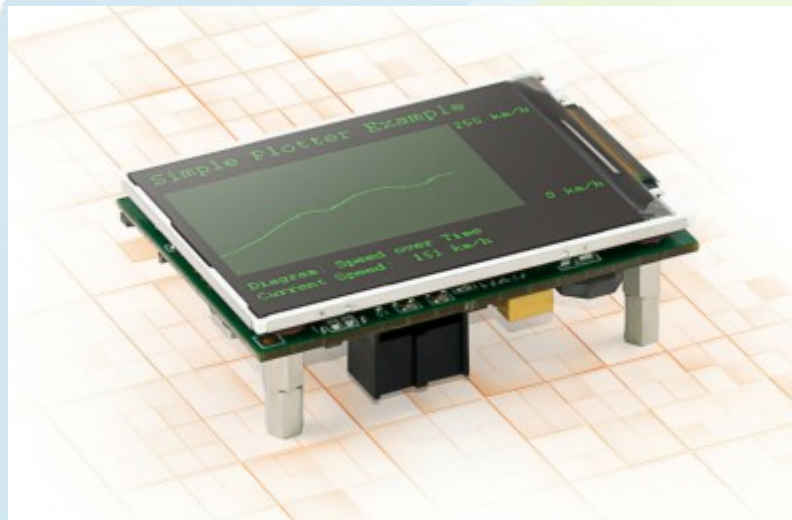


PCAN-MiniDisplay

시각화 및 차량 데이터의 기록

사용자 설명서



Products taken into account

Product name	Model	Firmware	Part number
PCAN-MiniDisplay	Display with three mounting pegs	V1.1.x	IPEH-002262
PCAN-MiniDisplay with casing	Display in casing with push buttons		IPEH-002262-KSM1

The front page shows the PCAN-MiniDisplay with three mounting pegs.

Document version 1.2.2 (2015-08-19)

이 문서는 기본적으로 제공되는 영문 매뉴얼을 해석한 것입니다.
내용 중 잘못 해석된 것이 있을 수 있습니다.
자세한 것은 영문 매뉴얼을 참조하십시오.

Contents

1 소개	5
1.1 속성 요약	5
1.2 공급 범위	6
2 커넥터	7
2.1 J1 외부 버튼	7
2.2 J2 System	8
2.3 J3 USB	8
2.3.1 USB 연결 제거	9
2.3.2 PCAN-MiniDisplay 기능에 대한 제한	9
2.4 J4 JTAG	10
2.5 J5 microSD	10
2.6 J6 Serial	11
3 Scene 정의	12
3.1 *.ins 파일의 일반 개요(설명)	12
3.2 *.ins 파일의 예	13
3.3 여러 Scenes 사용	15
4 작업 단계	16
4.1 전제 조건	16
4.2 메모리 카드 준비	16
4.3 장치 전원 켜기	17
4.4 메뉴	17
4.4.1 상태 표시	17
4.5 Startup Options 설정	19
5 CAN Tracing	20
5.1 CAN 트래픽 기록	20
5.2 기록된 CAN 트래픽 재생(Playing Back)	21
5.3 기록된 CAN 트래픽을 PC에서 사용하기	22
6 CAN 트래픽 필터링	24
7 설정 및 유지 보수에 대한 메뉴 참조	25
7.1 장치 설정	25
7.1.1 Ch. 1/2 CAN 비트 레이트 검출	25
7.1.2 Ch. 1/2 CAN 비트 레이트	25
7.1.3 Ch. 1/2 Listen-only 모드	26
7.1.4 Ch. 1/2 필터	26
7.1.5 Ch. 2 Transceiver 모드	26
7.1.6 User CAN 비트 레이트	26
7.1.7 신호음	27
7.1.8 Show startscreen	27
7.1.9 날짜와 시간	27
7.1.10 파일 인덱스 초기화	27
7.2 Internal Statistics(내부 통계)	28
7.3 메모리 카드	28
8 기술 사양	30

부록 A CE 인증	31
부록 B 도면 치수	32
부록 C 메모리 카드에 있는 파일 구조	33
부록 D 정의 파일 참조	34
D.1 Scene 정의 (*.ins)	34
D.1.1 [global] 섹션	34
D.1.2 [instrumentX] 섹션	36
D.1.1 [fontX] 섹션	39
D.1.1 [variableX] 섹션	40
D.1.1 [labelX] 섹션	44
D.1.1 [plotterX] 섹션	47
D.2 Scenes 프로젝트 파일 (*.inp)	49
D.2.1 [global] 섹션	49
D.2.1 [scenes] 섹션	49
D.3 CAN ID 필터 (*.flt)	50
D.3.1 필터 파일 형식 설명	50
D.3.2 필터 파일 예	51
부록 E 케이스로 된 모델	52
E.1 공급 범위	52
E.2 푸시 버튼	53
E.3 커넥터	54
E.4 miniSD 카드 슬롯	55

1 소개

PCAN-MiniDisplay는 CAN 데이터의 시각화를 위한 인간-머신 인터페이스로서 이용됩니다. CAN 버스에 연결의 경우, High-speed 및 single-wire CAN 컨트롤러입니다. 들어오는 CAN 데이터의 그래픽 표현은 장치를 사용하기 전에 구성된 후 TFT display 에서 수행됩니다.

CAN 데이터의 표시 외에, PCAN-MiniDisplay는 데이터 기록에 사용될 수 있습니다. 데이터 트래픽은 메모리 카드에 기록되고, 나중에 PC에서 CAN 버스로 재생할 뿐만 아니라 분석될 수 있습니다.

PCAN-MiniDisplay board 장착에 사용되는 푸시 버튼이 있는 플라스틱 케이스 버전이 있습니다. 이것은 애플리케이션-특정 구성을 생성하고 평가하는데 사용될 수 있습니다. 또한, 사용자는 작업 중에 서로 다른 구성을 전환할 수 있습니다.

이 사용자 설명서는 장착에 사용되는 PCAN-MiniDisplay 보드를 의미합니다. 케이스와 버튼(IPEH-002262-KSM1) 모델의 특수성은 52 페이지의 부록 E에 설명되어 있습니다.

참고: 이 매뉴얼은 펌웨어 버전 1.1.x 로 작동되는 장치입니다.

1.1 속성 요약

- High-speed CAN channel (ISO 11898-2) 및 Single-wire CAN channel (SAE J2411)을 통해 CAN 연결
- 320 x 240 pixel 해상도를 갖는 TFT display
- microSD memory card (max. 32 Gbyte)를 위한 Slot, 512MByte card 포함.
- PC를 사용하여 메모리 카드에 액세스하기 위한 High-speed USB 2.0 port
- CAN 데이터를 자유롭게 구성할 수 있는 시각화
- 메모리 카드에서 실행 구성
- 푸시 버튼을 통해 다중 디스플레이 구성을 전환(선택사항)
- 기본 이름을 사용하여 구성 옵션 자동 시작
- 메모리 카드에 들어오는 CAN 메시지 기록.
- 윈도우 소프트웨어를 사용하여 다양한 출력 포맷

- trace 파일 Playback
- CAN 채널당 CAN ID 필터링
- CAN을 통해 Wake-up
- 크기 : 70 x 50 mm
- 공급 전원 7 ~ 30 VDC
- 작동 온도 범위 -20 ~ 70 °C (-4 ~ 158 °F)

1.2 공급 범위

- PCAN-MiniDisplay
- 주변 장치(전원, 필드 버스, 디지털 입력)를 위한 끝이 오픈된 연결 케이블 어셈블리(50 cm 길이)
- open wire 연결 케이블 어셈블리(50 cm 길이)
- microSD memory card (512 Mbyte)
- PDF 형식으로 된 User Manual

케이스와 버튼으로 된 모델(IPEN-002262)의 특정 기능은 52 페이지에 있는 부록 E에 설명되어 있습니다.

2 커넥터

회로 보드의 커넥터는 다음과 같습니다:

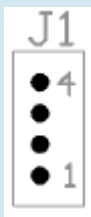
- J1 External Buttons (2.1 below)
- J2 System (2.2 on page 8)
- J3 USB (2.3 on page 8)
- J4 JTAG (2.4 on page 10)
- J5 microSD (2.5 on page 10)
- J6 Serial (2.6 on page 11)

기본 사용을 위해, 적어도 전력 공급시 (J2), CAN 연결(J2) 및 microSD 메모리 카드(J5)가 필요합니다.

2.1 J1 외부 버튼

3개 푸시 버튼은 PCAN-MiniDisplay 사용자 인터페이스 메뉴를 탐색하는데 사용되고 연결하는데 사용될 수 있습니다. 이것은 유지 보수 및 디버깅 목적을 위한 것입니다.

예를 들어, PCAN-MiniDisplay의 사용자는 통신 설정을 변경하거나 CAN tracing을 수행하는 것을 가능하게 할 수 있습니다.



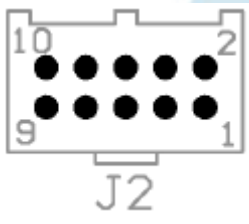
외부 버튼 핀 배열

Pin	Function
1	Din1: Down ↓
2	Din2: Up ↑
3	Din3: Enter ↵
4	GND

3개 푸시 버튼은 GND에 각 기능 핀(1~3)을 연결합니다.

2.2 J2 System

- Power supply
- 1 High-speed CAN channel (CAN1), 1 Single-wire CAN channel(CAN2), CAN2 외부 트랜시버의 대체 사용
- 디지털 출력 1개, 디지털 입력 2개, 아날로그 입력 2개



System 핀 배열

핀	기능
1	Vb (7 - 30 V DC)
2	GND
3	Dout1
4	Din4
5	Ain1
6	Ain2
7	Not connected or CAN2-RxD ¹
8	CAN2_SW or CAN2-TxD ¹
9	CAN1_L
10	CAN1_H

2.3 J3 USB

USB 연결을 통해 PC에서 PCAN-MiniDisplay 메모리 카드에 액세스할 수 있습니다. PCAN-MiniDisplay가 스위치 오프되는 경우, 메모리 카드는 액세스될 수 없습니다. 장치의 시작 시에, USB 연결이 잠시 중단됩니다.

1 외부 CAN 트랜시버 옵션을 위한 TTL 신호



USB 핀 배열

핀	기능
1	5 V USB
2	D-
3	D+
4	GND
5	GND

2.3.1 USB 연결 제거

PC 나 PCAN-MiniDisplay에서 USB 케이블을 분리하기 전에, 장치가 운영 체제에서 로그 아웃해야 합니다. 이 절차는 운영 체제가 올바르게 PCAN-MiniDisplay의 메모리 카드에 입력 처리를 완료한 것을 보장합니다.



참: 작업 표시 줄의 **Safely remove hardware** 아이콘

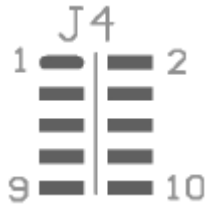
2.3.2 PCAN-MiniDisplay 기능에 대한 제한

장치가 연결된 PC와 동시에 메모리 카드에 액세스할 수 있기 때문에, PC에 USB 연결동안 PCAN-MiniDisplay의 일부 기능은 제한됩니다:

- Scene 정의(instruments)를 로드할 수 없습니다.
- 기록 및 CAN 트래픽(trace)의 재생이 작동하지 않습니다.
- **Memory Card** 메뉴의 명령이 작동하지 않습니다.

2.4 J4 JTAG

디버깅을 위해 마이크로컨트롤러에 액세스할 수 있습니다.



JTAG 핀 배열

핀	기능
1	GND
2	GND
3	Reset#
4	3.3 V
5	TCK
6	TMS
7	TDO
8	TDI
9	RTCK
10	TRST

2.5 J5 microSD

MicroSD 메모리 카드를 위한 슬롯(최대 32 GByte).



PCAN-MiniDisplay의 회로 보드의 microSD 카드 슬롯의 위치(아래 보기)

2.6 J6 Serial

Customer-specific 확장을 위한 직렬 신호를 제공합니다.



Serial 핀 할당

핀	기능
1	GND
2	5 V
3	TxD RS-232
4	RxD RS-232
5	SCL (Software I2C)
6	SDA (Software I2C)

3 Scene 정의

PCAN-MiniDisplay는 그래픽 형태로 CAN 데이터의 표시를 위한 scene를 사용합니다.

3.1 *.ins 파일의 일반 개요(설명)

Scene은 *.ins 확장자를 가진 텍스트 파일에 instruments, labels, fonts, 등이 정의되어 있습니다. Scenes 또한 비트 맵 그래픽 파일(*.bmp) 및 글꼴 파일(*.fon)을 포함합니다.

scene 정의 파일의 내용은 각각 정의 항목을 포함하는 다른 섹션으로 분할됩니다.

- [global]: scene에 대한 일반적인 정의
- [instrumentX]: 사용된 비트맵
- [labelX]: 텍스트 기반 라벨
- [variableX]: 라벨, 플로터, 및 instrument에 사용되는 CAN 변수
- [fontX]: 라벨에 사용되는 글꼴
- [plotterX]: line writer 도표

키워드에 대한 참조는 34 페이지의 부록 섹션 D.1에 위치하고 있습니다.
예제 파일은 다음과 같습니다.

정의 파일은 더블 슬래시(//)에 의해 인도되는 주석을 포함할 수 있습니다.

*.ins 파일이 PCAN-MiniDisplay에 의해 해석된 바와 같이 이중 슬래시 후에 라인의 모든 문자는 무시됩니다.

load time 동안, PCAN-MiniDisplay가 화면 정의 (*.inb)의 바이너리 버전을 생성합니다.

패널 내부의 모든 변환이 이미 포함되어 있기 때문에, 나중에 사용하기에 훨씬 더 빨리 로드됩니다.
이것은 기본 동작이지만, Global 섹션에서 구성할 수 있습니다.

Scene의 정의를 변경하는 것은 단지 텍스트 기반의 파일로 수행될 수 있습니다.

참고: scene의 정의가 변경되면, 이전에 생성된 이진 파일(*.inb)을 수동으로 PCAN-MiniDisplay의 메모리 카드에서 삭제해야 합니다, 그래서 로더는 새로운 이진 텍스트 파일에서 생성되어야 함을 알 수 있습니다. 이진 파일을 삭제하지 않을 경우, 정의 파일의 변경 사항이 적용되지 않습니다.

3.2 *.ins 파일의 예

참고: 텍스트 파일 (*.ins)은 끝에서 빈 줄이 있어야 합니다.

```
[global]
format=1
instruments=2
variables=1
labels=1
flashing_time=300
compress=0

[instrument1]
pictures=1
name="Background"
no_restore=1
0="Background_320_240.bmp", "", 0,0,0,0

[instrument2]
pictures=2
name="turn_left"
no_restore=0
offset=0,0
var_name="clig"
0="empty_28_20.bmp", "", 0,0,0,0
1="turn_l_28_20.bmp", "", 0,0,1,1

[variable1]
name="oiltemperature"
canid=779 // ID in decimal notation
position=8,8
frametype=0
byteorder=0
scale=0.62745
offset=-10
vartype=0
datatype=1

[label1]
name="Oil Temperature"
position=0,106 // lucida_13_18
font_idx=3
length=5
```

```
fmt_string="% 4d°"           // Special treatment for degree character
                             // "°" (ASCII 127)

initval="----°"
fontcolor=200,200,0         // green
bgcolor=0,0,0              // black

sector=1
range=-10,124
fontcolor=200,200,0
bgcolor=0,0,0
flashing=0

sector=1
range=125,150
fontcolor=255,0,0
offcolor=127,0,0
bgcolor=0,0,0
flashing=1
```

3.3 여러 Scenes 사용

PCAN-MiniDisplay에서 하나 이상의 scene을 사용하는 것이 가능합니다.
Scene은 디지털 입력(푸시 버튼 등)을 사용하여 전환할 수 있습니다.
이를 달성하기 위해, scene 정의 파일 목록은 scene 프로젝트 파일(*.inp)에서 생성되어야 합니다.
이 목록은 startup에서 사용되는 scene을 정의합니다.

키워드에 대한 참조는 49 페이지의 부록 섹션 D.2에 위치하고 있습니다.
예제 파일은 다음과 같습니다.

참고: 텍스트 파일 (*.ins)은 끝에서 빈 줄이 있어야 합니다.

```
[general]
version=1

[scenes]
scenes=4
startscene=2
1="speedo0.ins"
2="speedo1.ins"
3="picture1.ins"
4="picture2.ins"
```

4 작업 단계

PCAN-MiniDisplay에 전원을 입력하면 내부에 저장된 startup options를 찾습니다. Display를 위한 scene, CAN 기록을 위한 tracer, 또는 모두가 자동으로 시작될 수 있습니다.

팁: 납품된 경우, PCAN-MiniDisplay는 Default.inp 라는 scene을 시작하도록 구성되어 있습니다. 따라서, 이 장치는 푸시 버튼을 통해 초기 사용자 상호 작용없이 표시될 수 있습니다.

이 장에서는 PCAN-MiniDisplay를 동작하는 방법을 단계별로 설명합니다. 다음의 각 섹션을 통해 이동하시기 바랍니다.

4.1 전제 조건

- microSD 메모리 카드, 최대 32기가 바이트(공급되지 않음)
- SD 카드 리더가 장착된 PC (microSD 카드 용 어댑터 포함)
또는
PCAN-MiniDisplay에 PC USB 연결(J3 에 연결)
- 전원 공급 장치는 7~30V DC(J2에 연결)

4.2 메모리 카드 준비

장치가 올바르게 시작되도록 상기 메모리 카드는 특정 디렉토리 구조를 포함해야 합니다. 이것은 납품된 제품에 존재합니다. 디렉토리 구조에 자신의 scene을 추가할 수 있습니다.

⇒ 메모리 카드를 준비하려면 다음을 수행하십시오:

scene 메모리 카드를 준비하려면 다음과 같이 하십시오:

1. PC에 microSD 메모리 카드에 액세스하기 위해서 PCAN-MiniDisplay와 PC 사이의 USB 연결을 합니다. 다른 방법으로 PCAN-MiniDisplay로부터 카드를 제거하고, PC에 연결된 카드 판독기(card reader)의 슬롯에 삽입합니다. 메모리 카드는 PC 운영체제에서 별도의 대용량 저장 장치로 나타납니다.
2. 이전에 만든 하나 또는 여러 scene(12 페이지에 있는 '3장 Scene 정의'을 참조) 메모리 카드의 다음 디렉토리에 복사합니다: /MiniDisplay/Scenes/ 각 scene는 자신의 서브 디렉토리가 있어야 합니다. 납품될 때, PCAN-MiniDisplay는 자동으로 전원을 켤 때 다음 scene을 표시하도록 구성되어 있습니다: /MiniDisplay/Scenes/Default/Default.inp
3. PC의 운영체제에서 USB 연결을 해제하고 USB 케이블을 분리합니다. 만약 PC에서 카드 판독기를 사용한 경우, microSD 카드를 제거하고 PCAN-MiniDisplay의 슬롯에 삽입합니다.

⇒ 필요한 경우(예를 들어, 새 카드를 구현하는 방법 또는 디렉토리 구조가 엉망되면), 카드의 디렉토리 구조를 복원하려면 다음을 수행합니다:

1. PC에서 microSD 메모리 카드에 액세스하기 위해서 PCAN-MiniDisplay와 PC 사이의 USB 연결을 합니다. 다른 방법으로 PCAN-MiniDisplay로부터 카드를 제거하고, PC에 연결된 카드 판독기의 슬롯에 삽입합니다. 메모리 카드는 PC 운영체제에서 별도의 대용량 저장 장치로 나타납니다.
2. 운영체제에서 FAT32 파일 시스템으로 메모리 카드를 포맷합니다.
3. 제공된 DVD에서, 메모리 카드의 루트 디렉토리에 대한 모든 서브 디렉토리 및 파일을 포함하는 MiniDisplay 디렉토리를 복사합니다. DVD에서 MiniDisplay 디렉토리는 아래에 있습니다:
/Tools/PCAN-MiniDisplay/
추가 내용은 33 페이지의 “부록 C 메모리 카드에 있는 파일 구조”를 참조하십시오.
4. PC의 운영 체제에서 USB 연결을 해제하고 USB 케이블을 분리합니다. PC에서 카드 판독기를 사용한 경우는 microSD 메모리 카드를 제거하고 PCAN-MiniDisplay의 슬롯에 삽입합니다.

4.3 장치 전원 켜기

PCAN-MiniDisplay는 J2 커넥터에 7~30 V DC 범위의 전압을 입력하면 전원이 켜집니다.

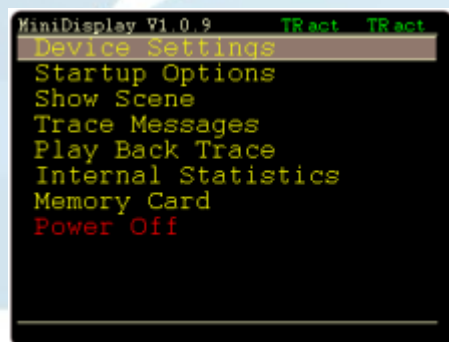
이 장치는 자동으로 시작됩니다.

납품된 후 변경되지 않았으면, PCAN-MiniDisplay는 이름이 Default.inp인 scene을 보여줍니다.

이전에 전압이 공급된 장치를 해당 메뉴 항목을 사용하여 꺼진 경우, 커넥터 J1 아래 쪽에 있는 down ↓ 버튼으로 다시 시작될 수 있습니다.

커넥터에 대한 자세한 내용은 7 페이지의 2장을 참조하십시오.

4.4 메뉴



PCAN-MiniDisplay는 기능의 설정을 변경하기 위해 메뉴에 대한 사용자 인터페이스를 제공합니다. 탐색은 예를 들면, 누름 버튼이 장착될 수 있는 커넥터 J1의 디지털 입력을 통해 이루어집니다. 연결에 대한 자세한 내용은 7 페이지의 2.1절을 참조하십시오.

팁: PCAN-MiniDisplay 케이스에 내장된 기본 버튼이 있습니다. 이 모델에 대한 자세한 내용은 52 페이지의 부록 E를 참조하십시오.

메뉴와 기능을 통한 검색은 위 ▲와 아래 ▼ 버튼으로 이동하고, 선택한 항목은 Enter ↵ 버튼으로 활성화 합니다.

Scene을 표시하고 Enter ↵ 버튼을 누르면 다시 메인 메뉴를 제공합니다.

팁: 메뉴의 사용은 scene의 순수 표시에 필요하지 않습니다. 왜냐하면 기본으로 제공되는 설정에서 자동으로 로드된 정의 파일 Default.inp에 기본 설정들이 정의될 수 있기 때문입니다. 따라서, 이 경우 버튼은 생략될 수 있습니다.

4.4.1 상태 표시

메뉴를 사용하는 동안, 화면 오른쪽 위의 아이콘은 CAN 채널 1 (고속 CAN) 및 2(Single-wire CAN)를 CAN 버스 통신의 상태를 나타냅니다.

아이콘	의미
	CAN traffic: T = Transmit, R = Receive 깜박임 : 발신 / 수신 CAN 메시지 Green: 일반 트래픽 Yellow, red: 결함 트래픽
	CAN 컨트롤러 상태(active , passive , bus off)에 대해 통지합니다. Bus-off 상태일 때, 높은 (전송) 오류율로 인해, 더 이상 메시지를 전송하거나 수신할 수 없습니다. 이 경우, 버스 문제(예를 들어, 잘못된 CAN 비트율)를 해결한 후에, 전원을 순환(off 후 on) 하여 CAN 컨트롤러를 다시 수행해야 합니다.
	CAN 채널은 수신 전용 모드에서 작동합니다. 이것은 장치 설정(26 페이지의 섹션 7.1.3 참조)을 사용할 수 있습니다.

4.5 Startup Options 설정

메인 메뉴 항목 **Startup Options** 에서 전원의 동작을 설정할 수 있는 가능성을 제공합니다.

⇒ startup할 때 **scene**을 표시하려면 다음과 같이 하십시오:

참고: 납품되었을 때, PCAN-MiniDisplay는 Default.inp 라는 scene을 시작하도록 구성되어 있습니다.

1. **Show Scene**을 클릭합니다.
2. 표시된 디렉토리 구조에서, scene(*.ins) 또는 프로젝트 (*.inp)를 선택하면 startup할 때 표시됩니다. 선택한 파일은 하단 상태 표시 줄에 표시됩니다.
3. 오른쪽에 있는 체크 상자를 선택하고 maker 를 설정합니다.
4. **OK**를 클릭합니다.

⇒ startup할 때 **trace**를 시작하려면 다음을 수행하십시오:

1. **Start Tracer**, 오른쪽에, 체크 상자를 선택하고 마커 를 설정합니다.
2. **Tracer timeout** 에서, CAN 메시지가 더 수신되지 않을 때 Tracer를 정지하는 기간을 선택합니다(1,3,5, 10, 30 sec, 1, 5 min). **never** 항목은 tracer 실행을 벗어나지 않습니다.
3. **OK**를 클릭합니다.

5 CAN Tracing

5.1 CAN 트래픽 기록

> Menu item **Trace Messages**

이 기능으로, RTR 프레임 및 에러 프레임을 포함한 입력되는 전체 CAN 트래픽은 PCAN-MiniDisplay의 메모리 카드의 trace 파일에 기록됩니다.
또한 타이밍도 관련됩니다.

참고: CAN tracing은 CAN 채널 1(High-speed CAN)에만 적용됩니다.

나중에 trace 파일(파일 이름: 연속 번호 trc00000.btr)은 CAN 버스에 기록된 CAN 메시지의 재생을 위해 사용될 수 있습니다(21 페이지의 섹션 5.2 를 참조).
대안으로서, 더 사용하고 평가를 위해 다른 형식으로 PC에서 기록을 변환할 수 있습니다(22 페이지의 섹션 5.3 참조).

참고: 필터는 특정 ID를 갖는 CAN 메시지를 통해 들어오는 CAN 트래픽에 적용될 수 있습니다. 필터링에 대한 자세한 내용은 24 페이지의 6장을 참조하십시오.

⇒ 기록하려면 다음과 같이 하십시오:

1. USB 연결이 PCAN-MiniDisplay와 PC 사이에 존재하는지 확인합니다.
2. **Start** 를 클릭합니다. 기록은 지정된 **File** 에 수행됩니다.
3. **Stop tracing** 으로 기록을 종료합니다.

표시	의미
File	현재 기록에 대한 trace 파일의 이름입니다. 파일 이름(trc0000.btr)이 자동으로 연속 번호와 함께 배치됩니다.
CAN queue level in %	수신 큐의 현재와 최대 적재 레벨(괄호 후자). 큐가 100%의 충전 레벨에 도달하면, 들어 오는 일부 CAN 메시지가 기록되지 않을 것입니다.
CAN messages total	이미 trace 파일에 기록된 CAN 메시지의 수.
File size	Trace 파일의 현재 크기 (kByte), 최대 가능한 파일 크기의 퍼센트에서 이미 사용된 저장 공간. 기록 중, trace 파일은 512 바이트 블록 단위, 각 25개 CAN 메시지 포함. 따라서, 1 Mbyte는 51200 CAN 메시지를 보유할 수 있습니다.

5.2 기록된 CAN 트래픽 재생(Playing Back)

> 메뉴 항목 **Play Back Trace**

PCAN-MiniDisplay는 연결된 CAN 버스로 binary trace 파일에서 다시 CAN 메시지를 재생할 수 있습니다. Trace 파일의 기록할 때 발생한 원래대로 CAN 메시지의 타이밍은 유지됩니다.

참고: CAN tracing은 CAN 채널 1(High-speed CAN)에만 적용할 수 있습니다.

⇒ trace 파일을 재생하려면 다음을 수행합니다:

1. USB 연결이 PCAN-MiniDisplay와 PC 사이에 존재하는지 확인하십시오.
2. 메인 메뉴에서, **Play Back Trace** 를 선택합니다.
3. **Select File** 을 사용하고 목록에서 재생(playback)할 trace 파일(*.btr)을 선택합니다.
4. trace가 반복적으로 재생하지 않으려면, 사이클을 원하는 번호(**1~9**)로 무한의 루프 모드를 수정합니다.
5. 여러 CAN 채널에서 CAN 메시지를 포함하는 trace 파일을 사용하는 경우 (예를 들어, 다른 장치에서 가져온 파일), trace에서 재생되는 **Channel** 번호를 선택합니다.
6. 재생을 시작하려면 **Play** 을 클릭합니다.
7. **Pause playback** 을 사용할 수 있습니다. 다음과 같은 옵션이 있습니다:

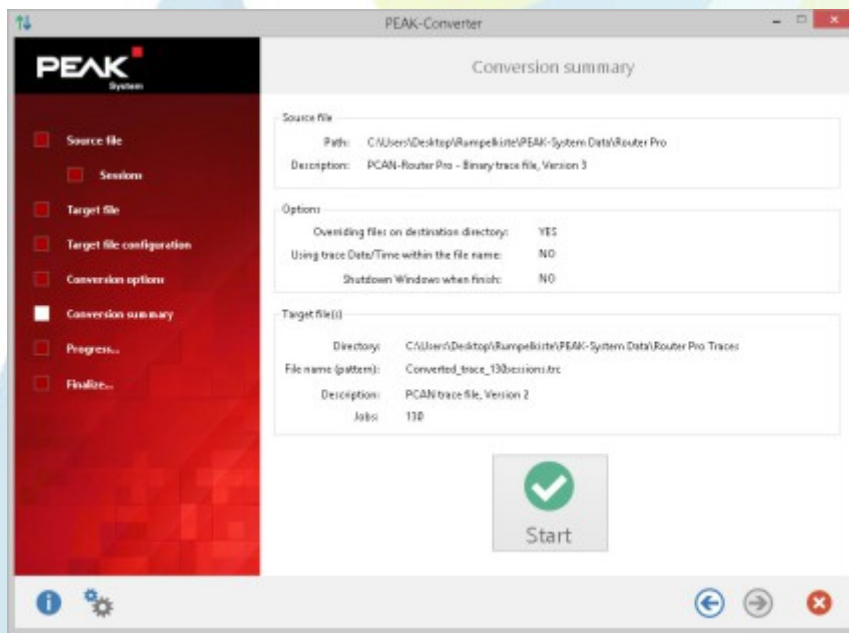
Function	Executed action
Exit	재생을 종료
Restart	trace 파일의 처음부터 재생을 다시 시작합니다.
Reset	CAN 컨트롤러를 재설정합니다. CAN 버스의 고장 보수 후에 유용합니다.
Continue	중단 전에 발생한 지점에서 재생을 계속합니다.

5.3 기록된 CAN 트래픽을 PC에서 사용하기

기록된 CAN 트래픽은 PCAN-MiniDisplay의 메모리 카드에서 USB 연결을 통해 PC에 의해 판독될 수 있습니다. 그것은 다음 디렉토리에 이진-코드 trace 파일의 trc00000.btr(연속 번호와 파일 이름)에 저장됩니다.

/MiniDisplay/Projects/Traces/

자세한 내용은 적절한 형식으로 데이터를 변환해야 사용할 수 있습니다. 이러한 목적으로 윈도우 프로그램 PEAK-Converter는 DVD와 PCAN-MiniDisplay의 메모리 카드에 공급됩니다.



PEAK-Converter의 사용자 인터페이스

가능한 변환 대상:

대상 형식	파일 확장자	설명/ 사용
PCAN-Trace	.trc	PEAK-System에 의해 텍스트 기반 trace 형식; PCANTrace 프로그램으로 CAN 메시지의 재생 또는 PCAN-Explorer에서 데이터 보기. 팁: PCAN-MiniDisplay의 trace 파일과 관련하여, 우리는 형식 버전 1.1을 사용하는 것이 좋습니다, PCAN-MiniDisplay의 기록은 하나의 채널이 있기 때문에 이 형식 버전은 PEAK-System의 모든 프로그램에서 사용할 수 있기 때문입니다.
Vector ASC Trace	.asc	일부 타사 프로그램에서 사용할 수 있는 벡터 회사에 의한 텍스트 기반 trace 형식입니다.
Character Separated Values(SCV)	.csv	일반적으로, 스프레드 시트로 가져 오기위한 텍스트 기반 형식(구분자로 세미콜론).

⇒ 다음과 같이 trace 데이터의 추가 사용을 위해 진행:

1. 제공된 USB 케이블로 PC에 PCAN-MiniDisplay를 연결합니다. PCAN-MiniDisplay는 전원을 켜 필요 없습니다.
2. 윈도우에서, /MiniDisplay/Tools 디렉토리에 있는 PCAN-MiniDisplay의 메모리 카드에서 PCAN-Converter.exe 프로그램을 시작합니다.
3. Source 에 trace 파일을 선택합니다(파일 이름: 연속 번호 trc00000.btr). Project 디렉토리에서 trace 파일을 찾을 수 있습니다: /MiniDisplay/Traces/
4. destination 파일을 지정하고 원하는 target format을 선택합니다(위 참조).

6 CAN 트래픽 필터링

일반적으로 들어오는 모든 CAN 메시지는 scene 또는 trace에 사용됩니다.
그러나, CAN 트래픽은 CAN ID의 목록에 의해 필터링 할 수 있습니다.
이 기능은 주로 tracing에 사용하기 위한 것입니다.

필터 목록은 .flt 확장자를 가진 텍스트 파일에 정의되어 있고, PCAN-MiniDisplay의 장치 설정에 로드 됩니다.
모든 언급한 ID들을 필터 통과, 다른 것은 하지 않습니다.
필터 파일은 메모리 카드에 저장하고 PCAN-MiniDisplay의 장치 설정에 로딩 됩니다.

⇒ CAN ID 필터를 적용하려면 다음을 수행하십시오:

1. 텍스트 편집기를 사용하여, 필터를 통과할 수 있는 ID를 나열하는 필터 파일 (*.flt)을 만들 수 있습니다. 파일 형식은 50 페이지 부록 D.3에 기재되어 있습니다.

팁: 배송될 때, PCAN-MiniDisplay는 Default.flt 라는 필터 파일을 사용하도록 설정되어 있습니다.

2. MiniDisplay 디렉토리에 PCAN-MiniDisplay의 메모리 카드에 필터 파일을 복사합니다. 당신의 PC의 USB에 장치를 연결하거나 당신의 PC에서 카드 리더를 사용합니다.
3. 필터 파일 Default.flt을 갖고 있는 경우 배송할 때 장치 설정을 사용하여 추가 단계가 필요하지 않습니다. 그 밖에, 다음 단계로 이동합니다.
4. PCAN-MiniDisplay에서, **Device Settings** 이동하고 **Ch.1 Filter** (High-speed CAN 채널 용 필터) 또는 **Ch.2 Filter** (Single-wire CAN 채널 용 필터) 중 1개를 선택합니다.
5. 이전에 생성된 필터 파일(*.flt)을 선택합니다.
6. 변경된 설정을 적용하려면 **OK** 또는 **Save&OK** 을 사용합니다.

Filter1.flt		
[GLOBAL]	[RANGE_11BIT]	[RANGE_29bit]
version=1	0x200-0x340	0x9000-0x10000
enable=1	4-13	500-550
	0x7f0-0x7fe	0x1F80000-
		0x1FA00000
[SINGLE_11BIT]	[SINGLE_29BIT]	
0x100	0x123	
0x250	0x11111	
1023	125000	

7 설정 및 유지 보수에 대한 메뉴 참조

7.1 장치 설정

여기서 장치의 사용을 위한 CAN 버스에 연결 및 설정을 지정.

이 설정을 변경한 경우, **Save&OK** 으로 영구적으로 저장할 수 있습니다. 일시적으로 변경된 설정을 사용하려면 **OK** 을 클릭합니다. off-on 주기 후 다시 초기 설정을 사용합니다.

7.1.1 Ch. 1/2 CAN 비트 레이트 검출

PCAN-MiniDisplay에 연결된 CAN 버스의 비트 레이트를 알 수 없는 경우, 장치는 자동으로 검출할 수 있습니다. 이것은 CAN 버스에 데이터 트래픽을 요구합니다.

다음 시리즈에서 비트 전송률(kbit/s)로 인식하고 있습니다:

1000; 800; 500; 250; 200; 125; 100; 95.2; 83.3; 50.0; 47.6; 33.3; 20.0; 10.0

7.1.2 Ch. 1/2 CAN 비트 레이트

CAN 비트 레이트의 시리즈에서 선택할 경우, 순서대로 연결된 CAN 버스의 하나에 대응합니다.

고정 비트 레이트 값 외에도, 목록에 사용자-정의 비트 레이트가 8개 포함되어 있습니다. 그것들은 **User CAN bitrates** 설정에서 관리됩니다.

Single-wire CAN 채널(**Ch.2**)을 사용하면, CAN 표준으로 제공하는 것보다 높은 비트율(최대 100kbit/s)을 선택될 수 있습니다. 그러나, 이것은 경고 및 비 작동 CAN 통신을 발생할 수 있습니다.

7.1.3 Ch. 1/2 Listen-only 모드

장치가 CAN 버스 트래픽에 영향을 주지 않으려면, 즉, 순수한 모니터링 도구로 사용하려면, Listen-only 모드로 활성화(**On**)해야 합니다. 장치가 CAN으로 들어오는 메시지에 응답하지 않도록(수신확인) 및 에러 프레임 전송하지 않습니다.

7.1.4 Ch. 1/2 필터

CAN ID 필터링은 메모리 카드에서 필터 파일(*.flt)을 선택하여 각 CAN 채널에 독립적으로 적용될 수 있습니다. 선택된 파일은 메뉴 항목 옆에 표시되고 대응 CAN 채널에 적용됩니다.

CAN ID 필터링을 사용하지 않으려면, 파일 선택 화면에서 **Exit** 로 이동합니다. **None** 은 다음 메뉴 항목 옆에 표시됩니다.

필터링에 대한 자세한 내용은 24 페이지의 6장을 참조하십시오.

7.1.5 Ch. 2 Transceiver 모드

CAN 채널 2의 Single-wire CAN Transceiver 는 3가지 모드 중 하나를 실행할 수 있습니다:

- **Normal:** 33.3 kbit/s까지, 파 형성.
- **Wake-up:** 일반 모드처럼, 하지만 증가된 신호 레벨로 전송, 슬립 모드에서, 버스 노드를 활성화. PCAN-MiniDisplay는 자체적으로 슬립 모드로 할 수 없습니다.
- **High-speed:** 83.3 kbit/s까지, 파 형성하지 않음, 특정의 경우에 사용합니다(소프트웨어 또는 진단 데이터의 전송 용). 일반 작업에 적합하지 않습니다.

7.1.6 User CAN 비트 레이트

특정 조건에 적응하기 위해, 전문가들은 직접 통합 CAN 컨트롤러의 버스 타이밍 레지스터(BTR)에 액세스할 수 있습니다. 레지스터 설정은 16Mhz의 클럭 주파수로 동작하는 CAN 컨트롤러 SJA1000의 아날로그입니다.

사용자 정의 항목 8개를 편집할 수 있습니다. 각 항목은 버스 타이밍 레지스터와 임의의 이름에 대한 2 바이트 값(16진수 4 자리)이 포함되어 있습니다. 사용자 정의 비트 레이트(bitrate)는 나중에 자신의 이름을 가진 고정 비트 레이트 값 아래의 **CAN bitrate** 목록에 나타납니다.

팁: 레지스터 값을 쉽게 결정하기 위해, 제공된 DVD에 Windows 프로그램(/Tools/BRCAN.exe)이 포함되어 있습니다.

각 항목에 대해 편집을 위해 테이블은 주어진 레지스터 값의 결과 매개 변수를 보여줍니다:

bitrate, sample point (SP), 그리고 synchronization jump width (SJW). 항목 재설정하려면 (Name = UserX, Bitrate = 미확정), BTR에 0000 값을 설정합니다.

7.1.7 신호음

PCAN-MiniDisplay는 여러 이벤트에 음향 피드백을 제공할 수 있습니다. 다른 것 중, CAN 버스 상태가 변화 되면 신호됩니다. **Off** 설정은 장치의 음향 신호 기능을 사용할 수 없습니다.

7.1.8 Show startscreen

PCAN-MiniDisplay 에 전원이 on된 후 시작 화면으로 몇 초 동안 비트 맵 파일을 보여줍니다. 이 화면을 전환할 수 있습니다.

자체적으로 사용되는 비트 맵 파일을 대체할 수 있습니다.

startscreen의 속성	
파일 이름	Intro.bmp
메모리 카드의 저장 경로	/MiniDisplay
형식	Windows bitmap
해상도	320 x 240 pixels
색상	24 bits

7.1.9 날짜와 시간

Set 으로 장치 날짜와 시간이 조정됩니다. 메모리 카드에 파일을 저장할 때 날짜와 시간이 사용됩니다.

7.1.10 파일 인덱스 초기화

트레이스 파일 이름은 카운터에서 오는 번호를 저장합니다. 현재 카운트가 괄호로 표시되어 **Reset** 을 클릭하여 0으로 설정될 수 있습니다.

7.2 Internal Statistics (내부 통계)

이 페이지는 장치의 내부에 대한 개요를 제공합니다. 사양은 일반적으로 지원 목적으로 사용됩니다.

또한, 하드웨어 기능은 장치의 유지 가능합니다. 그것들은 다음에 간략하게 설명되어 있습니다.

중요 참고: 이 기능을 잘못 사용하면 장치의 가용성이 발생할 수 있습니다. PEAK-System의 기술 지원에서 요청하는 경우에만 기능을 사용합니다.

Update Firmware

펌웨어 업데이트(*.bin)는 메모리 카드에 있는 /MiniDisplay/Firmware/ 디렉토리에 배치되어야 합니다. 업데이트 기능을 가진 파일이 선택됩니다. 업데이트 절차를 시작합니다.

Factory Defaults

모든 설정은 현재 펌웨어에 의해 정의된 기본 상태로 재설정됩니다.

Bootloader

CAN을 통해 펌웨어 업데이트에 대한 부트 로더를 시작합니다. 화면에는 PCAN-MiniDisplay의 일련 번호를 보여줍니다.

7.3 메모리 카드

PCAN-MiniDisplay 는 메모리 카드로부터 디렉토리 및 비트 맵을 표시하는 기능이 있습니다.

참고: PCAN-MiniDisplay는 USB 연결이 PC에 되어 있는 동안 메모리 카드를 액세스할 수 없습니다.

PCAN-MiniDisplay가 PC의 USB에 연결되어 있는 경우, Show Directory 기능을 실행하면 아래와 같이 표시되고 진행이 되지 않습니다. PC의 USB와 연결이 해제됩니다.

Please remove the USB cable

Press button to exit

PCAN-MiniDisplay가 PC의 USB에 연결되어 있는 경우, View Bitmap 기능을 실행하면 아래와 같이 표시되고 PC의 USB와 연결이 해제됩니다. View Bitmap 기능을 실행할 수 있습니다.

Please remove the USB cable

Press button to exit

Show Directory

파일이 존재하는지 참조하기 위해 메모리 카드의 디렉토리를 보여줍니다.

View Bitmap

비트 맵 파일(*bmp)만 디렉토리(예를 들어, instruments)에 표시됩니다.

비트 맵 파일을 보려면 enter ↵ 버튼을 누릅니다. 다시 누르면 벗어납니다.

8 기술 사양

Power supply	
Supply voltage	7 – 30 V DC
Current consumption	135 mA (at 12 V) typical

Display	
Type	a-Si TFT active matrix
Resolution	320 x 240 pixels
Active area	57.6 x 43.2 mm (W x L)

Memory card	
Type	microSD
Max. capacity	32 GByte
File system	FAT; FAT32 if capacity > 2 GByte
Name of the USB device	PCAN-MiniDisplay

CAN busses	
CAN 1	High-speed CAN (ISO 11898-2), not terminated
CAN 2	Single-wire CAN (SAE J2411)

Measures	
Size	69.5 x 49.5 mm (W x L) 32 페이지에 있는 '부록 B 도면 치수'를 참조하십시오.
Weight	50 g

Environment	
Operating temperature	-20 – +70 °C (-4 – +158 °F)
Temperature for storage and transport	-30 – +80 °C (-22 – +176 °F)
Relative humidity	15 – 90%, not condensing
EMC	EN 55024:2003-10 EN 55022:2007-04 EC directive 2001/108/EG

부록 A CE 인증

PCAN-MiniDisplay IPEH-002262 – EC Declaration of Conformity
PEAK-System Technik GmbH



Notes on the CE Symbol

The following applies to the "PCAN-MiniDisplay" product with the item number(s) IPEH-002262.

EC Directive This product fulfills the requirements of EU EMC Directive 2004/108/EC (Electromagnetic Compatibility) and is designed for the following fields of application as for the CE marking:

Electromagnetic Immunity/Emission
DIN EN 61326-1, publication date 2013-07
Electrical equipment for measurement, control and laboratory use – EMC requirements – Part 1: General requirements (IEC 61326-1:2012);
German version EN 61326-1:2013

Declarations of Conformity In accordance with the above mentioned EU directives, the EC declarations of conformity and the associated documentation are held at the disposal of the competent authorities at the address below:

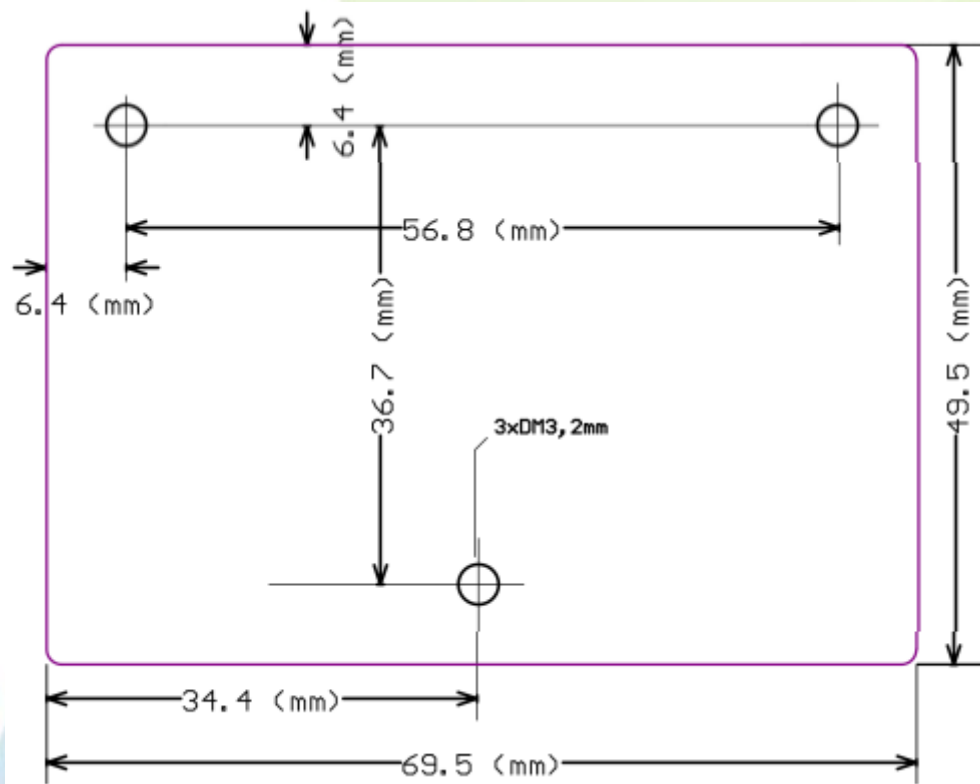
PEAK-System Technik GmbH
Mr. Wilhelm
Otto-Roehm-Strasse 69
64293 Darmstadt
Germany

Phone: +49 (0)6151 8173-20
Fax: +49 (0)6151 8173-29
E-mail: info@peak-system.com



Signed this 26th day of January 2015

부록 B 도면 치수



그림은 원래 크기를 표시하지 않습니다.

부록 C 메모리 카드에 있는 파일 구조

다음 표는 MiniDisplay 디렉토리 트리의 구조를 표시하고 기능을 설명합니다.

디렉토리-파일	기능
/MiniDisplay/	PCAN-MiniDisplay 또는 이 장치와 관련된 것에 의해 액세스되는 파일에 대한 고정 저장 지점.
Intro.bmp	장치가 시작될 때 화면 (320 x 240 픽셀).
Documentation/	PCAN-MiniDisplay에 대한 문서, 예를 들면 이 설명서.
Filters/	가능한 곳은 모든 필터 파일을 함께 유지합니다.
Firmware/*.bin	여기에 펌웨어를 업데이트 하는 파일을 놓습니다.
Help/*.dhp	장치 도움 파일.
Resources	자신의 scenes를 만들기 위한 기초로서 일부 비트 맵, 글꼴. 게다가, 당신은 Windows 용 PCAN-Explorer 을 소유하고 있다면, 당신은 PCAN-MiniDisplay scene 예에 대한 몇 가지 데이터를 생성하기 위해 제공되는 프로젝트 예제를 사용할 수 있습니다.
Scenes/<scene name>/	각 scene의 scene 이름과 하위 디렉토리.
*.ins	scene 정의 파일(12 페이지의 '3장 Scene 정의' 참조).
.inb	빠른 로딩을 위한 scene(.ins) 이진(binary) 대응(12 페이지 '3장 Scene 정의' 참조)
*.inp	디지털 입력으로 전환하는 여러 scene들의 사용을 위한 scene들의 목록.
*.fon	Scene에서 사용되는 Bitmap 폰트.
*.bmp	Instrument에서 사용되는 Bitmap.

디렉토리-파일	기능
Tools/	소프트웨어 툴은 PCAN-MiniDisplay에 사용됩니다(다음은 실행 파일을 나열합니다).
PEAK-Converter.exe	다른 형식으로 기록된 이진 trace 파일(*.btr)의 변환을 위한 Windows 프로그램 PEAK-Converter입니다.
Traces/	Trace를 위한 고정 하위 디렉토리는 PCAN-MiniDisplay에 의해 기록됩니다.
trc00000.btr	기록 기능으로 부터 이진 코드로 된(Binary-coded) trace 데이터, 재생하거나 PC에서 변환 후 사용 가능, 내부 카운터에 의해 연속 번호.

부록 D 정의 파일 참조

다음과 같은 정의 파일의 키워드는 이 부록에 설명되어 있습니다:

- D.1 Scene 정의 (*.ins), 아래에
- D.2 Scenes 프로젝트 파일 (*.inp), 49 페이지에
- D.3 CAN ID 필터 (*.flt), 50 페이지에

D.1 Scene 정의 (*.ins)

Scene 정의에 대해 사용 가능한 모든 키워드를 설명은 다음과 같습니다. 그들은 *.ins 파일에 기록되고 섹션에서 구성됩니다. Section 이름은 대괄호, 예를 들어 [global]로 표시됩니다. 항목은 각각 동일한 문자(공백없이) 및 할당된 값으로 키워드를 포함하는 라인에 배치됩니다.

참고: 텍스트 파일 (*.ins)는 그것의 끝에 빈 줄이 있어야 합니다.

D.1.1 [global] 섹션

format

이 때, 1 형식만 지원됩니다.

```
format=1
```

bitrate0 bitrate1

High-speed CAN 채널(bitrate0)과 Single-wire CAN 채널(bitrate 1)의 bitrate를 결정합니다. 이러한 키워드가 생략되면, **Device Settings** 메뉴에서 정의된 bitrate가 유효합니다(기본은 **BusOff**). 정의된 bitrate를 가진 scene이 닫혀 있는 경우, 그 bitrate는 활성 상태를 유지합니다.

참조: 장치의 시동 중, scene의 비트 전송률을 사용하기 전에 **Device Settings** 메뉴에 정의된 비트 레이트는 짧은 시간 동안 활성화됩니다.

가능한 bitrates (in kbit/s): 1000, 800, 500, 250, 200, 100, 95, 83, 50, 47, 33, 20, 10

```
bitrate0=250
bitrate1=20
```

instruments

이 scene에서 instrument들의 개수, 범위: 1... 32.

```
instruments=4
```

labels

사용된 label들의 개수, 범위: 1...64.

```
labels=5
```

variables

사용된 variable들의 개수, 범위: 1...96.

```
variables=7
```

plotter

사용된 plotter들의 개수, 범위: 1...4.

```
plotter=1
```

fonts

이 프로젝트에 대해 지정된 추가 글꼴들의 개수, 범위: 1...27.

그것은 사용 가능한 3개 내장 폰트입니다. 내장 폰트를 사용하려면, 폰트 이름 대신 font_idx를 사용합니다. 자세한 내용은 39 페이지에 있는 D.1.3 [fontX] 섹션을 참조하세요.

```
fonts=2
```

flashing_time

“flashing” 키워드로 instrument들을 위한 깜박임 주파수, 범위 50...2500, 기본값 500. 값은 “on-time”과 “off-time”에 대한 밀리 초 단위의 시간입니다.

```
flashing_time=250 // 250 ms on, 250 ms off, resulting in 2 Hz
```

compress

Scene를 로드하는 동안 compression(압축) 함수가 호출할지 여부를 결정합니다.

값	설명
0	텍스트 기반 구성 파일(*.ins)이 로드될 때 압축된 버전이 생성되지 않습니다.
1	텍스트 기반의 구성 파일이 로드되면, 바이너리 버전(*.inb)이 생성됩니다. 이 scene에 대한 inb 파일이 이미 존재하는 경우, 이 대신 텍스트 기반의 구성에 로드됩니다.

참고: scene의 정의가 변경되면, 로더는 새로운 바이너리 텍스트 파일에서 생성해야 볼 수 있도록 이전에

생성된 바이너리 파일(*.inb), PCAN-MiniDisplay의 메모리 카드에서 삭제해야 합니다.
당신이 바이너리 파일을 삭제하지 않는 경우, 정의 파일의 변경 사항이 적용되지 않습니다.

D.1.2 [instrumentX] 섹션

각 instrument는 [instrumentX] 섹션 구분 기호로 표시됩니다, X는 연속된 수이며, 1부터 시작.
Instrument의 수는 [global] 섹션에서 “instruments” 키워드로 정의됩니다.

[instrument1]

pictures

instrument 당 그림의 개수, 범위: 1...99.

pictures=4

name

Instrument이 이름입니다. 단지 문서화입니다. 문자열은 인용 부호로 작성해야 합니다.

name="turn_left"

var_name

Instrument를 제어하는 변수의 이름입니다. 문자열은 인용 부호로 작성해야 합니다.

var_name="speed"

initval

이 instrument에 처음 실제 CAN 데이터가 수신될 때까지 instrument의 초기 값을 결정합니다.

initval=267

no_restore

그리는 동안 배경의 처리를 결정합니다.

값	설명
0	Instrument 순서가 그려지기 전에, 배경이 저장됩니다. 새로운 instrument 순서가 그려지기 전에 그런 배경을 복원할 수 있습니다. 서열이 서로 다른 크기가 있는 경우에 유용할 수 있습니다.
1	그림 순서는 배경 저장없이 그려질 것입니다 - instrument의 모든 그림이 같은 크기로, 같은 위치에서 그린 경우만. CPU 부하의 최적화를 위해 사용됩니다.

no_restore=1

restore

위의 no_restore 가 아니라, 반전 논리와 동일합니다.

transparency

그리는 동안 투명 영역으로 간주해야 하는지 여부를 정의합니다.

값	설명
0	시퀀스는 배경과 관련되지 않게 그려집니다. 이것은 배경 순서의 일부인 경우에 사용됩니다.
1	*.ins 파일을 로드하는 동안, 모든 단일 사진이 배경으로 다시 계산됩니다. 따라서, instrument0은 키워드 “no_restore=1”만 항목으로 글로벌 배경 그림으로 정의해야 합니다. 재 계산에 사용할 두 가지 옵션이 있습니다: a) 사진은 “transparency” 컬러(RGB 값 255,0,255)를 갖습니다. b) 사진은 알파 채널을 갖습니다. 알파 채널은 별도의 계조(grayscale) 비트 맵에 의해 정의됩니다. 그레이 스케일 값은 전경 (foreground) 화상의 투명도를 정의합니다. RGB 값 255,255,255: 전경 사진은 100% 사용할 수 있고, 배경의 어떠한 부분도 볼 수 없습니다. RGB 값 127, 127, 127 : 전경 50%를 사용할 수 있고, 배경 50%를 사용할 수 있습니다. RGB 값 0,0,0: 전경 사진을 사용할 수 없고, 배경 화상 100%를 사용할 수 있습니다.

transparency=1

offset

본 instrument의 그림 목록의 모든 그림은 x 및 y 방향으로 이동합니다(offset).

첫 번째 매개변수: x 옵셋(범위: 0...319)

두 번째 매개변수: y 옵셋(범위 0...239)

offset=100,50

background

이 instrument에 대한 개별 배경 그림을 정의합니다.

값	설명
0	이 instrument에 대한 배경이 없습니다.
1	Instrument는 transparency=1 사용할 경우 개별 배경(background) 사진을 갖습니다. *.ins 파일을 로드하는 동안, 사진은 배경에서 계산됩니다. 기본적으로 (background=0) instrument0은 global 배경 사진입니다. 그럼에도 불구하고, 각각의 instrument에 대한 개별 배경 사진은 정의될 수 있습니다. 개별 배경 사진은 사진 목록의 마지막 사진입니다. 배경 사진은 알파 채널이 없는, 그래서 instrument0에서 global 배경 사진을 대체합니다. 일반적으로, 개별 배경 사진은 instrument의 단일 사진과 동일한 크기를 갖습니다.

background=1

update_rate

이 instrumnet의 업데이트 속도(밀리 초 단위). 정의되지 않은 경우, 50 ms의 기본 업데이트 속도가 사용됩니다. Instrument는 각각의 새로 들어오는 CAN 메시지와 함께 업데이트 될 수 있습니다. 그러나, 현재의 값으로 인해 너무 높은 업데이트 속도로 판독되지 않은 경우가 발생할 수 있습니다. 이 키워드는 instrument의 업데이트 신뢰성을 유지되도록 하여 업데이트 속도를 제한합니다.

```
update_rate=500 // Redraws the instrument
                // two times a second at most
```

Picture List

“var_name” 키워드는 CAN 신호를 정의합니다. 이 것이 수신되면, 이 값은 디코딩될 것입니다. 그림은 디코딩 된 CAN 신호에 할당되어 표시됩니다.

그림 목록 항목의 형식:

<index>=<pic-name>,<transparency-name>,<xpos>,<ypos>,<low-val>,<high-val>

picture list 항목의 부분	설명
<index>	그림의 수입입니다. {0 ~ pictures - 1} 사이에 있어야 합니다.
<pic-name>	경로없는 그림의 이름. 그림은 *.ins 파일이 저장되어 있는 같은 디렉토리에 저장해야 합니다.
<transparency-name>	Gray scale(알파 채널) 그림의 파일 이름. 이 그림이 없으면 빈 문자열 ""을 제공해야 합니다. 경로없이 파일 이름(<pic-name> 처럼).
<xpos>	x 위치 {0 ~319}. 매개 변수 x-Offset에 추가됩니다.
<ypos>	y 위치 {0~ 239}. 매개 변수 y-Offset에 추가됩니다.
<low-val>	데이터 범위의 가장 낮은 값.
<high-val>	데이터 범위의 가장 높은 값.

```
0="arrow_left_off.bmp", "",20,100,0,3
  // Show picture "arrow_left_off.bmp"
  // at position x=20, y=100
  // if the variable is in the range of 0 to 3
1="arrow_left_on.bmp", "",20,100,4,7
  // Show picture "arrow_left_on.bmp"
  // at position x=20, y=100
  // if the variable is in the range of 4 to 7
  // 변수가 범위 4~7에 있다면
```

D.1.3 [fontX] 섹션

각 사용된 글꼴은 [fontX] 섹션 구분 기호로 표시됩니다, 여기서 X는 1에서 시작하는 연속적인 번호입니다. 글꼴의 수는 [global] 섹션에서 “fonts” 키워드로 정의됩니다.

```
[font1]
```

font

비트 맵 폰트의 파일 이름. 문자열은 인용 부호로 작성해야 합니다. 글꼴 파일 이름의 확장자는 *.fon 입니다.

```
font="arial_20.fon"
```

현재 다음과 같은 글꼴은 PCAN-MiniDisplay에서 사용할 수 있습니다:

- lucida_13_18.fon
- century_13_16.fon
- century_16_18_numbers.fon
(only numbers from 0 to 9)
- Palatino_Linotype_12_14.fon
- Palatino_Linotype_16_20.fon
- Palatino_Linotype_24_28.fon
- Palatino_Linotype_24_32_numbers.fon
(only numbers from 0 to 9)
- Palatino_Linotype_24_32_numbers_bold.fon
(only numbers from 0 to 9)
- Courier_13_18_bold.fon
(bold characters)
- gear_40_49.fon
(numbers from 0 to 7 are displayed as N, 1, 2, 3, 4, 5, 6, R in a size of 40 x 49 pixels)

name

글꼴은 사용하는 요소에 대한 참조로 글꼴의 이름입니다.

```
name="arial_20"
```

type

글꼴의 종류. 현재 고정된 크기만한 색 글꼴을 사용할 수 있습니다(유형 0).

```
type=0
```

D.1.4 [variableX] 섹션

각 변수는 [variableX] 섹션 구분 기호로 표시됩니다, 여기서 X는 1에서 시작하는 연속적인 번호입니다. 변수들의 개수는 [global] 섹션의 “variables” 키워드에 의해 정의됩니다.

```
[variable1]
```

name

참고로 변수의 이름은 instrument 또는 label과 함께 사용할 수 있습니다.

```
name="Speed"
```

channel

채널을 사용합니다(카운트가 0에서 시작). 지원 채널의 수는 소스 유형(아래 “source” 키워드 참조)에 따라 달라집니다.

소스 유형	값	설명
CAN	0	High-speed CAN
	1	Single-wire CAN
AIN	0	Internal NTC
	1	External input
	2	External input
DIN	0	External input (e.g. switch)

```
channel=0 // Use first CAN channel ;CAN 채널 최초 사용
```

source

Source의 유형.

Available source types: CAN, AIN, DIN
 사용 가능한 source 유형: CAN, AIN, DIN

```
source=CAN
```


canid

11-비트 또는 29-비트 정수 값(10진수 또는 16진수).

```
canid=256
canid=0x100
```

position

첫 번째 매개 변수: 신호의 시작 비트 위치, 범위: 0~63

참고: 비트는 Intel과 Motorola 형식에서 다르게 계산됩니다.

두 번째 매개 변수: 신호의 비트 길이, 범위: 1~32

```
position=0,8 // CAN 데이터 필드의 첫 번째 바이트
position=8,16 // CAN 데이터 필드의 2와 3번째 바이트
```

frametype

Standard 프레임 또는 Extended 프레임과 같은 CAN 프레임 유형을 정의합니다.

값	설명
0	Standard Frame (11-bit CAN ID)
1	Extended Frame (29-bit CAN ID)

```
frametype=0 // 11비트 ID인 표준 프레임
```

byteorder

CAN 데이터의 바이트 순서를 정의합니다.

값	설명
0	Intel ("Little Endian")
1	Motorola ("Big Endian")

```
byteorder=0 // Intel format
```

muxtype

(아직 지원되지 않음)

muxval

(아직 지원되지 않음)

scale

입력되는 신호 값을 위한 Scale. 값은 float(32-비트)로 저장될 것입니다.

scale=2.5

offset

입력 신호 값을 위한 Offset. 값은 float(32 bit)으로 저장합니다.

offset=5.2

maxraw

내부 데이터 float 형 32 비트와 최대 원시(raw) 값(scale/offset 계산 전 CAN 버스로부터 값). 이보다 큰 값은 이 값에 한정됩니다. maxraw 또는 maxval 중 하나를 사용합니다.

maxval

내부 데이터 float 형 32 비트와 최대 physical 값(scale/offset 계산 후 CAN 버스로부터 값). 큰 값은 이 값에 한정됩니다. maxval 또는 중 maxraw 하나를 사용합니다.

Minraw

내부 데이터 float 형 32 비트와 최소 원시(raw) 값(scale/offset 계산 전 CAN 버스로부터 값). 이보다 작은 값은 이 값에 상승됩니다. minraw 또는 minval 중 하나를 사용합니다.

minval

내부 데이터 float 형 32 비트와 최소 physical 값(scale/offset 계산 전 CAN 버스로부터 값). 작은 값은 이 값에 상승됩니다. minval 또는 minraw 중 하나를 사용합니다.

vartype

CAN 원시 데이터로부터 추출한 변수의 데이터 유형.

값	설명
0	unsigned, integer
1	signed, integer
2	float (32 bits)

vartype=1

8비트 길이 부호있는 정수 변수는 -128에서 127까지의 값의 범위를 갖습니다.

datatype

CAN 원시 데이터로 부터 값을 읽은 후 scale/offset 계산, 결과 값은 이 키워드에 따라 저장됩니다.

값	설명
2	unsigned, integer
1	signed, integer
2	float (32 bits)

vartype=2

값은 내부적으로 float 값으로 저장됩니다. 값이 분수로 표시할 경우에 유용합니다(labels 참조).

variable의 예

```
[variable1]
name="odometer"
canid=0x250           // CAN ID
position=8,16        // Start bit and bit length
frametype=0          // 0: 11-bit, 1: 29-bit
byteorder=1          // 0: Intel, 1: Motorola
scale=1.5             // Value=raw * scale + offset
offset=0              // Always float type
vartype=0             // Representation of CAN data:
                       // 0: unsigned, 1: signed, 2: float
minval=3              // Lower limit is 3
maxval=99999          // Any value > 99999 is set to 99999
datatype=1           // Representation of result:
                       // 0: unsigned, 1: signed, 2: float
```

D.1.5 [labelX] 섹션

각각의 텍스트 레이블은 [labelX] 섹션 구분 기호로 표시됩니다, 여기서 X는 1에서 시작하는 연속적인 번호입니다. 라벨의 번호는 [global] 섹션에서 “labels” 키워드에 의해 정의됩니다.

[label1]

name

문서에 대한, label의 이름입니다.

name="odometer"

position

Label의 x와 y의 위치입니다.

첫 번째 매개변수: X 위치(수평), 범위: 0 ~ 31
두 번째 매개변수: Y 위치(수직), 범위: 0 ~ 239

position=10,30

font_name

Font의 이름, 섹션 fonts에서 font 참조.

font_name="Palatino_Linotype_12_14.fon"

이것은 14 픽셀의 높이와 12 픽셀의 폭으로 된 font를 선택합니다.
사용 가능한 font의 목록을 참조하십시오.

font_idx

포함된 font의 인덱스입니다.

값	설명
0	“Courier_8_12”, width=8, height=12
1	“Courier_10_14”, width=10, height=14
2	“Courier_12_18”, width=12, height=18

font_idx=2

length

출력 문자의 길이. 문자의 수.

length=5

fmt_string

Format string은 변수의 값을 출력 문자열로 변환합니다. printf 프로그래밍 기능과 같은 형식으로 사용.

```
fmt_string="%02d km"
```

var_name

섹션 variables에서 변수 참조.

```
var_name="speed"
```

initval

중요한 변수가 수신되기 전에 시작 후 출력을 초기 값이다.??

```
initval="-----"
```

fontcolor

빨강, 녹색, 파랑의 폰트의 색상을 정의합니다.

```
fontcolor=255,255,255 // white
```

bgcolor

빨강, 녹색, 파랑의 배경의 색상을 정의합니다.

```
bgcolor=255,0,0 // red
```

update_rate

밀리 초에서 label의 업데이트 속도를 정의합니다.

Label은 관련 변수의 수신과 상관없이 이보다 빨리 다시 그려지지 않습니다.

```
update_rate=100 // Redraw label not faster than every 100 ms
update_rate=0 // Updates label as fast as possible
// (each time the variable is received)
```

sector

변수의 수치 범위는 sector 5까지 나눌 수 있습니다. 각 부문에 대한 다른 글꼴 색상과 배경 색상을 정의할 수 있습니다. 각 sector는 1 또는 0으로 설정 "flashing" 키워드를 가질 수 있습니다.

sector는 enabled(1) 또는 disable(0) 으로 할 수 있습니다.

```
sector=1 // 섹션 활성화
range=0,10
fontcolor=255,0,0 // bright red
offcolor=127,0,0 // light red
bgcolor=0,0,0 // black
flashing=1
```

```

sector=1          // sector enabled
range=1,244
fontcolor=0,255,0 // bright green
bgcolor=0,0,0    // black
flashing=0       // no flashing

sector=1          // sector enabled
fontcolor=255,0,0 // bright red
offcolor=127,0,0 // light red
bgcolor=0,0,0    // black
flashing=1

```

8-비트 unsigned 변수는 0~255의 값의 범위를 갖습니다.
 값이 0~10 이면, 표시된 값은 빨간색으로 flashing 됩니다.
 11~245의 범위에서는 녹색으로 표시됩니다.
 245~255의 범위는 빨간색으로 flashing 됩니다.

range

Sector는 소정의 범위에 대해 정의된다.

```
range=20,50
```

변수가 20~50 범위(20과 50 포함)에 있는 경우, 이 sector는 유효합니다.

offcolor

각 sector는 “flashing” 키워드를 가질 수 있습니다.

Sector는 활성화되고 label은 flashing됩니다, 색상은 “fontcolor”와 “offcolor” 사이를 전환합니다.

```
offcolor=127,0,0 // middle red
```

Label의 예

```

[label1]
name="odometer"          // 참조를 위한 label의 이름
position=10,30           // x-pos,y-pos in pixels
                        // 0, 0은 위/왼쪽 코너입니다.
//font_name=""          // font_name 또는 font_idx를 사용
font_idx=2               // 통합 글꼴 2를 사용
length=8                 // label의 문자 수
fmt_string="%05d km"    // printf()에 대한 형식 string은
                        // 표시 값으로 변수를 변환
var_name="odomtr"       // 변수 참조
startval="----- km"  // 표시되는 초기 값
fontcolor=0,255,0       // RGB 값 (밝은 녹색)
bgcolor=127,0,0         // RGB 값 (중간 빨간색)

```

```

sector=1           // 새로운 섹터, 1: 활성화
range=0,10        // 이 범위 내에서 값을 적용

fontcolor=0,255,0 // 이 범위의 글꼴 색상의 값
bgcolor=0,0,0     // 이 범위의 배경 색상의 값

flashing=0

sector=1
range=1,20
fontcolor=255,0,0
offcolor=127,0,0
bgcolor=0,0,0
flashing=1

```

키워드 “sector”가 사용되면, “fontcolor”와 “bgcolor”에 대한 매개 변수는 sector에 할당됩니다.

D.1.6 [plotterX] 섹션

각 플로터는 [plotterX] 섹션 구분 기호로 표시됩니다, 여기서 X는 1에서 시작하는 연속적인 번호입니다. 플로터의 수는 [global] 섹션에서 “plotter” 키워드로 정의됩니다.

name

플로터의 이름, 참조 용으로.

```
name="Vehicle Speed"
```

position

플로터의 X와 Y 위치.

```
position=100,50
```

size

플로터의 치수, 픽셀.

첫 번째 매개변수: 폭
두 번째 매개변수: 높이

```
size=100,50
```

plotcolor

RGB 에서 플롯 라인의 색상.

```
plotcolor=255,0,0 // Color is 100 % red  
plotcolor=127,127,127 // Color is mid gray
```

bgcolor

RGB에서 플로터의 영역에 대한 배경 색상.

```
bgcolor=255,255,255 // white background
```

enable

플로터를 활성화 하거나 비활성화.

```
enable=1
```

channels

이 플로터 도표를 위한 채널 수. 최대 4 채널을 플로터에 표시할 수 있습니다. 채널 인덱스는 이름, 변수 등을 지정하는데 사용됩니다.

```
channels=2
```

timespan

x 축이 나타내는 것은 밀리 초(milliseconds)입니다.

```
timespan=5000 // 5 s to draw the complete diagram
```

var_nameX

데이터 소스의 이름입니다. X는 플로터 다이어그램에서 최대 4개의 변수에 대해, 1, 2, 3, 또는 4 일 수 있습니다. “channels” 키워드를 참조하세요.

```
var_name1=speed  
var_name2=acceleration
```

yminX

수직 축의 최소 값. 이보다 작은 값이 최소로 제한됩니다.

```
ymin1=10
```

ymaxX

수직 축의 최대 값.
이보다 큰 값이 최대 값으로 제한됩니다.

```
Ymax2=250
```


D.2 Scenes 프로젝트 파일(*.inp)

D.2.1 [global] 섹션

version

버전의 형식입니다. 현재는 1입니다.

```
version=1
```

D.2.2 [scenes] 섹션

scenes

사용된 scenes의 개수입니다.

```
scenes=4
```

startscene

이 scene는 디폴트로서 scene 시작 후 사용될 것입니다, 기본값: 1.

```
startscene=2
```

List of Scenes

scenes 목록 entry의 형식:

```
<index>=<filename>
```

scenes는 연속적으로 번호가 매겨집니다, 1부터 시작. 매개변수로서, scene의 파일명은 표시되어야 합니다. Scene 이 키워드 매개변수 "compress=1"로 정의된 경우 이미 로드 되기 전에 압축된 버전이 사용됩니다. 이것은 로딩 시간을 감소시키는데 도움이 됩니다.

```
[scenes]
scenes=4
startscene=2
1="speedo0.ins"
2="speedo1.ins"
3="picture1.ins"
4="picture2.ins"
```

D.3 CAN ID 필터(*.flt)

CAN ID 필터 (*.flt 파일)의 사용은 PCAN-MiniDisplay의 **Device Settings** 메뉴에서 결정됩니다. 필터를 통과할 수 있는 필터 파일에 나열된 CAN ID들과 ID 범위, 다른 것은 하지 않습니다.

D.3.1 필터 파일 형식 설명

- .flt 텍스트 파일은 각 대괄호 섹션 이름에 의해 도입되는 여러 섹션이 있습니다.
- 섹션에 따라서, 그것은 CAN ID들, 또는 CAN ID 범위 키워드를 포함할 수 있습니다.
- CAN ID는 10진수 또는 16진수 값(점두사 0x 문자)으로 표시됩니다.
- 성능상의 이유로, 이를 위해 오름 순의 CAN ID를 나열하는 것이 좋습니다.
- 주석은 이중 슬래시로 시작으로 삼입할 수 있습니다.

섹션/키워드	설명	항목 예
[global]	이 섹션은 필수이며, 다음과 같은 두 가지 키워드를 포함합니다.	
version	버전 형식. 현재는 1만 유효.	version=1
enable	필터 활성화. 현재는 1만 유효.	enable=1
[single_11bit]	필터를 통과할 수 있는 11비트 CAN ID를 정의합니다.	0x100 1023
[range_11bit]	필터를 통과할 수 있는 11비트 CAN ID의 범위를 정의합니다.	4-13 0x200-0x340 // Full 11-bit range: 0x000-0x7ff
[single_29bit]	필터를 통과할 수 있는 29비트 CAN ID를 정의합니다.	0x123 0x11111 125000
[range_29bit]	필터를 통과할 수 있는 29비트 CAN ID의 범위를 정의합니다.	500-550 0x9000-0x10000 0x1F80000-0x1FA0000 // Full 29-bit range: 0x0000000-0x1FFFFFFF

D.3.2 필터 파일 예

```
[global]
version=1
enable=1

// This is a comment
[single_11bit]
0x100
1023 // = 0x3FF - Another comment
0x5AB

[range_11bit]
4-13
0x200-0x340
0x7f0-0x7fe

[single_29bit]
0x123
0x11111
125000 // = 0x1E848

[range_29bit]
500-550
0x9000-0x10000
0x1F8000-0x1FA0000
```

부록 E 케이스로 된 모델

이 부분은 케이스와 푸시 버튼 모델의 특수성을 설명합니다(IPEM-002262-KSM1).

이 제품 모델은 PCAN-MiniDisplay를 위한 구성의 개발을 촉진할 수 있는 환경을 제공합니다.



케이스와 푸시 버튼 모델

E.1 공급 범위

- 3개의 푸시 버튼이 있는 플라스틱 케이스로 된 PCAN-MiniDisplay
- 2 spring terminal blocks (10-pole, 4-pole)
- microSD memory card (512 MByte)
- PDF 형식의 사용 설명서 (DVD)

E.2 푸시 버튼

3개의 푸시 버튼 위 “↑”, 아래 “↓”, 그리고 엔터 “↵”는 장치 내부적으로 PCAN-MiniDisplay의 J1에 연결되어 있습니다 (자세한 내용은 7 페이지 2.1 참조). 사용은 장치의 current mode에 의존합니다.

Current Mode	Up ↑	Down ↓	Enter ↵
Scene display	여러 scene이 정의 된 경우 다음 scene으로 전환합니다.		Scene 표시를 벗어나 메인 메뉴를 표시합니다.
Menus	메뉴 또는 목록의 이전 또는 다음 항목으로 건너 뛴니다.		선택한 항목을 활성화합니다.
Off (supply voltage applied 공급 전압 인가)	No function	Start device	No function

J1의 디지털 입력은 4극 콘센트에도 사용 가능합니다.

E.3 커넥터

케이스로 된 모델은 뒷면에 3개의 커넥터를 제공합니다:

- Mini USB socket
- 스프링 터미널 블록 4극 커넥터(J1 내부)

핀	기능
1	Din1: Down
2	Din2: Up
3	Din3: Enter
4	GND



- 스프링 터미널 블록 10극 커넥터(J2 내부)

핀	기능
1	Vb (7 – 30 V DC)
2	GND
3	DDout1
4	Din4
5	Ain1
6	Ain2
7	Not connected or CAN-RxD ²
8	CAN2_SW or CAN2-TxD ²
9	CAN1_L
10	CAN1_H



² 옵션인 외부 CAN 트랜시버를 위한 TTL 신호

E.4 microSD 카드 슬롯

MiniSD 카드 슬롯은 케이스를 개방함으로써 접근할 수 있습니다.

⇒ miniSD 카드 슬롯에 액세스하려면 다음을 수행하십시오:

1. 케이스의 바닥에서 조심스럽게 4개의 플라스틱 캡을 제거합니다.
2. 캡에 포함된 홀에서 4개의 나사를 제거합니다.
3. 플라스틱 케이스의 상단(창 포함)을 제거합니다.

MiniSD 카드 슬롯은 하나의 버튼 근처 display 기판 아래에 있습니다.



microSD 카드 슬롯의 위치

4. microSD 카드를 제거하려면, 스프링 기구가 해제될 때까지 밀어 넣습니다(클릭 소리가 남).
5. microSD 카드를 삽입할 때, 스프링 기구가 잠길 때까지 밀어 넣습니다(클릭 소리가 남).
6. 케이스를 조립하려면 처음 세 단계를 반대 순서로 수행하십시오.