードとマスクを登録する前に、フィルタの自動リセットが実行されることに**注意してください**。

Initialization Status

このパラメーターを使用する前に、PCAN-Channel を初期化する必要があります。

When to Use

識別子が具体的なパターンに従っている特定の CAN メッセージの受信を allow または block する必要があり、そのパターンを単純なメッセージの範囲として表すことが難しい場合に使用されます。

Application – Example of Use

診断目的で ECU からデータを読み取るアプリケーションを作成するとします。ECU は定期的に多くの情報を送信しますが、関心があるのは 100h、400h、500h の 3 つのメッセージだけです。関数 CAN_FilterMessage を使用すると、各 ID に対して 1 回ずつ、合計 3 つのコールを実行することになり、3 回のハードウェア・リセットが発生します。パラメーター PCAN_ACCEPTANCE_FILTER_11BIT と値 **0000000000000500h** を使用して CAN_SetValue を 1 回コールするだけで同じ効果が得られ、アクセプタンス・フィルタはこれら 3 つの ID の受信のみを許可しますが、2 つの関数コールと 2 つの不要なハードウェア・リセットを節約できます。

```
Set the value PCAN_ACCEPTANCE_FILTER_11BIT on Channel-USED to 0x0000000000000500
If "PCAN_ACCEPTANCE_FILTER_11BIT result equals PCAN_ERROR_OK" Then
{
    Show Filter was configured to accept the standard IDs: 0x100, 0x400, and 0x500
}
Else
{
    Show Error: The 11-bit acceptance filter couldn't be set
}
```

PCAN_ACCEPTANCE_FILTER_29BIT

このパラメーターは、PCAN-Channel の受アクセプタンス・フィルタを、SJA1000 CAN コントローラーの受信フィルタで指定されているように、特定の 29 ビットのアクセプタンス・コードとアクセプタンス・マスクで設定するのに役立ちます。

このパラメーターでは、複雑なフィルターパターンを設定することができ、CAN に関する幅広い知識を持つユーザーを対象としています。マスクとコードパターンの計算は、単純な問題ではないことに注意してください。ほとんどのアプリケーションでは、関数 CAN_FilterMessages を使ってメッセージの受信範囲を設定する方が適切です。コードとマスクの計算については、付録 D に簡単な例があります。

Notes:

- アクセプタンス・コードとアクセプタンス・マスクは 64 ビット値で一緒にコード化され、それぞれが 4 バイトを使用します。アクセプタンス・コードは最上位バイトに格納されます。64 ビットの符号なし整数変数との間で値をコーディングおよびデコードするには、ビット単位のシフト演算が必要です。
- 11 ビット CAN ID を示すアクセプタンス・コードとアクセプタンス・マスクを設定するには、PCAN_ACCEPTANCE_FILTER_11BIT を使用する必要があります。
- SJA1000CAN コントローラーには、standard (11 ビット) ID と extended (29 ビット) ID の両方に対して 1 つのアクセプタンス・フィルタしかありません。11 ビット ID の設定を行う場合、適応策としてアクセプタンス・マスクとコードが内部的に左にシフトされます。これにより、不要なメッセージが受信される可能性もあります。このため、11 ビットと 29 ビットのフィルタを混在させることもお勧めしません。
- 内部ハードウェア・リセットは、アクセプタンス・フィルタが変更されるたびに実行されます。他のアプリケーションが同じデバイスを使用している場合、シナリオによってはその通信が影響を受ける可能性があります。

Availability

バージョン 4.2.0 以降で使用できます。

Supported by

PCAN-ISA (チャネル PCAN_ISABUS1 から PCAN_ISABUS8)
 PCAN-DNG (チャネル PCAN_DNGBUS1)
 PCAN-PCI (チャネル PCAN_PCIBUS1 から PCANPCIBUS16)
 PCAN-USB (チャネル PCAN_USBBUS1 から PCAN_USBBUS16)
 PCAN-PCC (チャネル PCAN_PCCBUS1 から PCAN_PCCBUS2)
 PCAN-LAN (チャネル PCAN_LANBUS1 から PCAN_LANBUS16)

Access Mode

このパラメーターは read / write です。設定して読み取ることができます。

Possible Values

このパラメーターの解像度は 4 ワードです。実際には、アクセプタンス・コードとアクセプタンス・マスクを表す 2 つのダブルワード値が含まれているため、このパラメーターで受け入れられる最大値の範囲は、内部値の制限である [0 ..4294967295] の間の範囲となります。つまり、64 ビット値としてのこのパラメーターの最大値は 18446744073709551615、つまり 16 進数の FFFFFFFF FFFFFFFFh となります。

Default Value

受信フィルタのデフォルトの状態は、すべてのメッセージを受信することです (PCAN_FILTER_OPEN)。これは、デフォルトのアクセプタンス・コード 0h とアクセプタンス・マスク 1FFFFFFFh (000000001FFFFFFFh) を表しています。このパラメーターを使用する前のフィルタの状態が PCAN_FILTER_OPEN であった場合、目的のコードとマスクを登録する前に、フィルタの自動リセットが実行されることに注意してください。

Initialization Status

このパラメーターを使用する前に、PCAN-Channel を初期化する必要があります。

When to Use

識別子が具体的なパターンに従っている特定の CAN メッセージの受信を allow または block する必要があり、そのパターンを単純なメッセージの範囲として表現することが困難な場合に使用されます。

Application – Example of Use

診断目的で ECU からデータを読み取るアプリケーションを作成するとします。ECU は定期的に多くの情報を送信しますが、関心があるのは 1100h、1400h、1500h の 3 つのメッセージだけです。関数 CAN_FilterMessage を使用すると、各 ID に対して 1 回ずつ、合計 3 回のコールが必要となり、3 回のハードウェア・リセットが発生します。パラメーター PCAN_ACCEPTANCE_FILTER_29BIT と値 0000100000000500h を使用して CAN_SetValue を 1 回呼び出すだけで、同じ効果を得ることができます。アクセプタンス・フィルタは、これら 3 つの ID の受信のみを許可しますが、2 つの関数呼び出しと 2 つの不要なハードウェア・リセットを節約できます。

```
Set the value PCAN_ACCEPTANCE_FILTER_29BIT on Channel-USED to 0x0000100000000500  
If "PCAN_ACCEPTANCE_FILTER_29BIT result equals PCAN_ERROR_OK" Then  
{  
    Show Filter was configured to accept the standard IDs: 0x1100, 0x1400, and 0x1500  
}  
Else  
{  
    Show Error: The 29-bit acceptance filter couldn't be set  
}
```

ロギングパラメータの使用

これらのパラメータは、デバッグ操作を支援することにより、PCAN-Basic プロジェクトの開発フェーズをサポートすることを目的としています。ロギングシステムを使用すると、API の使用における論理的な問題の発見、送受信されるデータの問題の検出、パラメーターデータの確認、コマンドの順序などに役立ちます。

また、アプリケーションのコードを変更せずに、ログ機能をアクティブ化または非アクティブ化、および設定することもできます。これにより、アプリケーションが既にリリースされた後にデバッグセッションを行うことができます。これについての詳細は、オンラインフォーラム [Activate debug-logging over Windows Registry](#) (Windows レジストリを介したデバッグログのアクティブ化)、または付録 A をご参照ください。

ロギング機能は、特に PCAN-Channel に関連付けられているのではなく、PCAN-Basic ライブラリ自体の使用に関連付けられています。これは、3 つの重要なポイントを意味します。

- PCAN_LOG_*パラメーターが使用されている場合、CAN_GetValue / CAN_SetValue で使用する PCAN-Channel ハンドルは PCAN_NONEBUS である必要があります。その他の値を指定すると、この関数は失敗します。
- ログに記録されるデータは、PCAN-Basic dll をロードしたプロセスによって発行された API コールに対応します。
- 同じアプリケーションの異なるスレッドに対してデバッグセッションを開始することはできません。

PCAN_LOG_LOCATION

この値は、デバッグセッション内で、ログファイルが保存されるコンピューター上のフォルダを設定するために使用されます。

この値を設定すると、デバッグ情報の記録が自動的に開始されることに**注意してください**。通常は実行する必要のないコードの任意の部分にこのパラメーターの呼び出しを含めることができるため、このポイントに達した場合は（一種のアサーションとして）ログファイルを通じて通知されます。

デバッグセッションが実行されている場合（ログファイルが書き込まれている場合）、PCAN_LOG_LOCATION は API に、現在のログファイルを閉じて、新しいフォルダ情報でプロセスを再開するように指示します。ログファイルの名前は指定できないことにも**注意してください**。ログファイルの名前は常に **PCANBasic.log** です。

Availability

バージョン 1.0.0 以降で使用できます。

Supported by

PCAN_NONEBUS:ロギングパラメータはグローバルに使用されます。つまり、特定の PCAN-Channel ではなく、特定のプロセスに関連付けられます。

Access Mode

このパラメーターは read / write です。設定して読み取ることができます。

Possible Values

この値は、実行中のコンピューター上の既存のディレクトリへの完全修飾された有効なパスを含む文字列です。デフォルトのパス（process パスの呼び出し）を使用するには、空の文字列を設定する必要があります。

Kind of Path	Value needed
CUSTOM Path	有効なディレクトリ文字列（ ファイルとパス ）。
DEFAULT Path	空の文字列（process フォルダを呼び出しています）。

Default Value

Default Value は、呼び出し元の process フォルダへのパスです。

Initialization Status

適用されません。このパラメーターを使用するために、PCAN-Channel を初期化する必要はありません。

When to Use

異なるパスを割り当て、複数の PCANBasic.log ファイルを作成することにより、デバッグセッションまたはログセッションを区別する場合に使用できます。

Application – Example of Use

同じプログラムの複数のインスタンスを開始し、それらすべてを同時にデバッグしたいとします。さらに、アプリケーションごとにログファイルを分けたいとします。それぞれのフォルダを作成し、各アプリケーションのパスを設定することで、それぞれが独自のログファイルを作成できるようにすることができます：

```
Create Folder ".¥Process-ID"
Mark NEW_PATH: ".¥Process-ID"
Do in each application
{
    Set the value PCAN_LOG_LOCATION to NEW_PATH
    If "PCAN_LOG_LOCATION" result equals PCAN_ERROR_OK Then
    {
        Show Logging is active. Path for the Log file is
        Print NEW_PATH
        Do Needed operations
    }
    Else
    {
        Show Error: Log's Path couldn't be configured
    }
}
```

PCAN_LOG_STATUS

このパラメーターは、ユーザーが PCAN-Basic API 内のデバッグセッションのアクティビティステータスを制御するのに役立ちます。

ログファイルの保存先パスを設定せず、また、ログに記録する情報を設定せずに、ロギングステータスを ON に設定した場合、セッションプロセスは Default Value で開始され、ログファイルは呼び出したプロセスがあるフォルダに置かれ、例外のみが記録されることとなりますので**ご注意ください**。

Availability

バージョン 1.0.0 以降で使用できます。

Supported by

PCAN_NONEBUS:ロギングパラメータはグローバルに使用されます。つまり、特定の PCAN-Channel ではなく、特定のプロセスに関連付けられています。

Access Mode

このパラメーターは read / write です。設定して読み取ることができます。

Possible Values

このパラメーターは、アクティブ化 または 非アクティブ化 できます。

Defined Value	Description
PCAN_PARAMETER_OFF	ロギングはオフです。
PCAN_PARAMETER_ON	ロギングはオンです。

Default Value

ロギングモードのデフォルト値は非アクティブになっています (PCAN_PARAMETER_OFF)。アクティブ化した後、ログ機能は明示的に非アクティブ化されるまでアクティブのままになります。

Initialization Status

適用されません。このパラメーターを使用するために、PCAN-Channel を初期化する必要はありません。

When to Use

デバッグセッションを中断 (start, stop, restart など) するために使用できます。

Application – Example of Use

例えば、アプリケーションをデバッグしたいとします。すでに、断続的な問題があることはわかっています。調査対象の問題に関する情報を含む可能性のあるログデータのみを取得するために、異常が発生した瞬間にのみデバッグセッションをアクティブ化することができます：

```
Function ActivateLogging
{
    Set the value PCAN_LOG_STATUS to PCAN_PARAMETER_ON
    If "PCAN_STATUS_LOG" result equals PCAN_ERROR_OK Then Then
    {
        Show Logging is active
    }
    Else
    {
        Show Error: Logging couldn't be activated
    }
}
Function Deactivatelogging
{
    Set the value PCAN_LOG_STATUS to PCAN_PARAMETER_OFF
    If "PCAN_STATUS_LOG" result equals PCAN_ERROR_OK Then Then
    {
        Show Logging is inactive
    }
    Else
    {
        Show Error: Logging couldn't be deactivated
    }
}
```

PCAN_LOG_CONFIGURE

この値は、PCAN-Basic API 内のデバッグセッションで生成されるログファイルに含まれるデバッグ情報の設定に使用されます。

Availability

バージョン 1.0.0 以降で使用できます。

Supported by

PCAN_NONEBUS : ログングパラメータはグローバルに使用されます。つまり、特定の PCAN-Channel に関連付けられていません。

Access Mode

このパラメーターは read / write です。設定して読み取ることができます。

Possible Values

このパラメータには、以下のいずれかの値、またはそれらを組み合わせた値を設定できます。

Defined Value	Description
LOG_FUNCTION_DEFAULT	この値は常にアクティブです。
LOG_FUNCTION_ENTRY	関数が入力されたときにログに記録します。
LOG_FUNCTION_PARAMETERS	関数に渡されたパラメーターをログに記録します。
LOG_FUNCTION_LEAVE	関数が終了したときとその戻り値をログに記録します。
LOG_FUNCTION_WRITE	CAN_Write 関数に渡されたパラメーターと CAN データをログに記録します。
LOG_FUNCTION_READ	CAN_Read 関数を介して受信したパラメーターと CAN データをログに記録します。

Default Value

このパラメーターの Default Value は、内部例外 (LOG_FUNCTION_DEFAULT) のみをログに記録することです。例外が発生することは非常に稀であるため、この Default Value のみを設定すると、データがまったくログに記録されない可能性があることに**注意してください** (この API のバグがないように最善を尽くします ☺)。

Initialization Status

適用されません。このパラメーターを使用するために、PCAN-Channel を初期化する必要はありません。

When to Use

特定のデバッグ情報のみが必要な場合に使用できます。

Application – Example of Use

例えば、いくつかの API 関数の呼び出し順序に問題があるアプリケーションがあり、どの関数の呼び出しが早いのか遅いのかを知りたいとします。デバッグセッションを設定して、関数の呼び出しのみをログに記録し、関数が処理される順番を確認することができます：

```

Set the value PCAN_LOG_CONFIGURE to LOG_FUNCTION_ENTRY
If "PCAN_LOG_CONFIGURE" result equals PCAN_ERROR_OK Then
{
    Set the value PCAN_LOG_STATUS to PCAN_PARAMETER_ON
    If "PCAN_LOG_STATUS" result equals PCAN_ERROR_OK Then
    {
        Do needed operation
        Set the value PCAN_LOG_STATUS to PCAN_PARAMETER_OFF
        Show Debug operation finished. Please check the log file
    }
    Else
    {
        Show Error: Logging couldn't be started
    }
}
Else
{
    Show Error: Logging cannot be configured
}

```

PCAN_LOG_TEXT

このパラメーターは、ユーザーがデバッグセッションで生成されたログファイルにカスタムテキストを挿入するのに役立ちます。

ロギング機能が非アクティブの場合、このパラメーターを使用すると、デバッグ情報の記録が自動的に開始されることに**注意してください**。通常は実行する必要のないコードの一部にこのパラメーターの呼び出しを含めることができます。これにより、不要な動作があった場合にデバッグセッションが開始されるようにすることができます（一種のウォッチドッグとして）。

Availability

バージョン 1.0.0 以降で使用できます。

Supported by

PCAN_NONEBUS : ロギングパラメータはグローバルに使用されます。つまり、特定の PCAN-Channel に関連付けられていません。

Access Mode

このパラメーターは write のみ可能です。

Possible Values

このパラメーターには、ログファイルに挿入するデータを含む文字列を指定します。文字列の長さに制限はありませんが、MAX_PATH（255 バイト）を超えない長さを使用することをお勧めします。

Default Value

適用されません。これは、write しかできない値です。

Initialization Status

適用されません。このパラメーターを使用するために、PCAN-Channel を初期化する必要はありません。

When to Use

これは、ログ機能を独自の目的で使いたい場合に使用することができます。例えば、独自のプロセスや動作をデバッグしたり、実行されたコードの場所をマークするなどです。

Application – Example of Use

アプリケーションを作成していて、その内部で実行されている他のプロセスのデバッグ情報を含めたいとします。例えば、アクセス違反が発生したときや、ユーザーが設定を変更したときなどをログに記録したいとします。独自のデバッグログを実装する代わりに、このパラメーターを使用することで、実装時間を短縮することができます。なぜなら、このログファイルは動作し、すでにテストされており、エントリがいつ行われたか、どのスレッドから行われたかなどの情報がすでに含まれているからです：

```
Function LogThisMessage(Message_to_log)
{
  Set the value PCAN_LOG_TEXT to Message_to_log
  If "PCAN_LOG_TEXT result not equals PCAN_ERROR_OK" Then
  {
    Show Error: Log couldn't be written
  }
}

..... At any part of the application .....
{
  Do "Some-Operation"
  If "Some-Operation result is OK" Then
  {
    LogThisMessage ("MyAPP: Some-Operation completed successfully")
  }
  Else
  {
    LogThisMessage ("MyAPP: Some-Operation has failed")
  }
}
```

トレースパラメータの使用

これらのパラメータは、すべての CAN 通信を任意のテキストエディタでロードできる ASCII フォーマットのファイルに記録および保存できるようにすることで、PCAN-Basic API を使用した CAN アプリケーションの開発時間とコストを最小限に抑えることを目的としています。構造化された保存データのおかげで、独自のアプリケーションでも簡単に解析することもできます（付録 B および付録 C を参照）。

Trace フォーマットはいくつかの Peak-System アプリケーションで公式に使用されているため、これらのトレースファイルを読み込んで処理できるツールがすでにいくつかあり、独自のソフトウェアプログラミングへの投資をさらに最小限に抑えています。例えば、記録された情報は PCAN-Explorer を使用して検査でき、PCAN-Trace アプリケーションを使用してシミュレーション目的で再生することもできます。

トレース機能が各 PCAN-Channel で使用可能であることを考慮してください。これは、3 つの重要なポイントを意味します。

- トレースセッションを開始する前に、PCAN-Channel を最初に初期化する必要があります。
- アプリケーション内で使用／初期化された PCAN-Channel と同じ数のトレースセッションを同時に開始することができます。
- トレースされるデータは、FD として初期化されたチャンネルの場合は関数 `CAN_ReadFD` および `CAN_WriteFD` を使用し、通常モードで初期化されたチャンネルの場合は関数 `CAN_Read` および `CAN_Write` を使用して、PCAN-Channel を介して正常に送信されたデータに対応します。アプリケーションがこれらの関数を呼び出さない場合、データはトレースされないことに**注意してください**。

PCAN_TRACE_LOCATION

この値は、PCAN-Trace ファイルが保存されるコンピューター上のフォルダを設定するために使用されます。セッションが実行されている場合（PCAN-Trace ファイルが書き込まれている場合）、PCAN_TRACE_LOCATION は、現在の PCAN-Trace ファイルを閉じ、新しいフォルダ情報でプロセスを再開するように API に指示します。

トレースファイルの名前は自由に指定できないことに**注意してください**。トレースファイルのベース名は、常に使用されている PCAN-Channel の名前です（たとえば、**PCAN_USBBUS1.trc**）。ファイルの作成日時で名前を拡張することのみが可能です。

Availability

バージョン 1.3.0 以降で使用できます。

Supported by

PCAN-ISA（チャンネル PCAN_ISABUS1 から PCAN_ISABUS8）
 PCAN-DNG（チャンネル PCAN_DNGBUS1）
 PCAN-PCI（チャンネル PCAN_PCIBUS1 から PCANPCIBUS16）
 PCAN-USB（チャンネル PCAN_USBBUS1 から PCAN_USBBUS16）
 PCAN-PCC（チャンネル PCAN_PCCBUS1 から PCAN_PCCBUS2）
 PCAN-LAN（チャンネル PCAN_LANBUS1 から PCAN_LANBUS16）

Access Mode

このパラメータは read / write です。設定して読み取ることができます。

Possible Values

この値は、実行中のコンピューター上の既存のディレクトリへの完全修飾された有効なパスを含む文字列です。デフォルトのパス（process パスの呼び出し）を使用するには、空の文字列を設定する必要があります。

Kind of Path	Value needed
CUSTOM Path	有効なディレクトリ文字列（ ファイルとパス ）。
DEFAULT Path	空の文字列（process フォルダを呼び出しています）。

Default Value

Default Value は、呼び出し元の process フォルダへのパスです。

Initialization Status

このパラメーターを使用する前に、PCAN-Channel を初期化する必要があります。

When to Use

実行中のトレースセッションを並べ替える場合に使用できます。

Application – Example of Use

さまざまなモード（flashing、diagnostic、custom、user など）で動作するアプリケーションがあるとします。モードごとにフォルダを作成して、使用するアプリケーションのモードごとにトレースファイルを自動的に分類することができます：

```
Function SetTracingPath(Current_App_Mode)
{
    According to Current_App_Mode
    Mode1: Mark DirectoryName: Mode1Dir
    Mode2: Mark DirectoryName: Mode2Dir
    ModeN: Mark DirectoryName: ModeNDir

    Create DirectoryName with Create-Directory
    Set the value PCAN_TRACE_LOCATION to DirectoryName
    If "PCAN_TRACE_LOCATION" result equals PCAN_ERROR_OK Then
    {
        Show Trace location successfully set to
        Print DirectoryName
    }
    Else
    {
        Show Error: Trace location couldn't be changed
    }
}
```

PCAN_TRACE_STATUS

このパラメーターは、ユーザーが PCAN-Basic API 内のトレースセッションのアクティビティステータスを制御するのに役立ちます。

トレースファイルの保存先パスを設定していない、または Trace モードを設定していない状態で、Trace ステータスが ON に設定されている場合、セッションプロセスは Default Value で開始されることに**注意してください**。

- PCAN-Trace ファイルは、呼び出し中のプロセスがあるフォルダに置かれます。
- 使用するファイル名は、使用する PCAN-Channel の名前です（たとえば、**PCAN_USBBUS1.trc**）。
- 既存のファイルは上書きされません。つまり、トレースプロセスの開始は失敗します。

- API は 1 つの PCAN-Trace ファイルを作成し、ファイルが **10 MB (メガバイト)** のサイズに達するまでデータを入力します。

重要な注意：トレースファイルにメッセージを書き込むためには、アプリケーションが送信のみを行っている場合でも、受信キューがアクティブに読み込まれている必要があります。送信されたメッセージは、受信キュー上でも同期されます。

Availability

バージョン 1.3.0 以降で使用できます。

Supported by

PCAN-ISA (チャンネル PCAN_ISABUS1 から PCAN_ISABUS8)

PCAN-DNG (チャンネル PCAN_DNGBUS1)

PCAN-PCI (チャンネル PCAN_PCIBUS1 から PCANPCIBUS16)

PCAN-USB (チャンネル PCAN_USBBUS1 から PCAN_USBBUS16)

PCAN-PCC (チャンネル PCAN_PCCBUS1 から PCAN_PCCBUS2)

PCAN-LAN (チャンネル PCAN_LANBUS1 から PCAN_LANBUS16)

Access Mode

このパラメーターは read / write です。設定して読み取ることができます。

Possible Values

このパラメーターは、アクティブ化 または 非アクティブ化 できます。

Defined Value	Description
PCAN_PARAMETER_OFF	トレースはオフです。
PCAN_PARAMETER_ON	トレースはオンです。

Default Value

トレースモードの Default Value は非アクティブ (PCAN_PARAMETER_OFF) です。アクティブにした後は、以下のいずれかの可能性が起こるまで、トレース機能はアクティブなままです。

- トレースセッションは明示的に非アクティブ化されています。
- 使用されている PCAN-Channel が切断されています (たとえば、関数 CAN_Uninitialize を使用)。
- トレースセッションの設定が、トレースを停止するように API に指示します (たとえば、トレースファイルの最大サイズに達しました)。

Initialization Status

このパラメーターを使用する前に、PCAN-Channel を初期化する必要があります。

When to Use

トレースセッション (start、stop、restart など) を制御するために使用できます。

Application – Example of Use

例えば、アプリケーションのユーザーが、データをいつトレースするかを決めることができるようにしたいとします。これを可能にするには、ユーザーがボタンをクリックすることで起動できる関数を使って、このパラメータを呼び出すだけです：

```
Function ActivateTracing()
{
    Set the value PCAN_TRACE_STATUS to PCAN_PARAMETER_ON
    If "PCAN_TRACE_STATUS" result equals PCAN_ERROR_OK Then
    {
        Show Trace session started successfully
    }
    Else
    {
        Show Error: Couldn't start a trace session
    }
}

Function DeactivateTracing()
{
    Set the value PCAN_TRACE_STATUS to PCAN_PARAMETER_OFF
    If "PCAN_TRACE_STATUS" result equals PCAN_ERROR_OK Then
    {
        Show Trace session finished
    }
    Else
    {
        Show Error: Couldn't stop the trace session
    }
}
```

PCAN_TRACE_SIZE

このパラメータは、単一の PCAN-Trace ファイルが持つことができる最大サイズを MB 単位で設定するために使用されます。トレースセッションがアクティブな場合、ファイルのサイズを設定しようとすると失敗することに注意してください。

Availability

バージョン 1.3.0 以降で使用できます。

Supported by

PCAN-ISA (チャンネル PCAN_ISABUS1 から PCAN_ISABUS8)

PCAN-DNG (チャンネル PCAN_DNGBUS1)

PCAN-PCI (チャンネル PCAN_PCIBUS1 から PCANPCIBUS16)

PCAN-USB (チャンネル PCAN_USBBUS1 から PCAN_USBBUS16)

PCAN-PCC (チャンネル PCAN_PCCBUS1 から PCAN_PCCBUS2)

PCAN-LAN (チャンネル PCAN_LANBUS1 から PCAN_LANBUS16)

Access Mode

このパラメータは read / write です。設定して読み取ることができます。

Possible Values

この値は、ファイルが保存できる MB 数を表す整数です。デフォルトのサイズ (10 MB) を使用するには、値 0 を設定する必要があります。

Kind of Size	Valid Value
CUSTOM Size	1～100MB の値。
DEFAULT Size	値 0 (デフォルトは 10MB) 。

Default Value

デフォルトのサイズ値は 10 MB です。これにより、約 166.000～CAN メッセージ（標準フレーム、8 データバイト）を記録できます。

Initialization Status

このパラメーターを使用する前に、PCAN-Channel を初期化する必要があります。

When to Use

1 つのファイルに保存するデータの量を制御するために使用できます。トレースの設定に応じて、このパラメーターを使用してトレースセッションを自動的に停止することができます（例：指定された制限値に達するまでデータを記録する）

Application – Example of Use

例えば、アプリケーションのユーザーがトレースの大きさを決定できるようにしたいとします。これを可能にするには、ユーザーがボタンをクリックして起動する関数を使って、このパラメーターを呼出すだけでよいのです：

```
Function SetMaximumTraceSize(Size_To_Set)
{
    Set the value PCAN_TRACE_SIZE to Size_To_Set
    If "PCAN_TRACE_SIZE" result equals PCAN_ERROR_OK Then
    {
        Show Trace size set to
        Print Size_To_Set
    }
    Else
    {
        Show Error: Couldn't configure the size for the trace file
    }
}

Function SetDefaultTraceSize()
{
    Set the value PCAN_TRACE_SIZE to 0
    If "PCAN_TRACE_SIZE" result equals PCAN_ERROR_OK Then
    {
        Show The default trace file size was successfully set
    }
    Else
    {
        Show Error: Couldn't set the default size for the trace file
    }
}
```

PCAN_TRACE_CONFIGURE

このパラメーターは、トレースプロセスとトレースセッションで生成されるファイルの設定に使用します。トレースセッションがアクティブな場合、トレースプロセスの設定を行うと失敗することに**注意してください**。

Availability

バージョン 1.3.0 以降で使用できます。

Supported by

PCAN-ISA（チャンネル PCAN_ISABUS1 から PCAN_ISABUS8）

PCAN-DNG（チャンネル PCAN_DNGBUS1）

PCAN-PCI（チャンネル PCAN_PCIBUS1 から PCANPCIBUS16）

PCAN-USB（チャンネル PCAN_USBBUS1 から PCAN_USBBUS16）

PCAN-PCC（チャンネル PCAN_PCCBUS1 から PCAN_PCCBUS2）

PCAN-LAN（チャンネル PCAN_LANBUS1 から PCAN_LANBUS16）

Access Mode

このパラメーターは read / write です。設定して読み取ることができます。

Possible Values

このパラメーターは、次のいずれかの値またはそれらの組み合わせで構成できます。

Defined Value	Description
TRACE_FILE_SINGLE	トレースセッションは単一のファイルに保存され、ファイルが設定された最大のファイルサイズに達するか、無効化されるか、使用されている PCAN-Channel が切断されるまで有効です。
TRACE_FILE_SEGMENTED	トレースセッションは複数のファイルに保存されます。新しいファイルは、前のファイルが設定された最大サイズに達すると作成されます。トレースセッションは、無効化されるか、使用されている PCAN-Channel が切断されるまで有効です。
TRACE_FILE_DATE	トレースファイルの名前には、トレースセッションの開始日も含まれます。日付は、 YYYYMMDD のフォーマットで 8 桁を使用して表されます。ここで、YYYY は年を 4 桁、MM は月を 2 桁、DD は日を 2 桁です。2013 年 2 月 28 日の "20130228_PCAN_USBBUS1.trc" 。TRACE_FILE_DATE と TRACE_FILE_TIME の両方が構成されている場合、ファイル名は常に日付 "20130228140733_PCAN_USBBUS1_1.trc" で始まります。
TRACE_FILE_TIME	トレースファイルの名前には、トレースセッションの開始時刻も含まれます。時間は、 HHMMSS のフォーマットで 6 桁を使用して表されます。ここで、HH は 24 時間形式の時間の場合は 2 桁、MM は分の場合は 2 桁、SS は秒の場合は 2 桁です。14:07:33 (02:07:33 PM) の "140733_PCAN_USBBUS1.trc" 。TRACE_FILE_DATE と TRACE_FILE_TIME の両方が構成されている場合、ファイル名は常に日付 "20130228140733_PCAN_USBBUS1_1.trc" で始まります。
TRACE_FILE_OVERWRITE	この値を設定すると、新しいトレースセッションが開始されたときに、存在するトレースファイルが上書きされてしまいます。この値が設定されていない場合、生成するファイル名が既存のファイルで使用されているものと同じであれば、トレースプロセスを開始しようとしても失敗します。

Default Value

このパラメーターの Default Value は TRACE_FILE_SINGLE で、これは単一のファイルが作成され、設定された最大ファイルサイズに達するまで記入されることを意味しています。

使用するファイルの名前は、トレースされる PCAN-Channel の名前であることに**注意してください**（例：PCAN_USBBUS1.trc）。同じ名前のファイルがすでに存在する場合、トレースセッションのアクティブ化は失敗します。

Initialization Status

このパラメーターを使用する前に、PCAN-Channel を初期化する必要があります。

When to Use

必要なトレース動作がデフォルトの動作以外の場合に使用できます。

Application – Example of Use

CAN データをトレースしたいが、何バイトをトレースするかわからない、あるいはトレース情報が許容される最大ファイルサイズ（100 MB）を超えることがわかっているとします。このような場合には、トレースプロセスで複数のファイルを使用するように設定し（セグメンテーション）、使用する保存単位のみを制限することができます。この方法では、プロセスを停止するか、ファイル作成時にエラーが発生するまで、アプリケーションは異なるファイルにデータをトレースし続けます：

```
Set the value PCAN_TARCE_SIZE to 20
If "PCAN_TARCE_SIZE result equals PCAN_ERROR_OK" Then
{
  Mark TraceConfig: TRACE_FILE_SEGMENTED Or TRACE_FILE_OVERWRITE
  Set the value PCAN_TRACE_CONFIGURE To TraceConfig
  If "PCAN_TRACE_CONFIGURE result equals PCAN_ERROR_OK" Then
  {
    Set the value PCAN_TRACE_STATUS To PCAN_PARAMETER_ON
    If "PCAN_TRACE_STATUS result equals PCAN_ERROR_OK" Then
    {
      Show Trace configured and started successfully
    }
    Else
    {
      Show Error: Couldn't start a trace session
    }
  }
  Else
  {
    Show Error: couldn't configure the trace session
  }
}
Else
{
  Show Error: couldn't configure the size of the trace file
}
```

電子回路パラメーターの使用

これらのパラメーターは、機能を凝縮することを目的としています。一部の CAN デバイスには、デジタルおよびアナログ信号用のピンが装備されており、電子回路に I/O 機能を提供することができます。このために個別の API を提供する代わりに、関数 CAN_GetValue / CAN_SetValue を介して I/O 機能にパラメーターとしてアクセスできます。

このドキュメントの執筆時点では、PCAN-Chip USB モジュールのみが、デジタル出力として設定可能な 5 つのデジタル入力ピンと 1 つのアナログ入力ピンの形式で I/O 機能を提供しています。

PCAN_IO_DIGITAL_CONFIGURATION

このパラメーターは、デバイスで使用可能な**すべての**デジタル Input / Output ピンの出力モードを設定するために使用されます。ビットマスク値として最大 32 本のピンの設定が可能です。

このドキュメントの執筆時点では、ファームウェアバージョンが 3.3.0 以上の PCAN-Chip USB ベースのデバイスのみが、最大 5 つのデジタルピンのコンフィギュレーションをサポートしていることに**注意してください**。このため、他のすべての（未使用の）ビットは、パラメーターとして渡されたビットマスク値から自動的に破棄されます。存在しないピンにビットを設定してもエラーは発生しません。

Availability

バージョン 4.3.0 以降で使用できます。

Supported by

PCAN-USB（チャネル PCAN_USBBUS1 から PCAN_USBBUS16）

Access Mode

このパラメーターは read / write です。設定して読み取ることができます。

Possible Values

このパラメーターの解像度はダブルワードであり、[0~4294967295]の範囲で値を設定できます。これは**ビットマスク**であり、すべてのビットがデジタル入力を表します。デバイスの各デジタル入力ピンは、デジタル出力として設定できます。

Bit Value	Description
0	ピンはデジタル入力として機能します。
1	ピンはデジタル入力 および 出力として機能します。

最下位ビットである Bit0 は Pin0 に、Bit1 は Pin1 に対応し、これが Pin31 まで続きます。

Default Value

各ピン（およびマスク全体）の Default Value は 0 です。これは、すべてのピンがデジタル入力のみとして設定されていることを意味します（デジタル出力はアクティブではありません）。

Initialization Status

このパラメーターを使用する前に、PCAN-Channel を初期化する必要があります。

When to Use

CAN 環境から電子回路にデジタル I/O 機能を持たせたい場合に使用できます。

Application – Example of Use

いくつかの LED を搭載した電子ユニットがあるとしましょう。デジタルピンをデジタル出力として設定して、CAN アプリケーションからそれらの LED のオン・オフを可能にできます。

PCAN_IO_DIGITAL_VALUE

このパラメーターは、デバイスで使用可能なデジタルピンによって表される出力値をビットマスク値として設定するために使用されます。PCAN_IO_DIGITAL_SET や PCAN_IO_DIGITAL_CLEAR とは異なり、この操作は全てのピンに適用されます。つまり、使用可能な各ピンは 2 つの可能な状態のいずれかに設定されます。

ビットマスク (Bit mask) では 32 本のピンの値を設定できますが、このドキュメントの執筆時点では、5 つのデジタルピンしかサポートされていないことに**注意してください**。このため、他のすべての (使用されていない) ビットは、パラメーターとして渡されたビットマスク値から自動的に破棄されます。存在しないピンにビットを設定してもエラーは発生しません。

Availability

バージョン 4.3.0 以降で使用できます。

Supported by

PCAN-USB (チャネル PCAN_USBBUS1 から PCAN_USBBUS16)

Access Mode

このパラメーターは read / write です。設定して読み取ることができます。

Possible Values

このパラメーターの解像度はダブルワードであり、[0~4294967295]の範囲で値を設定できます。これはビットマスク (Bit mask) であり、すべてのビットがデジタル出力ピンの値を表します。デバイスの各デジタルピンの値は、“Low” または “High” に設定できます。

Bit Value	Description
0	デジタルピンの値は "Low" です。
1	デジタルピンの値は "High" です。

最下位ビットである Bit0 は Pin0 に、Bit1 は Pin1 に対応し、これが Pin31 まで続きます。

Default Value

各ピン (およびマスク全体) の Default Value は 0 です。これは、すべてのデジタル出力ピンの値が “low” に設定されていることを意味します。

Initialization Status

このパラメーターを使用する前に、PCAN-Channel を初期化する必要があります。

When to Use

CAN 環境から電子回路にデジタル I/O 機能を持たせたい場合に使用できます。

Application – Example of Use

いくつかの LED を備えた電子ユニットがあるとしましょう。すべてのデジタルピンを “High” に設定して、接続されているすべての LED をオンにすることができます。

PCAN_IO_DIGITAL_SET

このパラメーターは、ビットマスク値を使用して、デバイス内で**選択した**デジタル出力ピンの値を "High" に設定するために使用されます。PCAN_IO_DIGITAL_VALUE とは異なり、必要なピンのみが "High" に設定されます。不要なものは変更されません。つまり、再設定されません。

このドキュメントの執筆時点では、ファームウェアバージョンが 3.3.0 以上の PCAN-Chip USB ベースのデバイスのみが、最大 5 つのデジタルピンのコンフィグレーションをサポートしていることに**注意してください**。このため、他のすべての（使用されていない）ビットは、パラメーターとして渡されたビットマスク値から自動的に破棄されます。存在しないピンにビットを設定してもエラーは発生しません。

Availability

バージョン 4.3.0 以降で使用できます。

Supported by

PCAN-USB（チャネル PCAN_USBBUS1 から PCAN_USBBUS16）

Access Mode

このパラメーターは write 専用です。設定のみ可能です。

Possible Values

このパラメーターの解像度はダブルワードであり、[0~4294967295]の範囲で値を設定できます。これは **Bit mask**（ビットマスクであり）、すべてのビットがデジタルピンを表します。

Bit Value	Description
0	この位置のデジタルピンは無視されます。
1	この位置のデジタルピンの値は "High" に設定されます。

最下位ビットである Bit0 は Pin0 に、Bit1 は Pin1 に対応し、これが Pin31 まで続きます。

Default Value

適用されません。

Initialization Status

このパラメーターを使用する前に、PCAN-Channel を初期化する必要があります。

When to Use

CAN 環境から電子回路にデジタル I/O 機能を持たせたい場合に使用できます。

Application – Example of Use

5 つの LED を搭載した電子ユニットがあるとします。あなたはすでにそれらのうちの 4 つをオンにしています。このパラメーターを使用すると、他の 4 つの LED を再度設定しなくても、最後のパラメーターをオンにすることができます。

PCAN_IO_DIGITAL_CLEAR

このパラメーターは、ビットマスク値を使用して、デバイス内で**選択された**デジタル出力ピンの値を“Low”に設定するために使用されます。PCAN_IO_DIGITAL_VALUE とは異なり、必要なピンのみが“Low”に設定されます。不要なものは変更されません。つまり、再設定されません。

このドキュメントの執筆時点では、ファームウェアバージョンが 3.3.0 以上の PCAN-Chip USB ベースのデバイスのみが、最大 5 つのデジタルピンのコンフィグレーションをサポートしていることに**注意してください**。このため、他のすべての（使用されていない）ビットは、パラメーターとして渡されたビットマスク値から自動的に破棄されます。存在しないピンにビットを設定してもエラーは発生しません。

Availability

バージョン 4.3.0 以降で使用できます。

Supported by

PCAN-USB（チャネル PCAN_USBBUS1 から PCAN_USBBUS16）

Access Mode

このパラメーターは write 専用です。設定のみ可能です。

Possible Values

このパラメーターの解像度はダブルワードであり、[0～4294967295]の範囲で値を設定できます。これはビットマスクであり、すべてのビットがデジタルピンを表します。

Bit Value	Description
0	この位置のデジタルピンは無視されます。
1	この位置のデジタルピンの値は "Low" に設定されます。

最下位ビットである Bit0 は Pin0 に、Bit1 は Pin1 に対応し、これが Pin31 まで続きます。

Default Value

適用されません。

Initialization Status

このパラメーターを使用する前に、PCAN-Channel を初期化する必要があります。

When to Use

CAN 環境から電子回路にデジタル I/O 機能を持たせたい場合に使用できます。

Application – Example of Use

5 つの LED を備えた電子ユニットがあるとします。あなたはすでにそれらのうちの 3 つをオンにしています。このパラメーターを使用すると、残りの 4 つを新たにに設定しなくても、そのうちの 1 つをオフにすることができ、それぞれ 2 つの LED がオン、2 つの LED がオフにすることができます。

PCAN_IO_ANALOG_VALUE

このパラメーターは、デバイスのアナログ入力ピンからアナログ電圧を読み取るために使用されます。

このドキュメントの執筆時点では、ファームウェアバージョンが 3.3.0 以上の PCAN-Chip USB ベースのデバイスのみが、1つのアナログピンのみをサポートしていることに**注意してください**。

Availability

バージョン 4.3.0 以降で使用できます。

Supported by

PCAN-USB (チャネル PCAN_USBBUS1 から PCAN_USBBUS16)

Access Mode

このパラメーターは read 専用です。書くことはできません。

Possible Values

このパラメーターの解像度はダブルワードであり、[0~4294967295]の範囲で値を設定できます。戻り値は、符号なし整数値である A/D コンバータからの直接値を表します。戻り値は、外部配線を考慮して符号付きの値に変換する必要があります。

Default Value

適用されません。

Initialization Status

このパラメーターを使用する前に、PCAN-Channel を初期化する必要があります。

When to Use

CAN 環境から電子回路にデジタル I/O 機能を持たせたい場合に使用できます。

Application – Example of Use

ポテンシオメータ付きの電子ユニットがあるとしましょう。その状態を読み取って、計算された値をアプリケーションに表示したり、現在の値に応じて何らかの判断を下すことができます。

付録 A : レジストリのデバッグログ

ここでは、Windows のレジストリを使用して PCAN-Basic のログ機能をアクティブ化／非アクティブ化する方法について説明します。

Log Session のアクティブ化

1. PCAN-Basic を使用してすべてのアプリケーションを停止します。
2. Windows のレジストリを開きます（たとえば、Windows の Start（スタート）メニュー／“Execute...” を使用して “regedit” と入力します）。
3. [HKEY_CURRENT_USER] ハイブの下に次のレジストリキーを作成します。
¥ Software ¥ PEAK-System ¥ PCAN-Basic ¥ Log
4. ログに記録するデータを指定するには、前に作成したキーに新しい DWORD 値を追加し、名前を “Flags” とします。
5. 必要に応じて “Flags” の値を設定します。この値は、任意の LOG_FUNCTION_* 定義の数値、またはそれらの論理 OR の組み合わせの数値です。
6. ログファイルを作成するディレクトリを指定するには、前に作成したキーに新しい STRING 値を追加し、それを “Path” と呼びます。
7. “Path” の値を目的のディレクトリへのフルパスで設定します。

この時点で、PCAN-Basic API を使用するアプリケーションを起動すると、デバッグセッションが自動的に生成されます。

Log Session の非アクティブ化

1. PCAN-Basic を使用しているすべてのアプリケーションを停止します。
2. Windows のレジストリを開きます（たとえば、Windows の [スタート] メニュー／[実行...] を使用して “regedit” と入力します）。
3. レジストリハイブ [HKEY_CURRENT_USER] を検索します。
4. 次のレジストリキーを検索します。
¥ Software ¥ PEAK-System ¥ PCAN-Basic ¥ Log
5. 該当キーおよびその値を削除します。

この時点で、PCAN-Basic を使用するアプリケーションを起動しても、ロギング操作は行われなくなります。

非常に重要な注意

デバッグセッションが終了したら、作成したキーを削除することを忘れないでください。 キーを残しておくと、Windows アカウントで動作しているすべての PCAN-Basic アプリケーションがログファイルにデータを書き込んだままになり、ハードディスクの容量を無駄に消費する巨大なテキストファイルが生成されてしまいます。

付録 B : PCAN Trace Format 1.1

PCAN-Basic API は、PCAN-Explorer 3.0.2、PCAN-Explorer 4、PCAN-Trace 1.5、PCAN-View 3、および Peak-Converter 1 で使用されている、通常の CAN (non FD) を備えたチャネル用の PCAN-Trace フォーマット 1.1 を使用します。

このフォーマットは、“normal mode”(ノーマルモード)で初期化されたチャネルに使用されます。つまり、CAN_Initialize 関数を使用して初期化されたチャネルは、CAN_Read および CAN_Write 関数を介して通信を行います。

例

```

; $FILEVERSION=1.1
; $STARTTIME=37704.5364870833
;
; C:\TraceFile.trc
;
; Start time: 24.03.2003 12:52:32.484
; PCAN-Net: TestNet
;
; Columns description:
; ~~~~~
; +-Message Number
; |               |
; |               | +Time Offset (ms)
; |               | |
; |               | +Type
; |               | |
; |               | +ID (hex)
; |               | |
; |               | +Data Length Code
; |               | |
; |               | +Data Bytes (hex) ...
; |               | |
; +-----+-----+-----+-----+-----+-----+
; 1)      1059.9 Rx      0300 7 00 00 00 00 04 00 00
; 2)      1283.2 Rx      0300 7 00 00 00 00 04 00 00
; 3)      1298.9 Tx      0400 2 00 00
; 4)      1323.0 Rx      0300 7 00 00 00 00 06 00 00
; 5)      1346.8 Warnng FFFFFFFF 4 00 00 00 04 BUSLIGHT
; 6)      1349.2 Error   0008 4 00 19 08 08

```

説明

ファイルコーディング :

トレースファイルは ASCII コードです。

コメント行 :

セミコロンで始まる行は “Comments” (コメント) であり、\$-Keywords を除いて、Trace (トレース) ファイルのロード中は無視されます。

\$-Keywords :

これらは、Trace (トレース) ファイルに関するさまざまな情報を提供する定義済みの情報です。コメント行として表示されます。可能なキーワードは次のとおりです。

- \$ FILEVERSION : ファイルフォーマットのメジャーバージョンとマイナーバージョン、つまりこのバージョンの “1.1” が含まれます。
- \$ STARTTIME : トレースファイルの絶対的な開始時刻が含まれます。
 - 形式 : 浮動小数点、小数点としてのポイント。
 - 値 : 整数部分は、1899 年 12 月 30 日から経過した日数を表します。小数部分 (24 時間経過した日の一部) の分解能は 1 ms (ミリ秒) です。

Columns :

トレースファイルに含まれる情報は、5つのカラムに分類されます。

- メッセージ番号：記録されたメッセージのインデックス（トレースファイルのロード中は無視されます）。
- タイムオフセット（ms）：トレースセッションの開始からのタイムオフセット。時間の分解能は 1/10 ミリ秒です。
 - 形式：浮動小数点、小数点としてのポイント。
 - 値：整数部分はミリ秒オフセットを表します。小数部分は 1/10 ms（1桁）です。
- タイプ：記録されたメッセージの種類を表します。可能なメッセージタイプは次のとおりです。
 - "Rx"：メッセージを受信しました（PCAN-Basic では、関数 CAN_Read を使用）。
 - "Tx"：メッセージが送信されました（PCAN-Basic では、関数 CAN_Write を使用）。
 - "Warn"：メッセージは受信した Warning-Frame（警告フレーム）を表します。
 - "Error"：メッセージは Error-Frame（エラーフレーム）を表します。
- ID（hex）：CAN-ID を 16 進表記で表します。Possible Values は次のとおりです。
 - 11 ビット CAN-ID（0000-07FF）の場合は 4 桁。
 - 29 ビット CAN-ID の場合は 8 桁（00000000-1FFFFFFF）。
 - 特殊なケース：警告フレームの "FFFFFFFF"。
- データ長コード：記録されたメッセージに含まれるデータの量を表す 0～8 の数値です。
- データバイト（16 進数）：録音されたメッセージのデータを表します。メッセージタイプに応じて、データは次のようになります。
 - メッセージが一般的な CAN データを表す場合：データ長コードが示すように、16 進表記の非常に多くのデータバイト。
 - メッセージがリモートリクエストフレームを表す場合："RTR"
 - メッセージが警告フレームを表す場合：モトローラ形式を使用して、16 進表記で表された 4 データバイト。この行の終わりに、警告の短い名前（トレースファイルのロード中は無視されます）。
例："00 00 0004BUSLIGHT"。
 - メッセージがエラーフレームを表す場合：16 進表記で表された 4 データバイト。

付録 C : PCAN Trace Format 2.0

PCAN-Basic API は、PCAN-View 4、PEAK-Converter 2、および PCAN-Explorer 6 で使用されている、FD 機能を持つチャンネル（CAN-FD）用の PCAN-Trace フォーマット 2.0 を使用します。このフォーマットは初期化されたチャンネルに使用されます。“FD モード”では、関数 `CAN_InitializeFD` を使用して初期化されたチャンネルであり、関数 `CAN_ReadFD` および `CAN_WriteFD` を介して通信を行います。

例

```

; $FILEVERSION=2.0
; $STARTTIME=41736.3716038773
; $COLUMNS=N,O,T,B,I,d,R,l,D
;
; C:\TraceFile.trc
; Start time: 07.04.2014 08:55:06.575.0
; Generated by PCAN-View v4.0.7.340
;-----
; Bus Name Connection Protocol
; 1 Connection1 TestNet@pcan_usb CAN
;-----
; Message Time Type ID Rx/Tx
; Number Offset | Bus ID [hex] | Reserved
; | | [ms] | | | | Data Length
; | | | | | | | Data [hex] ...
; | | | | | | | |
;-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
; 1 1059.900 DT 1 0300 Rx - 7 00 00 00 00 04 00 00
; 2 1283.231 DT 1 0300 Rx - 7 00 00 00 00 04 00 00
; 3 1298.945 DT 1 0400 Tx - 2 00 00
; 4 1323.201 DT 1 0300 Rx - 7 00 00 00 00 06 00 00
; 5 1334.416 FD 1 0500 Tx - 12 01 02 03 04 05 06 07 08 09 0A 0B 0C
; 6 1334.222 ER 1 Rx - 04 00 02 00 00
; 7 1334.224 EV User-defined event text
; 8 1334.231 ST 1 Rx - 00 00 00 08
; 9 1334.268 ER 1 Rx - 04 00 02 08 00

```

說明

ファイルコーディング:

トレースファイルは ASCII コードです。

コメント行：

セミコロンで始まる行は“Comments”（コメント）であり、\$-Keywords を除いて、Trace（トレース）ファイルのロード中は無視されます。

\$-Keywords :

これらは、Trace（トレース）ファイルに関するさまざまな情報を提供する定義済みの情報です。コメント行として表示されます。可能なキーワードは次のとおりです。

- \$ FILEVERSION : ファイルフォーマットのメジャーバージョンとマイナーバージョン、つまりこのバージョンの "2.0" が含まれます。
- \$ STARTIME : トレースファイルの絶対的な開始時刻が含まれます。
 - 形式 : 浮動小数点、小数点としてのポイント。
 - 値 : 整数部分は、1899 年 12 月 30 日から経過した日数を表します。小数部分 (24 時間経過した日の一部) の分解能は 1 ms (ミリ秒) です。
- \$ COLUMNS : トレースファイルを含む列を表します。列の順序は変更できません。ただし、一部の列はオプションです。必須の順序は次のとおりです
(オプションの列は角括弧で囲まれています) : [N]、O、[B]、T、I、d、[B]、l / L、D。

Columns :

トレースファイルに含まれる情報は 10 列に収められていますが、一部はオプションです。

- N : メッセージ番号、録音されたメッセージのインデックス。オプション。
- O : トレースの開始からの時間オフセット。解像度 : 1 μ s。
小数点の前の値は ms を表します。小数点の後ろの値は μ s (3 桁) を表します。
- B : Bus (1-16)。 オプション。
- T : メッセージの時間 :
 - DT : CAN または J1939 データフレーム。
 - FD : CANFD データフレーム。
 - FB : BRS ビットが設定された CAN FD データフレーム (Bit Rate Switch)。
 - FE : ESI ビットが設定された CAN FD データフレーム (Error State Indicator)。
 - BI : BRS と ESI の両方のビットが設定された CAN FD データフレーム。
 - RR : リモートリクエスト (Remote Request) フレーム。
 - ST : ハードウェアステータスの変更。
 - ER : エラーフレーム。
 - EV : イベント。ユーザー定義のテキスト。2 桁のタイプインジケータの直後から始まります。
- I : CAN-ID (16 進数) :
 - 11 ビット CAN-ID (0000-07FF) の場合は 4 桁。
 - 29 ビット CAN-ID の場合は 8 桁 (00000000-1FFFFFFF)。
- d : Direction : メッセージが受信されたか ('Rx')、送信されたか ('Tx') を示します。
- R : Reserved (予約済み)。J1939 プロトコルにのみ使用されます。CAN バスの '-' が含まれています。J1939 プロトコルの場合、トランスポートプロトコル PDU2-large メッセージの宛先アドレスが含まれます。CAN または CAN FD フレームのみを含むファイルの場合はオプション。
- l : データ長 (0-1785)。これはデータバイトの実数であり、データ長コード (0..15) ではありません。オプション。省略した場合は、データ長コード列 ('l') を含める必要があります。
- L : データ長コード (0~15)。オプション。省略した場合は、データ長 ('l') を含める必要があります。
- D : データ。16 進表記の 0-1785 データバイト。

付録 D : アクセプタンス・コードとアクセプタンス・マスクの計算

アクセプタンス・フィルタは、アクセプタンス・コードとアクセプタンス・マスクで構成されています。これらの値は、アプリケーションのニーズに応じて、11 ビットのアクセプタンス・フィルタ（パラメーター PCAN_ACCEPTANCE_FILTER_11BIT を使用）または 29 ビットのアクセプタンス・フィルタ（パラメーター PCAN_ACCEPTANCE_FILTER_29BIT を使用）を設定するために使用されます。ID が 11 ビットであるか 29 ビットであるかに関係なく、コード値とマスク値の計算方法は同じです。

PCAN-Hardware のフィルタリングは SJA1000 CAN コントローラーに基づいていることを考慮してください。SJA1000 CAN コントローラーは、standard ID（11 ビット）と extended ID（29 ビット）の両方に 1 つのアクセプタンス・フィルタしか使用しないことを考慮してください。11 ビットと 29 ビットのフィルタを混在させることはお勧めできません。

例えば、standard ID (11 ビット) の 101h、401h、501h に対するアクセプタンス・フィルタを計算してみます：

コード

アクセプタンス・コードは、受信を希望するすべての ID の間で論理的な AND 演算を行った結果の値です。

	15				0
101h:	0000	0001	0000	0001	b
401h:	0000	0100	0000	0001	b
501h:	0000	0101	0000	0001	b
----- AND					
	0000	0000	0000	0001	=> Code=1h

マスク

アクセプタンス・マスクは、受信を希望するすべての ID の間に論理的排他的論理和（XOR）を適用した結果として得られる値です。つまり、受信を希望する ID 内の 2 つのビットの間に 1 つだけ違いがあれば、XOR 条件を満たし、そのビットを “don't care bit” としてマークすることができます（“don't care bit” の値は ‘1’ です）。

	15				0
101h:	0000	0001	0000	0001	b
401h:	0000	0100	0000	0001	b
501h:	0000	0101	0000	0001	b
----- XOR					
	0000	0101	0000	0000	=> Mask=500h

なお、アクセプタンス・フィルタを使用した場合でも、不要なメッセージを受信する可能性があります。たとえば、上記の例では、標準の ID 1h も受信してしまいます。

SJA1000 のアクセプタンス・フィルタについての詳細は、[SJA1000 の仕様書](#)に記載されています。