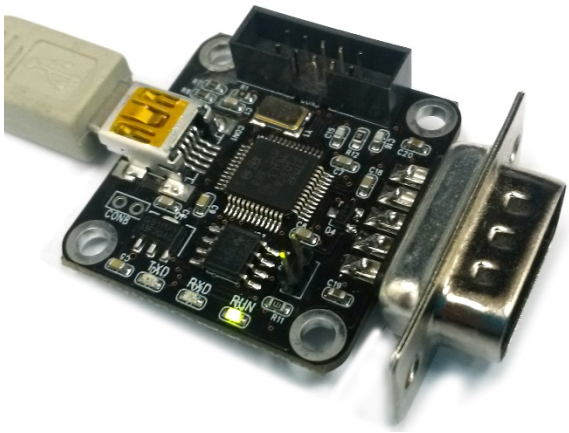
USB2CAN

USB2CAN-FIFO

USB2CAN-VCP



User's Manual

©Copyright NTREXLAB



사용자 매뉴얼에 포함된 정보는 정확하고 신뢰성이 있는 내용입니다. 그러나 출판 당시 발견되지 않은 오류가 있을 수 있으니 사용자는 자신의 제품 검증을 수행하시기 바라며, 전적으로 사용자 매뉴얼에 포함된 정보에 의존하지 마시기 바랍니다.

1 제품 소개 및 구성 1

1.1 USB2CAN 종류 **1**

1.1.1 USB2CAN-FIFO 2

1.1.2 USB2CAN-VCP 2

1.1 사용 예시 **2**

1.2 제품 구성 **3**

1.3 하드웨어 **3**

1.3.1 하드웨어 사양 3

1.3.2 도면 및 치수 4

1.3.3 설치 및 사용 4

1.4 소프트웨어 **4**

1.4.1 제공되는 소프트웨어 4

1.4.2 공장 출하시 기본 설정 값 4

1.5 커넥터 **5**

1.5.1 USB 연결 5

1.5.2 CAN 연결 5

1.1 연결 구성 **6**

1.6 120Ω 점퍼 **7**

1.7 LED 상태 표시 **7**

2 포트 설정과 연결 9

2.1 CAN 포트 **9**

2.1.1 CAN 속도(bitrate) 9

2.1.2 CAN 필터 10

2.2 가상 시리얼 포트 (USB2CAN-VCP 만 해당) **11**

2.2.1 USB2CAN UI 유틸리티를 사용한 연결 11

2.2.2 Hyperterminal을 사용한 연결 12

2.3 USB FIFO 포트 (USB2CAN-FIFO 만 해당) **14**

3 통신 프로토콜 (USB2CAN-VCP 만 해당) 15

3.1 텍스트 패킷 15
3.1.1 텍스트 명령 15
3.1.2 CAN 메시지 17

3.2 바이너리 패킷 17
3.2.1 명령어 패킷 18
3.2.2 메시지 패킷 19

4 USB2CAN UI 유틸리티 20

4.1 다운로드 및 실행 20

4.2 메인 화면 21

4.3 UI 설정 (USB2CAN-FIFO 만 해당) 21

4.4 연결 22

4.5 CAN 설정 22

4.6 송수신 메시지 표시 23

4.7 메시지 전송 24

5 장치 드라이버 25

5.1 장치 드라이버 다운로드 25

5.2 USB2CAN-FIFO 드라이버 설치 26

5.3 USB2CAN-VCP 드라이버 설치 27

1 제품 소개 및 구성

CAN bus는 메시지 기반 프로토콜로 차량용 어플리케이션으로 개발되었지만, 현재는 산업 자동화, 의료 장비 등에서 다양하게 사용되고 있습니다. 하지만 PC에서는 CAN bus에 연결하는 인터페이스를 직접적으로 제공하지 않기 때문에 CAN 통신을 하는 장치들을 PC에 연결하여 사용하려면 고가의 CAN 인터페이스 카드를 PCI 슬롯에 장착하여야 합니다.

USB2CAN 컨버터는 CAN bus에서 전송되는 메시지를 호스트 PC의 USB 포트로 읽고 쓰기 위한 CAN to USB 프로토콜 변환기입니다. 사용자는 CAN에 대한 기본 지식을 가지고 있다면 쉽게 응용할 수 있습니다.

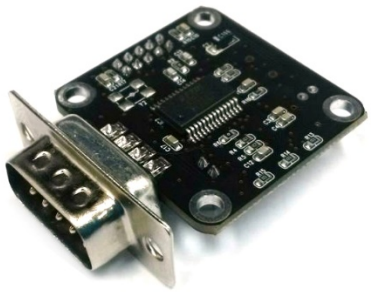


그림 1-1 USB2CAN 컨버터

USB2CAN 컨버터의 양단에는 USB 포트와 CAN 포트가 있습니다. USB2CAN 컨버터는 USB 전원을 사용하기 때문에 따로 전원을 공급할 필요가 없습니다.

*주) 호스트 PC는 USB2CAN 컨버터와 USB로 연결되어 CAN 메시지를 주고받거나 설정을 위해 사용됩니다.

1.1 USB2CAN 종류

USB2CAN은 사용한 FTDI 칩셋에 따라 **USB2CAN-FIFO**와 **USB2CAN-VCP** 제품이 있습니다. 사용자는 개발환경에 맞는 제품을 선택하시기 바랍니다:

- USB2CAN-FIFO – 전용 API를 사용하여 CAN 메시지 송수신
- USB2CAN-VCP – 가상 시리얼 포트(VCP; Virtual COM Port)를 통한 CAN 메시지 송수신

USB2CAN-FIFO와 USB2CAN-VCP 제품의 외형상 차이는 없습니다. 만일 구매한 제품의 종류가 확인되지 않는다면, USB2CAN 컨버터 뒷면의 FTDI 칩셋의 시리얼 넘버를 확인하기 바랍니다.



- FT245RL - USB2CAN-FIFO
- FT232RL - USB2CAN-VCP

1.1.1 USB2CAN-FIFO

USB2CAN-FIFO는 FT245 칩을 사용하여 USB와 MCU간 병렬로 데이터를 주고받는 형태로 데이터 전송률은 최대 1M byte/second가 됩니다.

FT245 칩을 사용한 제품의 UI(User Interface)와 DLL은 FTDI chip 사의 D2XX driver와 D2XX API를 사용하여 개발되었습니다.

1.1.2 USB2CAN-VCP

USB2CAN-VCP는 FT232 칩을 사용하여 USB와 MCU간 비동기 시리얼 데이터를 주고받는 형태로 데이터 전송률은 최대 300K byte/second가 됩니다.

FT232 칩을 사용한 제품의 UI와 DLL은 FTDI 드라이버에서 가상시리얼포트(VCP; Virtual COM Port)를 제공하기 때문에 일반적인 시리얼 통신 API를 사용하여 개발되었습니다.

1.1 사용 예시

CAN 통신 장치를 USB2CAN 컨버터로 호스트 PC와 통신하도록 하는 가장 간단한 구성은 다음 그림 1-2와 같습니다. CAN 포트를 제공하지 않는 PC에서 CAN 통신 장치를 연결할 때, 이러한 연결 구성이 가능합니다.

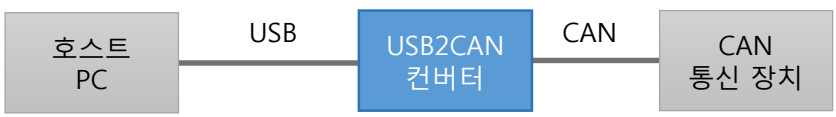


그림 1-2 간단한 연결 구성

다음 그림 1-3과 같이 호스트 PC는 하나의 USB2CAN 컨버터를 사용하여 여러 CAN 통신 장치를

연결할 수도 있습니다. 이는 CAN 프로토콜이 멀티 마스터 통신을 함으로 통신버스를 공유하고 있는 여러 CAN 장치들은 모두가 마스터 역할을 하여 언제든지 버스를 사용할 수 있게 되기 때문입니다. 여기서 USB2CAN 컨버터는 필터의 설정으로 특정 CAN 메시지만 수신하도록 제한할 수 있습니다.

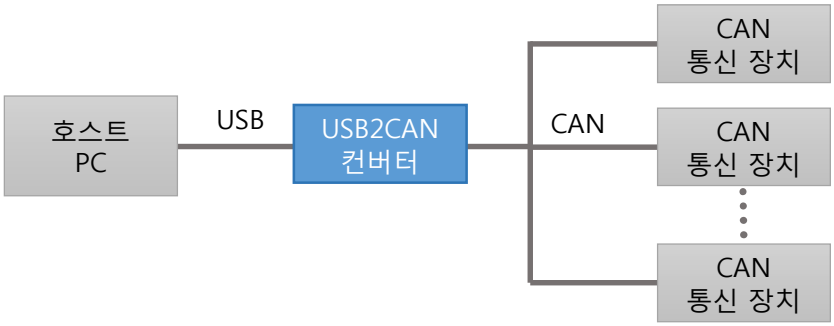
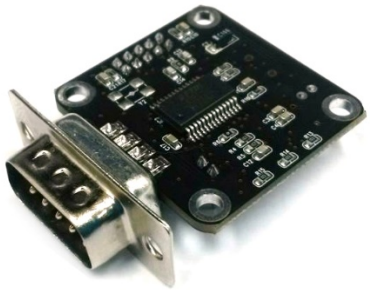


그림 1-3 여러 CAN 장치들의 멀티 연결 구성

1.2 제품 구성

- USB2CAN-FIFO or USB2CAN-VCP 1개
- Mini USB 케이블 1개





1.3 하드웨어

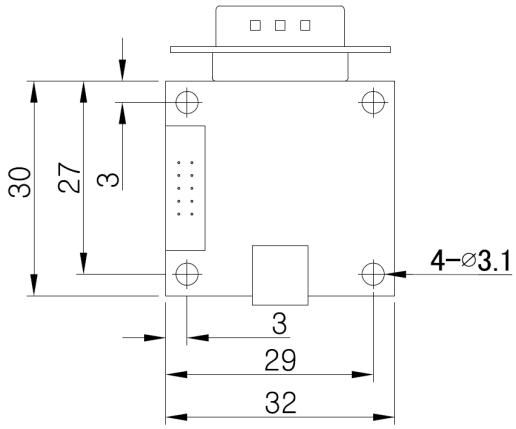
USB2CAN은 CAN bus를 통해 전송되는 CAN 메시지를 USB 프로토콜로 변환합니다. USB 연결에는 FTDI 사의 FT232 혹은 FT245 칩이 사용됩니다.

1.3.1 하드웨어 사양

- 입력 전압 : DC5V (USB 포트를 통해 공급됨)
- 소비 전류 : 80~100mA
- 동작 온도 : -20 ~ 65°C

- 크기 : 30x30 mm
- 무게 : 20g

1.3.2 도면 및 치수



1.3.3 설치 및 사용

※주의※ USBCAN은 케이스가 없는 벌크 타입의 제품입니다. 따라서 전자파 또는 자기장이 발생하거나 폭발성, 인화성, 부식성 물질 또는 물이 닿는 장소에서는 사용하지 마십시오.

1.4 소프트웨어

1.4.1 제공되는 소프트웨어

- USB2CAN UI 유틸리티 제공
- USB2CAN-FIFO, USB2CAN-VCP 장치 드라이버 제공
- USB2CAN-FIFO, USB2CAN-VCP를 엑세스하기 위한 API 라이브러리 제공

1.4.2 공장 출하시 기본 설정 값

- Bitrate : 1Mbps
- Filter identification : 0
- Filter mask : 0

1.5 커넥터

1.5.1 USB 연결

PC와 USB 연결에는 Mini USB B type의 커넥터를 사용합니다. 그리고 USB 연결을 통해서 USB2CAN 컨버터의 전원을 공급받습니다. 그렇기 때문에 따로 전원을 연결할 필요가 없습니다.



그림 1-4 USB 케이블과 연결

1.5.2 CAN 연결

CAN 연결에는 가장 일반적으로 사용하는 D-Sub 9pin male type 커넥터를 사용합니다. D-Sub 커넥터의 핀 위치를 확인하신 후 연결하기 바랍니다.

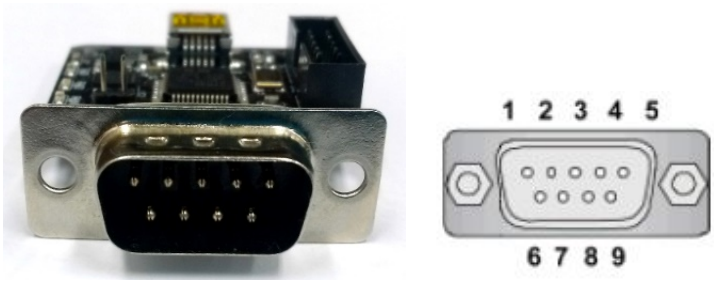


그림 1-5 D-Sub 9pin male 커넥터와 핀 위치

Pin	Function
1	-
2	CAN Low (CAN_L)
3	Ground (GND)
4	-
5	-
6	-
7	CAN High (CAN_H)
8	-
9	-

1.1 연결 구성

CAN 통신은 RS485와 유사하게 twist pair 2선을 사용하여 전기적 differential 통신을 하기때문에 전기적인 노이즈에 매우 강합니다. 그리고 비교적 저렴하게 네트워크를 구축할 수 습니다.

다음 그림 1-6은 USB2CAN 컨버터와 CAN 장치들이 CAN 버스에 연결되는 구성을 보여줍니다.

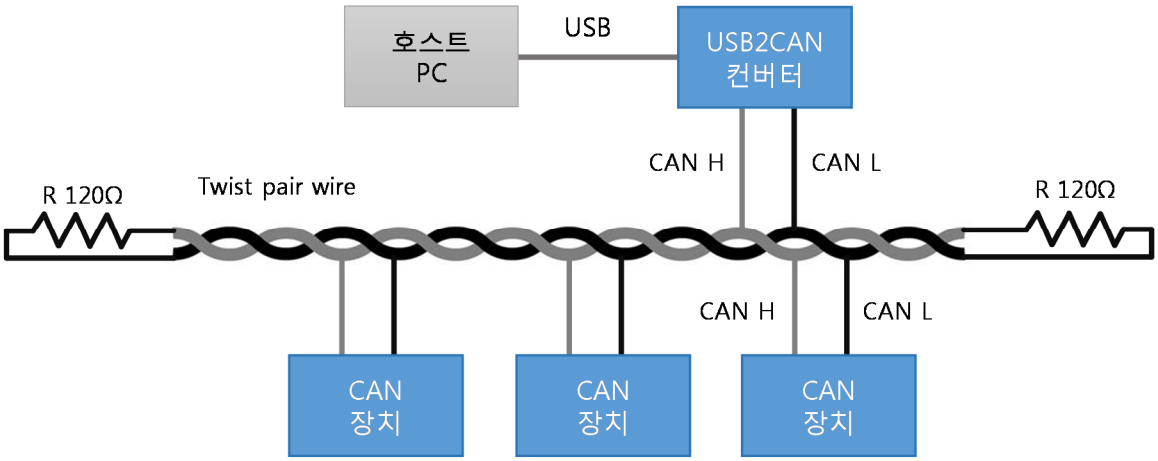


그림 1-6 CAN 버스의 연결 구성

USB2CAN 컨버터와 CAN 장치들의 CAN_H와 CAN_L를 CAN 버스 라인에 연결합니다. 상기 그림에는 표시되어있지 않지만, CAN 커넥터에는 GND의 연결도 필요합니다. 하지만 모터 제어기와 같이 다량의 전기적 노이즈를 방출하는 경우는 GND를 CAN 버스에 연결하지 않는 것이 좋습니다.

CAN 버스 신호선(CAN_H과 CAN_L)의 양 끝은 종단해주어야 합니다. 이것은 CAN 버스에서 통신이 양방향으로 이루어져야 하기 때문입니다. 통신 선로에서 종단 저항은 선로의 임피던스와 일치해야 합니다. ISO 11898는 임피던스가 120Ω이 될 것을 권장합니다. 만약 통신 선로에 여러 장치가 연결되는 경우 선로의 양 끝에 있는 장치에만 종단저항을 연결해야 합니다.

일반적으로 CAN 통신 속도와 통신 선로의 길이는 반비례하며, 통신 속도가 높으면 통신 선로의 길이는 짧아지고 반대로 통신 속도가 낮으면 먼 거리까지 통신할 수 있게 됩니다. ISO11898에서 1Mbps 속도에서 통신 거리를 40m정도로 규정하고 있습니다.

통신 속도	선로 길이
1Mbps	40m
500Kbps	100m
250Kbps	200m
125Kbps	500m

1.6 120Ω 점퍼

CON4 점퍼를 ON/OFF 함에 따라 CAN 버스 종단의 120Ω 저항을 연결하거나 끊습니다. CAN 버스의 종단에 120Ω 저항은 CAN H와 CAN L 사이의 전압을 2V로 유지해 줌으로써 버스 라인 종단에서 데이터가 중첩될 가능성이 있는 메아리 현상을 제거해 줍니다.



그림 1-7 USB2CAN 컨버터의 120Ω 점퍼

※ CAN 통신이 정상적으로 이루어지지 않을 경우는 CAN 버스에 연결된 모든 장치의 전원을 끄고 테스터로 통신 선로의 CAN_H와 CAN_L 단자간 저항을 측정해 봅니다. 양 단자간 저항이 60Ω (=120Ω+120Ω)이 되어야 합니다.

1.7 LED 상태 표시

USB2CAN은 3가지의 LED 표시등이 있습니다.

표 1-1 LED 색깔에 따른 기능

LED	Function
RUN (녹색)	Run/Error
RXD (파랑)	CAN RX
TXD (빨강)	CAN TX

USB2CAN의 USB 포트가 PC와 연결되면(전원이 공급되는 상황) 3개의 LED가 동시에 0.5초 동안 켜졌다가 꺼지며, 이후 동작상태를 표시하는 LED만 깜박이게 됩니다.

전원이 공급된 후, 3개의 LED가 모두 OFF 상태로 꺼져있거나 ON 상태로 켜져 있다면 USB2CAN이 정상적으로 초기화되지 않은 상태입니다. USB2CAN이 정상적으로 초기화 되었다면 동작상태를 표시하는 LED는 1초 주기로 깜박이게 됩니다.

CAN 포트로 데이터 전송 도중 에러가 발생하면 동작상태를 표시하는 LED가 0.2초 주기로 깜박이

게 됩니다. (빠른 주기로 LED가 깜박입니다.) 이러한 경우는 주로 USB2CAN의 CAN 포트가 CAN 네트워크에 연결되지 않아 전송 도중 에러가 발생한 경우입니다.

3개의 LED가 모두 0.25초 주기로 깜박이는 경우는 USB2CAN의 H/W나 S/W가 오동작을 일으켜 펌웨어 실행이 중단된 상황으로, 개발자에게 상황을 리포트 하여 문제를 수정해야 합니다.

2 포트 설정과 연결

2.1 CAN 포트

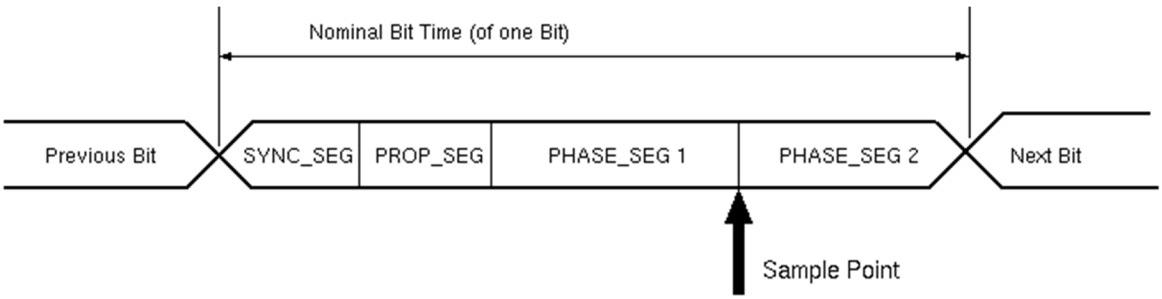
USB2CAN 컨버터의 CAN 포트를 CAN 네트워크에 연결하여 정상적으로 통신이 이루어 지려면 CAN 네트워크에 연결된 장치들의 CAN 포트 설정이 일치해야 합니다. USBCAN 컨버터의 CAN 포트에 관련된 파라미터는 다음과 같습니다:

- Bitrate: 1000K, 800K, 500K, 250K, 125K, 50K, 25K, 10K bps
- Filter identifier: Standard 인 경우 0 ~ 7FF, Extended인 경우 0 ~ 1FFFFFFF
- Filter mask: Standard 인 경우 0 ~ 7FF, Extended인 경우 0 ~ 1FFFFFFF
- Transfer mode: Disable, Binary, Text
  - Disable: CAN 메시지를 전송하지 않음
  - Binary: STX, ETX로 묶인 바이너리 형식의 패킷 포맷 사용
  - Text: 읽을 수 있는 텍스트 형식의 패킷 포맷 사용

2.1.1 CAN 속도(bitrate)

하나의 CAN bit에는 높은 수준의 fault tolerance를 보장하기 위해 sub-bit timing 설정이 사용됩니다. 하나의 CAN bit는 다음과 같이 4개의 세그먼트로 구분됩니다.

- SYNC_SEG – Bus에서 다양한 노드들을 동기화 하는데 사용
- PRPG_SEG – 물리적 지연보상 (physical bus와 internal CAN 노드 전달 지연)
- PHASE_SEG1, PAHSE_SEG2 – phase edge 오류를 보정하는데 사용



모든 CAN bit timing 계산은 최소 시간 단위로 정해진 time quanta(TQ)를 기반으로 합니다.

SYNC_SEG		1 TQ
PROP_SEG + PHASE_SEG1	BS1	1 ~ 16 TQ
PHASE_SEG2	BS2	1 ~ 8 TQ

CAN bus의 bitrate를 설정하기 위해서 BS1, BS2, SJW, Prescaler를 사용합니다. 여기서 SJW는 CAN bus의 시간 지연에 따른 보상시간 설정 값으로 실제 Bitrate 계산에서는 제외되며 BS1, BS2,

Prescaler 값이 사용됩니다. SJW는 보통 1로 설정하는데, 1로 설정하면 1us 이내의 시간지연에 대해서는 보상을 하게 됩니다.

표 2-1 USB2CAN 컨버터의 CAN Bitrate 설정에 사용된 bit timing:

Bitrate	Sampling point	BS1	BS2	SJW	Prescaler
1M	75%	2	1	1	9
800K	80%	3	1	1	9
500K	75%	2	1	1	18
250K	75%	5	2	1	18
125K	75%	5	2	1	36
50K	70%	6	3	1	72
25K	70%	6	3	1	144
10K	73.3%	10	4	1	240

- Clock = 36M
- Bitrate = Clock / Prescaler / (1 + BS1 + BS2)
- Sampling point = (1 + BS1) / (1 + BS1 + BS2)

Sampling point는 CAN bit의 0, 1을 판단하는 지점으로, 이 부분은 Bitrate를 계산하다 보면 달라질 수 있습니다. 따라서 CAN bus의 여러 장치간 통신에서 Bitrate는 반드시 일치시켜야 하지만 Sampling point는 차이가 나도 통신에는 문제가 없습니다.

2.1.2 CAN 필터

CAN Filter Identifier와 CAN Filter Mask를 설정하여 USB2CAN 컨버터로 받아들일 CAN 메시지의 ID를 결정한다. 필터 기능은 하드웨어적으로 수행됨으로 PC의 성능에 영향을 주지 않습니다.

Filter 설정 예) CAN ID가 0x030~0x03F 인 노드의 메시지만 수신하고자 할 때:

Bit	11	10	9	8	7	6	5	4	3	2	1
Mask (0x7F0)	1	1	1	1	1	1	1	0	0	0	0
Identifier (0x030)	0	0	0	0	0	1	1	X	X	X	X
Accept.	0	0	0	0	0	1	1	X	X	X	X

X는 0 혹은 1 중 아무 값

상기 표에서와 같이 설정된 CAN Filter Identifier와 수신된 CAN 메시지의 ID를 CAN Filter Mask에서 마스킹 된 부분만 비교하여 ID가 서로 일치하면 CAN 메시지를 받아들이게 됩니다.

2.2 가상 시리얼 포트 (USB2CAN-VCP 만 해당)

USB2CAN 컨버터는 PC와 USB 상의 VCP(가상 시리얼 포트)로 연결됩니다. USB2CAN 컨버터와 PC 간 정상적으로 통신이 이루어 지려면 USB2CAN 컨버터와 PC의 시리얼 포트 설정이 일치해야 합니다. USB2CAN 컨버터의 시리얼 포트 설정은 다음과 같습니다:

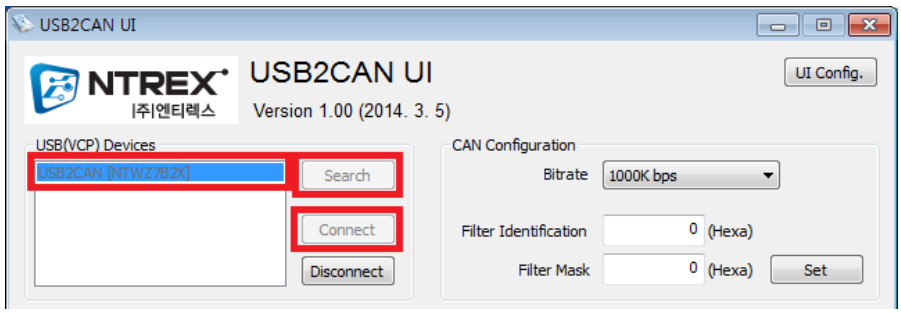
- Bits per second: 921600 bps
- Data bits: 8bits
- Parity: None
- Stop bits: 1bit
- Flow control: None

상기 값들은 USB2CAN-VCP 제품 생산 시 고정된 값으로 출하되며 사용자가 임의로 변경할 수 없습니다.

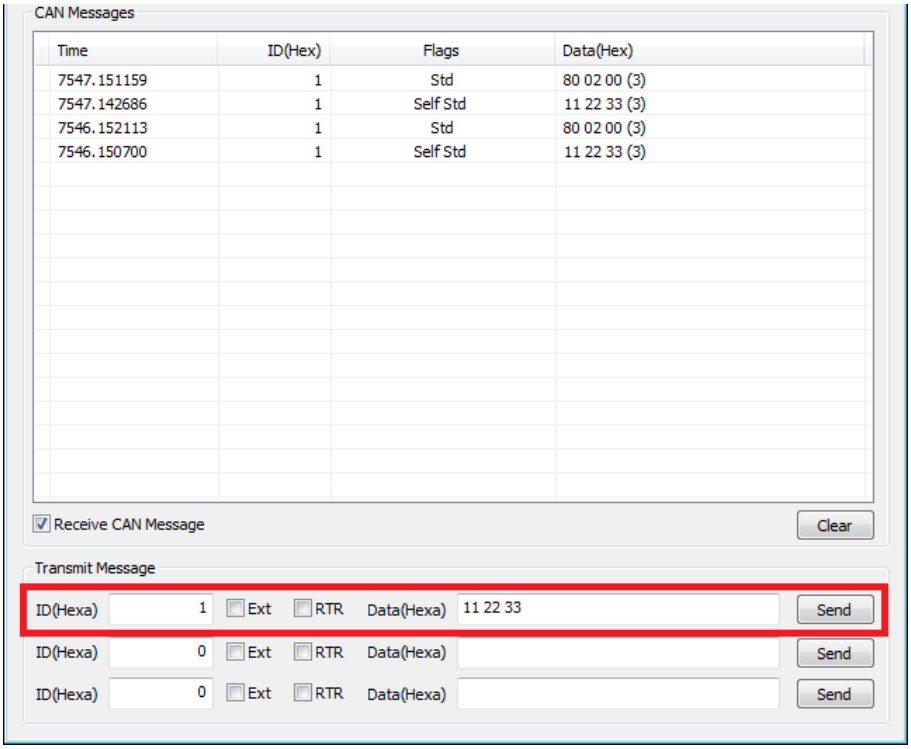
2.2.1 USB2CAN UI 유틸리티를 사용한 연결

USB2CAN 컨버터에 연결하는 가장 간단한 방법은 USB2CAN UI 유틸리티를 사용하는 것입니다. (USB2CAN UI 유틸리티에 대해서는 '4. USB2CAN UI 유틸리티'장에서 상세하게 설명합니다.) 이 과정에 앞서 USB2CAN 컨버터가 호스트 PC에 USB로 연결되어 있어야 하고, USB2CAN의 장치 드라이버가 설치되어 있어야 합니다. (장치 드라이버의 설치에 대해서는 '5. 장치 드라이버'장을 참조하기 바랍니다.)

다운로드 받은 UI 유틸리티를 실행합니다. 만일 USB2CAN 컨버터가 호스트 PC에 연결되어 있다면 다음 그림과 같이 USB(VCP) Devices 그룹에 연결된 컨버터가 표시됩니다. 만일 표시되지 않는다면, 컨버터가 USB 포트로 올바르게 연결되고 Run LED가 1초 주기로 깜박이는지 확인합니다. 그리고 장치관리자에서 장치 드라이버가 올바르게 설치되었는지 확인합니다.

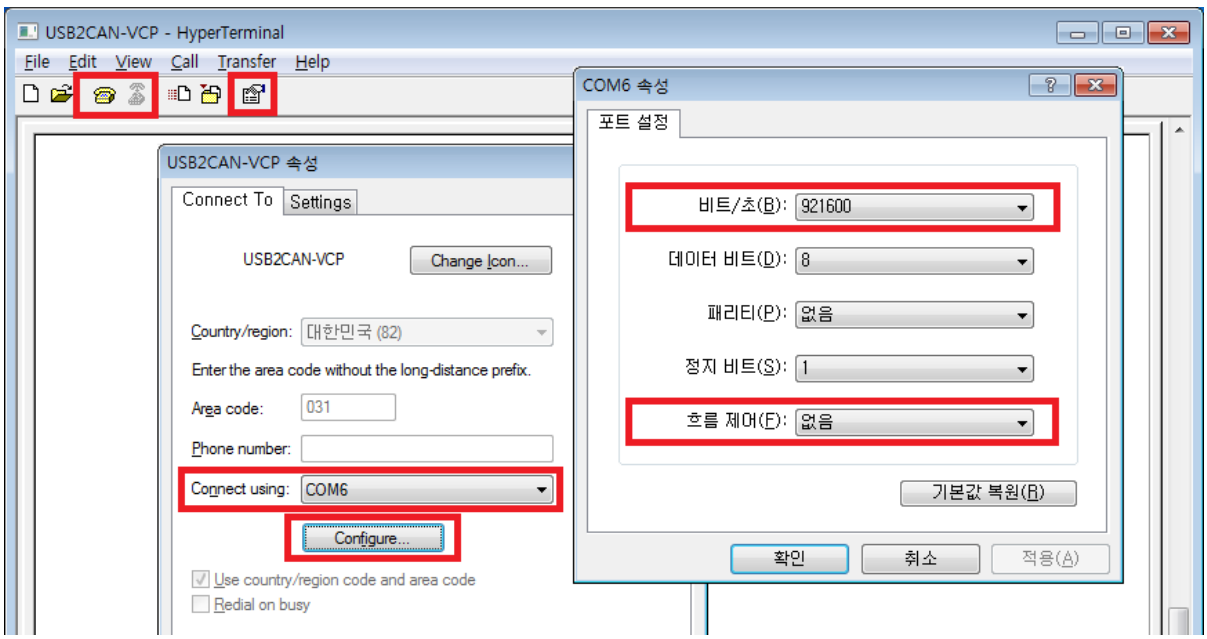


[Search] 버튼을 눌러 호스트 PC에 연결된 USB2CAN 컨버터를 검색합니다. 검색을 성공하면, 검색된 컨버터를 선택하고 [Connect] 버튼을 누릅니다. 연결을 성공하면, [Search] 버튼과 [Connect] 버튼은 비활성화 되고 하단의 [Send] 버튼으로 CAN 메시지를 주고받을 수 있게 됩니다.



2.2.2 Hyperterminal 을 사용한 연결

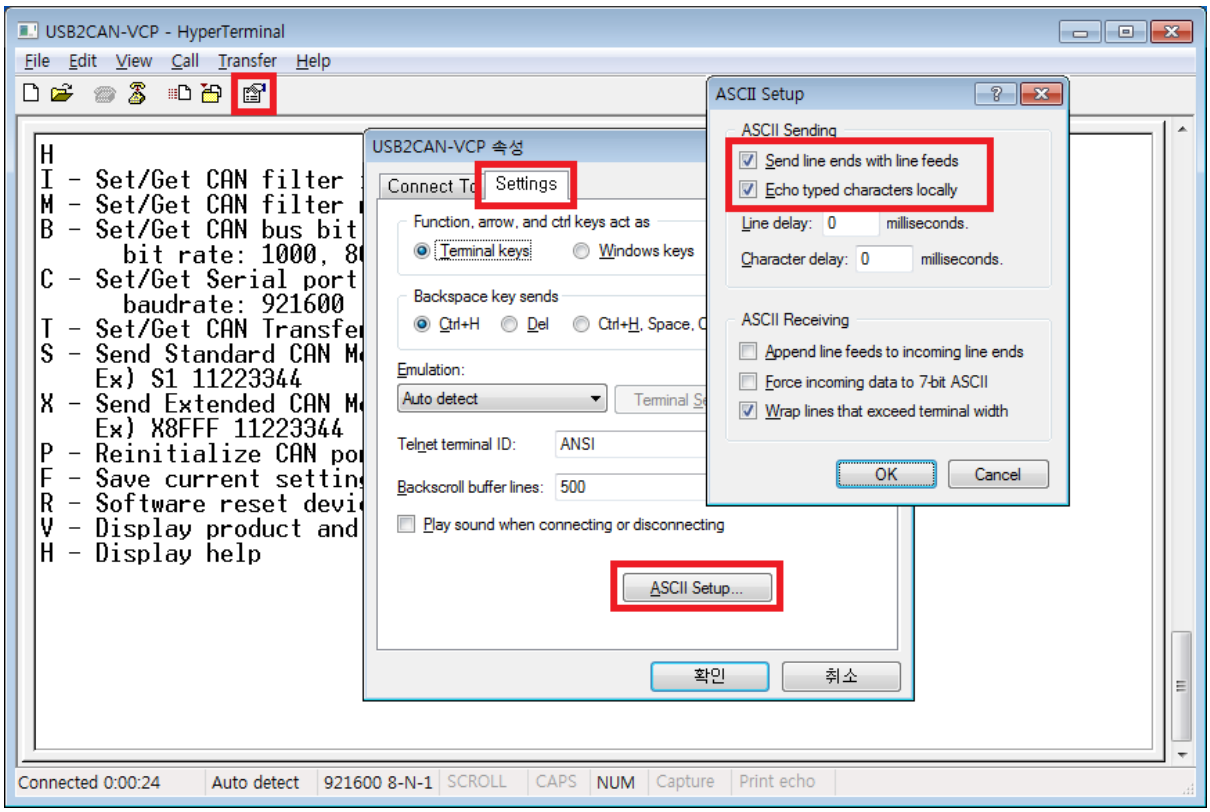
USB2CAN-VCP 모델은 호스트 PC에 가상 시리얼 포트(VCP; Virtual COM Port)를 통해 연결되기 때문에, Hyperterminal과 같은 시리얼 터미널 프로그램을 사용하여 연결할 수 있게 됩니다. 만일 사용자에게 익숙한 터미널 프로그램이 있다면, 이를 사용하여도 무방합니다. 여기서는 많이 알려진 Hyperterminal을 사용하여 USB2CAN 컨버터에 연결하는 방법을 소개합니다.



Hyperterminal을 실행하고 상단 툴바에서 연결이 끊어진 상태(전화기 모양의 아이콘에서 수화기가 놓인 모양의 아이콘이 활성화 되어 있음)를 확인합니다. 툴바에서 손모양의 Properties 아이콘을 클릭합니다. 그리고 다음 그림에서와 같이 USB2CAN-VCP가 연결된 COM 포트를 선택하고 [Configure...] 버튼을 누릅니다. COM? 속성 창에서 비트/초를 921600 으로 선택하고 흐름제어를 없음으로 선택합니다.

[확인]버튼을 눌러 대화상자를 모두 닫은 후, 상단 툴바의 전화기 모양의 Call 아이콘을 클릭합니다. 그러면 Hyperterminal은 USB2CAN 컨버터에 연결되어 컨버터의 각종 설정을 할 수 있고 CAN 버스에 연결된 다른 장치로 CAN 메시지를 주고받을 수 있게 됩니다.

Hyperterminal에서 사용자가 입력하는 문자를 화면에 표시하려면 다음 과정을 참조하기 바랍니다.



Hyperterminal 창의 툴바에서 속성(Properties) 아이콘을 클릭합니다. 그리고 다음 그림과 같이 ASCII Setup 창을 실행하여 'Send line ends with line feeds'와 'Echo typed characters locally' 체크박스를 체크합니다. 그리고 ASCII Receiving 그룹의 항목들은 그대로 둡니다.

이제 Hyperterminal에서 다음과 같은 명령을 입력해 봅니다. 명령은 USB2CAN 컨버터의 CAN 포트에 장치가 연결되어있지 않아도 실행 가능합니다.

```
V
H
B
T
```

그러면 다음과 같이 표시됩니다.

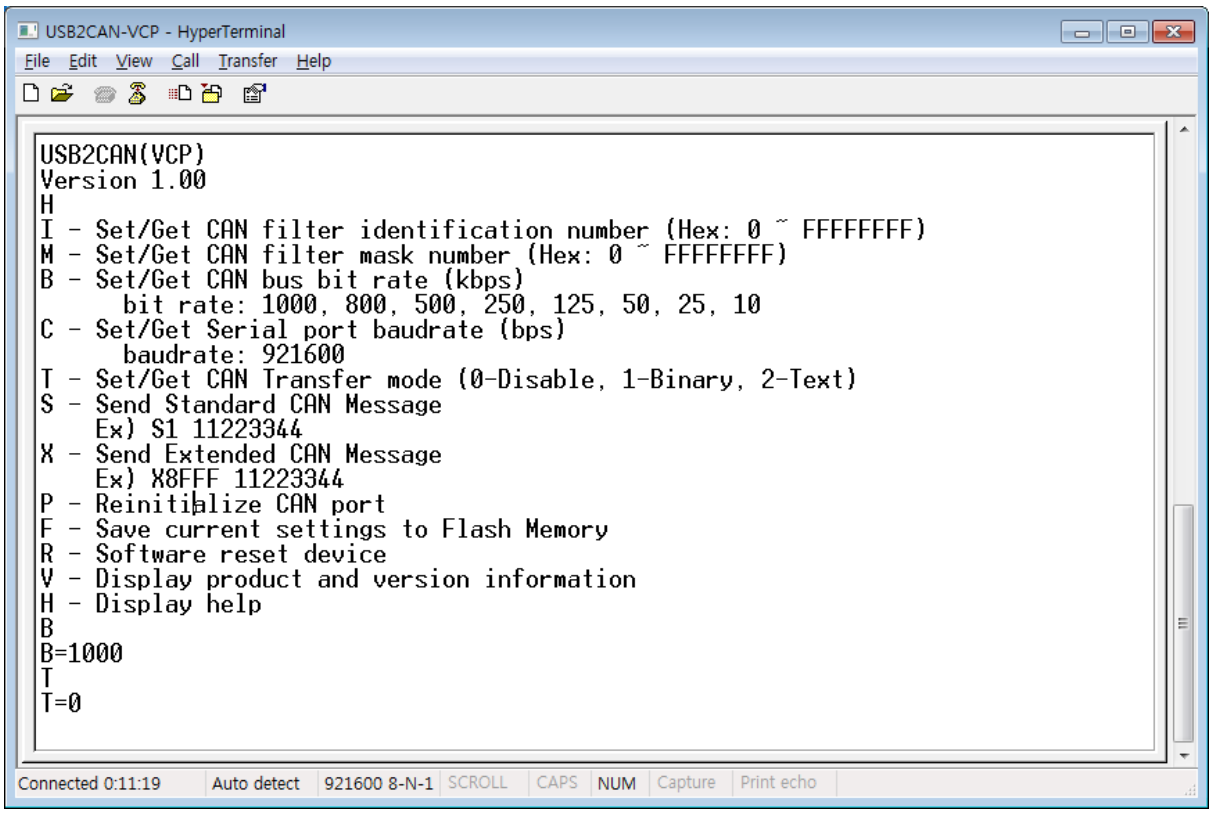


그림 2-1 Hyperterminal에서 텍스트 명령 전송과 회신

상기 명령은 USB2CAN의 Version 표시, 도움말 표시, CAN 통신 속도, CAN 전송 모드를 표시하는 명령입니다. 더 자세한 내용은 3장 통신 프로토콜을 참조하기 바랍니다.

※ Hyperterminal은 Windows XP까지 OS에 포함되어 있었으나, Windows Vista 부터는 기본적으로 제공되지 않습니다.

2.3 USB FIFO 포트 (USB2CAN-FIFO 만 해당)

'2.2.1 USB2CAN UI 유틸리티를 사용한 연결'절을 참조하기 바랍니다. USB2CAN-FIFO는 '2.2.2 Hyperterminal을 사용한 연결'절에서와 같이 일반적인 터미널 프로그램을 사용한 연결은 불가능합니다.

3 통신 프로토콜 (USB2CAN-VCP 만 해당)

USB2CAN-VCP 컨버터의 VCP(가상 COM 포트)로 CAN 메시지를 구성하는 방법은 바이너리와 텍스트 패킷 형식이 있습니다. 이번 장에서는 바이너리와 텍스트 패킷 기반의 통신 프로토콜에 대하여 설명합니다.

USB2CAN-VCP에서는 텍스트 패킷과 바이너리 패킷을 혼용하여 사용할 수 있습니다. 호스트 PC에서 컨버터로 텍스트 패킷을 전송한 경우, 이후 컨버터가 PC로 전송하는 패킷은 모두 텍스트 패킷이 됩니다. 호스트 PC에서 컨버터로 바이너리 패킷을 전송한 경우, 이후 컨버터가 호스트 PC로 전송하는 패킷은 모두 바이너리 패킷이 됩니다.

3.1 텍스트 패킷

텍스트 기반 명령은 명령에 대한 하나의 문자가 할당되어 있으며, 명령의 끝은 ↵ (ASCII 코드 13; 키보드의 Enter 키)으로 끝납니다. 텍스트 기반 명령은 CAN 메시지 전송을 제외하면 읽기와 쓰기 명령으로 구분됩니다. 먼저, 읽기 명령은 컨버터 내의 각종 변수를 읽는데 사용됩니다. 다음 예제를 보면 쉽게 이해할 수 있을 것입니다.

```
Ex) I↵
I=0↵
```

읽고자 하는 변수에 해당하는 문자를 입력하고 엔터를 치면, 변수의 값을 돌려줍니다.

쓰기 명령은 컨버터 내의 각종 변수에 대한 설정 값을 변경하는데 사용됩니다.

```
Ex) I=0↵
I=0↵
```

그러면 실제로 변수에 적용된 값을 되돌려줍니다.

3.1.1 텍스트 명령

다음 표 3-1은 USB2CAN 컨버터에서 사용하는 텍스트 패킷 명령에 대한 설명입니다.

표 3-1 텍스트 패킷 명령어

I	CAN Filter Identifier 값을 읽어오거나 설정합니다. 핵사 값으로 0x0 부터 0x1FFFFFFF 까지 사용합니다. 필터의 식별자 값을 변경한 경우에는 P 명령으로 CAN 포트를 초기화 하여야 바뀐 값이 적용됩니다. Ex) I↵ I=FF00↵

M	CAN Filter Mask 값을 읽어오거나 설정합니다. 핵사 값으로 0x0 부터 0x1FFFFFFF 까지 사용합니다. 필터의 마스크 값을 변경한 경우에는 P 명령으로 CAN 포트를 초기화 하여야 바뀐 값이 적용됩니다. Ex) M↵ M=FF00↵
B	CAN 버스의 통신 속도를 읽어오거나 설정합니다(단위: kBps). 통신 속도를 변경한 경우에는 P 명령으로 CAN 포트를 초기화 하여야 바뀐 값이 적용됩니다. • 10 - 10Kbps • 25 - 25Kbps • 50 - 50Kbps • 125 - 125Kbps • 250 – 250Kbps • 500 – 500Kbps • 800 - 800Kbps • 1000 – 1Mbps Ex) B↵ B=1000↵
C	시리얼 포트의 통신 속도를 읽거나 설정합니다(단위: Bps). (현재 시리얼 통신 속도는 921600bps로 설정되어 있으며 값을 바꿀 수 없음)
T	PC와 USB2CAN 컨버터간 메시지의 송수신 여부를 읽거나 설정합니다: • 0 – PC와 컨버터간 CAN 메시지 송수신 차단 • 1 – PC와 컨버터간 CAN 메시지 송수신 허용 (Binary 형태) • 2 - PC와 컨버터간 CAN 메시지 송수신 허용 (Text 형태) Ex) T↵ T=1↵
S	CAN 버스로 표준 CAN 메시지를 전송합니다. ('3.1.2 CAN 메시지'에서 자세히 설명) Ex) S1 11223344↵ S1 11223344AABBCCDD↵ S1R↵
X	CAN 버스로 확장 CAN 메시지를 전송합니다. ('3.1.2 CAN 메시지'에서 자세히 설명) Ex) X8FFF 11223344↵ X8FFF 11223344AABBCCDD↵ X8FFFR↵
P	CAN 버스의 통신속도, CAN 필터 값이 변경되었을 때 이 명령을 사용하여 CAN 포트를 초기화 합니다. Ex) P↵
F	설정 값들을 플래시 메모리에 저장 합니다. 만일 I, M, B, C, T 명령으로 설정 값을 변경하고 전원이 꺼졌다 켜졌을 때도 변경된 값이 유지되기를 원한다면 이 명령을 사용하여야 합니다. Ex) F↵

R	USB2CAN 어댑터를 소프트웨어적으로 리셋합니다. Ex) R↵
V	제품 정보와 버전 정보를 표시합니다. Ex) V↵
H	도움말을 표시합니다. Ex) H↵

3.1.2 CAN 메시지

CAN 메시지는 상기 예에서 든 I, M, B, T 등의 명령어 들과 구조가 다릅니다. CAN 장치와 CAN 메시지를 주고받는데 사용하는 명령은 S나 X로 시작하며 다음과 같은 구조를 가집니다. CAN 메시지도 다른 명령과 마찬가지로 끝은 ↵ 으로 끝납니다.

S X	ID	R Space	Data 1	Data 2	Data 3	...	↵

처음으로 오는 문자는 S 혹은 X인데, S는 표준(standard) CAN 메시지를 의미하고 X는 확장(extended) CAN 메시지를 의미합니다.

두 번째로는 CAN ID를 16진수(0~9, A~F)로 표시합니다. 표준 CAN 메시지의 경우 11bit의 ID를 사용하고 확장 CAN 메시지의 경우는 29bit의 ID를 사용할 수 있습니다.

- 표준 CAN (버전 2.0A): 11bit 식별자
- 확장 CAN (버전 2.0B): 29bit 식별자

세 번째로는 원격전송요청(RTR; Remote Transmission Request)을 표시하는데 R 이면 CAN 메시지가 RTR임을 의미합니다. 만일 공백(space) 이면 CAN 메시지가 데이터 프레임 이라는 것을 의미합니다. 원격 프레임은 데이터 버스상의 어떤 한 노드로부터 다른 노드로 데이터를 전송 요청할 때 사용 됩니다.

마지막으로 Data 1~8에는 각각 1byte 데이터를 16진수로 표시합니다. 전체 데이터는 0에서 8byte로 구성됩니다. 만일 8byte가 넘어가는 메시지를 구성하면, 넘어가는 메시지는 잘려나가게 됩니다.

3.2 바이너리 패킷

바이너리 패킷의 기본 구조는 명령어 패킷과 메시지 패킷으로 구분됩니다. 명령어 패킷은 USB2CAN 컨버터에 대한 명령 패킷이고, 메시지는 USB2CAN 컨버터에 연결된 장치로 CAN 메시지를 주고받기위한 패킷입니다.

3.2.1 명령어 패킷

명령어 패킷의 구조는 다음과 같습니다.

1byte	2byte	3byte	4byte	5~8byte	9~16byte	17byte	18byte
STX (0x02)	Type	Not used (0x00)	Command	Value	Not used (0x00)	Checksum	ETX (0x03)

(1) STX – 패킷 전송 시작 문자입니다(0x02).

(2) Type – 패킷의 종류 설정합니다:

- 0x00 – 최상위 비트가 0인 경우 Message Packet 임을 나타냄
- 0x80 – 최상위 비트가 1인 경우 Command Packet 임을 나타냄
- 0x10 – Error; Command Packet의 형식이 잘못되었거나 존재하지 않는 command를 전송한 경우에 대한 응답 패킷에 설정됨
- 0x08 – Command의 읽기 혹은 쓰기 속성을 표시;
  - 0인 경우 Read Command,
  - 1인 경우 Write command.
- 0x01 – Command에 대한 Acknowledge 임을 의미

(3) Command – 명령의 종류를 나타냅니다.

- 0 – USB2CAN 컨버터는 이 명령에 대하여 어떠한 작업도 수행하지 않으며, 단지 수신된 값을 그대로 회신함. 컨버터가 연결되어 동작 중인지 확인하기 위해 사용함
- 1 – CAN Filter Identifier를 읽거나 씀, 범위: 0x0 ~ 0x1FFFFFFF
- 2 – CAN Filter Mask를 읽거나 씀, 범위: 0x0 ~ 0x1FFFFFFF
- 3 - CAN 통신속도 읽기 (단위: kBps), 10K Bps ~ 1M Bps까지 설정 가능함
- 4 – 시리얼 포트 통신속도 읽기 (단위: Bps), 921600 Bps로 고정되어 있으며 바꿀 수 없음
- 5 – PC와 USB2CAN 컨버터간 메시지의 송수신 여부를 읽거나 설정함:
  - 0 – PC와 컨버터간 CAN 메시지 송수신 차단
  - 1 – PC와 컨버터간 CAN 메시지 송수신 허용 (Binary 형태)
  - 2 – PC와 컨버터간 CAN 메시지 송수신 허용 (Text 형태)
- 9 - CAN을 새로 초기화 함, CAN Filter identifier, Filter Mask 등이 변경된 경우 이를 적용하기 위해서는 CAN을 새로 초기화 하는 명령을 수행해야 함
- 10 – 설정 값을 Flash Memory에 기록함, 설정 값을 Flash Memory에 기록하지 않은 경우에는 컨버터가 재시작 되면서 설정 값을 잃어버리게 됨
- 11 – 컨버터의 CPU를 소프트웨어적으로 리셋 함, 컨버터의 펌웨어는 처음부터 실행 됨
- 20 – 컨버터의 버전을 읽음

(4) Value – Write command일 경우와 Read command에 대한 응답일 경우에 설정됩니다. Read

command일 경우는 무시됩니다(0으로 초기화).

(5) Checksum – 패킷의 2byte에서 16byte까지 모두 더한 값입니다.

(6) ETX – 패킷 전송 종료 문자입니다(0x03).

3.2.2 메시지 패킷

메시지 패킷의 구조는 다음과 같습니다.

1byte	2byte	3byte	4byte	5~8byte	9~16byte	17byte	18byte
STX (0x02)	Type (0x00)	DLC	Flags	ID	Data 1~8	Checksum	ETX (0x03)

(1) STX – 패킷 전송 시작 문자입니다(0x02).

(2) Type – 패킷의 종류를 설정합니다. 여기서는 Message Packet 이므로 0x00으로 고정됩니다.

- 0x00 – Message Packet

(3) DLC – 데이터의 길이를 나타냅니다. 이 값은 0 ~ 8의 범위가 됩니다.

(4) Flags – RTR, Standard/Extanded 패킷임을 나타냅니다:

- 0x20 – RTR (Remote transmission request)
- 0x40 – CAN Extended Data Frame을 나타냄
- 0x00 - 경우 CAN Standard Data Frame을 나타냄

(5) ID – CAN 식별자입니다:

- CAN Standard Frame Format (CAN 2.0 A)인 경우: 0x0 ~ 0x7FF (11bit)
- CAN Extended Frame Format (CAN 2.0 B)인 경우: 0x0 ~ 0x1FFFFFFF (29bit)

(6) Data 1~8 – CAN 데이터입니다. DLC가 8보다 작을 때 남는 데이터 영역에는 0x00을 채워야 합니다.

(7) Checksum – 패킷의 2에서 16byte까지 모두 더한 값입니다.

(8) ETX – 패킷 전송 종료 문자입니다(0x03).

* Remote frame이란: 일반적으로 데이터 전송은 센서와 같은 데이터를 발생시키는 노드에 의해 자동으로 Data Frame을 전송되게 됩니다. 어쨌거나, 데이터를 수집하는 노드가 Remote Frame을 보냄으로 데이터 전송을 요청하는 것도 가능합니다.

4 USB2CAN UI 유틸리티

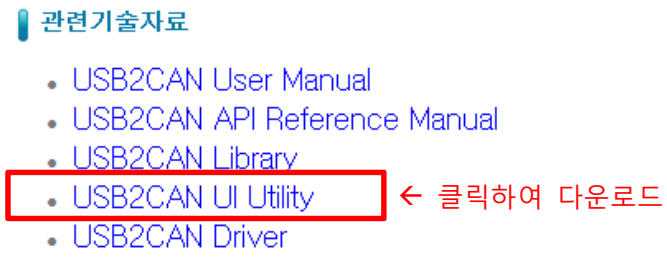
USB2CAN UI 유틸리티는 USB2CAN 컨버터의 각종 설정 및 USB2CAN 컨버터를 통해 연결된 다른 장치들과 CAN 메시지를 주고받을 수 있습니다.

4.1 다운로드 및 실행

USB2CAN 판매페이지 하단의 USB2CAN UI Utility 자료를 다운로드 합니다.

- 디바이스마트 제품구매 페이지로 이동: https://www.devicemart.co.kr/1111711





다운로드 한 파일은 zip으로 압축되어 있습니다. 압축을 풀면 exe 파일이 있는데, 이를 적당한 폴더에 옮겨두고 마우스로 더블클릭하여 실행하면 됩니다. USB2CAN UI 유틸리티는 따로 설치과정이 필요하지 않습니다.

4.2 메인 화면

USB2CAN UI 유틸리티의 메인 화면은 다음과 같습니다. 상단은 호스트 PC에 연결된 USB2CAN 컨버터 목록을 표시하고, 중앙에는 오고가는 CAN 메시지를 표시합니다. 하단에는 선택된 컨버터에 보내는 CAN 메시지를 편집합니다.

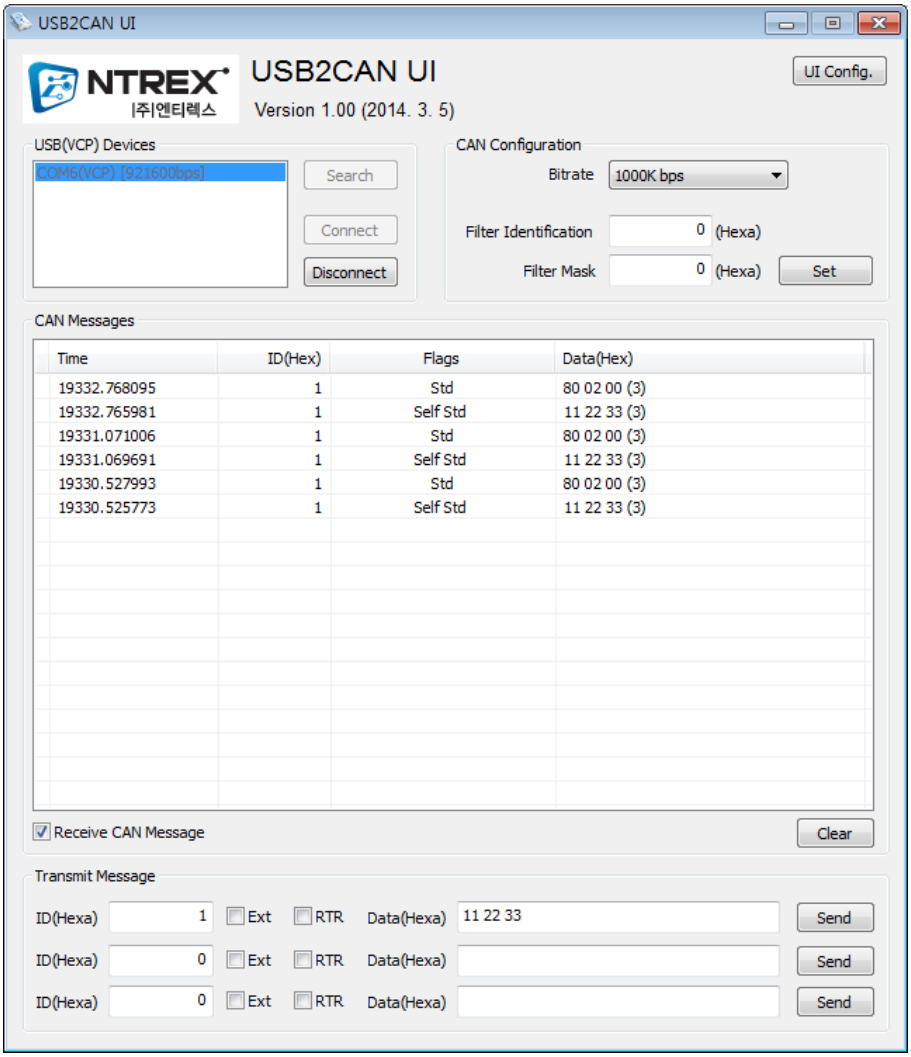


그림 4-1 USB2CAN UI 메인 화면

4.3 UI 설정 (USB2CAN-FIFO 만 해당)

CAN2USB UI와 관련된 설정을 하려면 메인 화면 우측 상단의 [UI Config.] 버튼을 누릅니다.

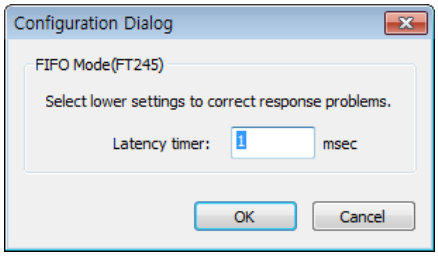


그림 4-2 Configuration Dialog 창

여기서 Latency timer는 내부 FIFO 큐에 CAN 메시지가 수신되었는지를 체크하는 시간입니다. CAN 메시지의 응답 지연이 문제되는 경우 낮은 값을 설정하여 해결할 수 있습니다. 0인 경우는 수신버퍼의 데이터를 체크하는 주기를 지연시간 없이 바로 처리함으로 응답 시간은 빨라지지만 CPU 자원을 많이 소비하게 됩니다.

4.4 연결

좌측 상단의 [Search] 버튼을 누르면 PC에 연결된 USB2CAN 컨버터를 검색하여 사용 가능한 장치를 리스트 박스에 표시합니다. 만일 하나 이상의 컨버터가 연결되어 있다면 아래 그림과 같이 연결 가능한 장치를 표시합니다. 여기서 연결할 장치를 선택한 후 [Connect] 버튼을 누릅니다.

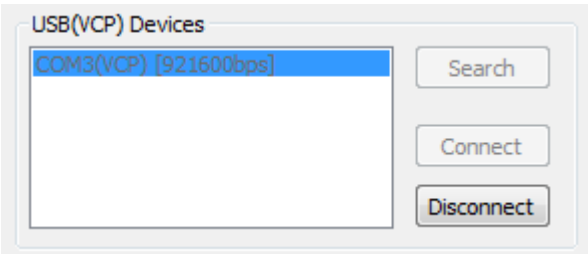


그림 4-3 USB2CAN 컨버터 목록 표시와 연결

만일 새로운 USB2CAN 컨버터가 PC에 연결되었다면 [Disconnect] 버튼을 눌러 현재 연결을 해제하고 [Search] 버튼을 누릅니다. 그러면 연결 가능한 컨버터 목록이 새롭게 갱신됩니다.

4.5 CAN 설정

연결이 정상적으로 이루어진 경우, 아래 그림과 같이 우측 상단의 CAN Configuration 데이터가 채워지며 [Set] 버튼이 활성화 됩니다. CAN Configuration 데이터는 USB2CAN 컨버터의 Flash Memory에 저장되어 전원이 꺼진 경우에도 지워지지 않고 보관됩니다.

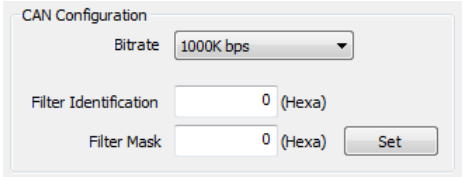


그림 4-4 CAN 포트 설정

ID와 MASK는 USB2CAN 컨버터가 받아들일 CAN Identifier와 Mask를 설정합니다. 상기 그림에서는 ID와 MASK가 둘 다 0으로 설정되어 있는데, 이러한 경우는 모든 CAN 메시지를 수신합니다. 다른 예로 ID 0x10부터 0x1F까지의 노드가 보내는 데이터만 받아들이고자 할 경우는 ID 0x10, MASK 0x10으로 설정하면 됩니다. (자세한 설명은 '2.1.2 CAN 필터'절을 참고하기 바랍니다.)

4.6 송수신 메시지 표시

CAN 네트워크를 통해 송수신되는 메시지는 다음 그림과 같이 메인 화면의 중앙에 있는 CAN Message 그룹에 표시됩니다.

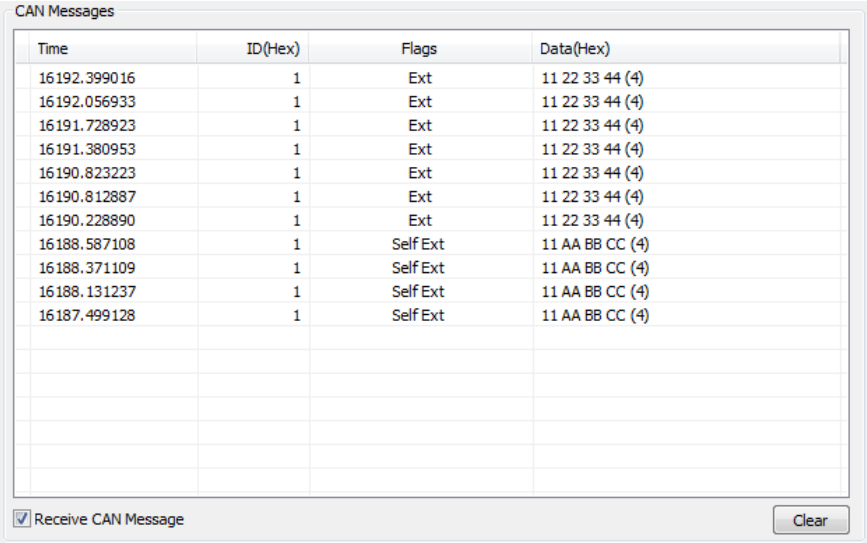


그림 4-5 CAN Messages 그룹

'Receive CAN Message' 체크박스는 USB2CAN 컨버터가 UI 프로그램으로 CAN 메시지를 전송할지 여부를 결정합니다.

[Clear] 버튼은 표시된 모든 CAN 메시지를 삭제합니다.

표시되는 CAN 메시지의 구조는 다음 표 4-1을 참조하기 바랍니다.

표 4-1 CAN 메시지 구조

Time	프로그램이 실행된 이후의 초단위 시간을 표시합니다. 23928.9748로 표시된 경우 23928초 974800마이크로초를 의미합니다.
ID	CAN ID를 16진수로 표시합니다.
Flags	CAN 메시지의 상태를 표시합니다: Self – USB2CAN 컨버터가 내보낸 메시지, Std – Standard,

	Ext – Extended, RTR – 리모트 프레임.
Data	CAN 메시지의 데이터를 16진수로 표시합니다. 괄호 속에는 데이터 길이를 표시합니다.

4.7 메시지 전송

CAN 네트워크에 연결된 장치로 메시지를 보내기 위해서는 하단의 Transmit Message 부분을 사용합니다. 아래 그림과 같이 각각 3종류의 메시지를 미리 설정해 두고 보낼 수 있습니다.

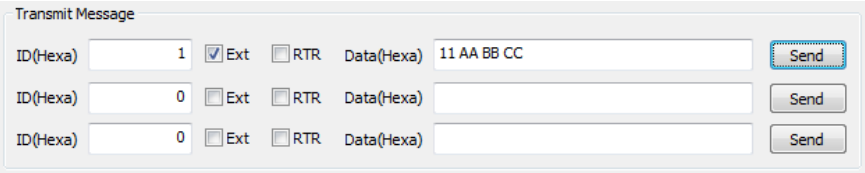


그림 4-6 Transmit Message 창

전송되는 메시지의 CAN ID 및 상태, 데이터는 다음 설명에 따라 입력합니다:

- ID 입력 박스에는 메시지를 수신할 제어기의 CAN ID를 ASCII 16진수로 입력함
- 체크박스 Ext는 CAN 2.0B(Extended)를 의미함, 체크 해제 시 CAN 2.0A(Standard)
- 체크박스 RTR은 리모트 프레임 요청을 의미함
- Data 입력 박스에는 제어기로 보낼 데이터를 ASCII 16진수로 입력함.

메시지의 구성이 끝났으면 [Send] 버튼을 눌러 제어기로 CAN 메시지를 보냅니다.

데이터 입력 형식은 다음과 같이 사용할 수 있습니다.

- 1 2 3 AA BB – 1byte씩 끊어서 입력
- 01 02 03 AA BB – 1byte씩 끊어서 입력
- 010203AABB – 모든 바이트를 연결하여 입력

상기 메시지는 모두 동일한 메시지이며, 입력된 데이터에서 8byte를 초과하는 데이터는 전송되지 않습니다.

5 장치 드라이버

호스트 PC에서 USB2CAN 컨버터를 사용하기 위해서는 USB2CAN 장치 드라이버가 설치되어야 합니다. 이 장에서는 USB2CAN 장치 드라이버를 설치하는 과정에 대해 설명합니다.

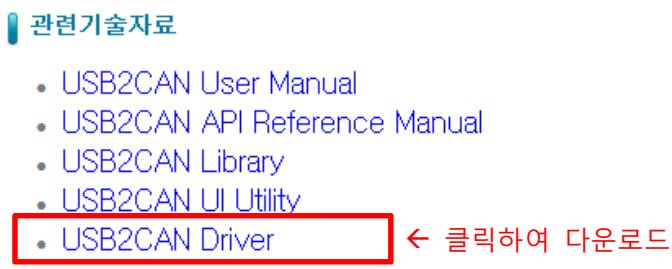
5.1 장치 드라이버 다운로드

USB2CAN-VCP와 USB2CAN-FIFO 컨버터는 동일한 장치 드라이버를 사용합니다. USB2CAN-VCP의 경우, 추가적으로 VCP 드라이버가 설치되어야 합니다.

두 컨버터의 장치 드라이버는 디바이스마트 제품구매 페이지에서 다운로드가 가능합니다.

- 디바이스마트 제품 구매 페이지로 이동: https://www.devicemart.co.kr/1111711





상기 페이지에서 USB2CAN Driver 자료를 클릭하여 다운로드 하고, 적당한 폴더에 압축을 풀어 놓습니다.

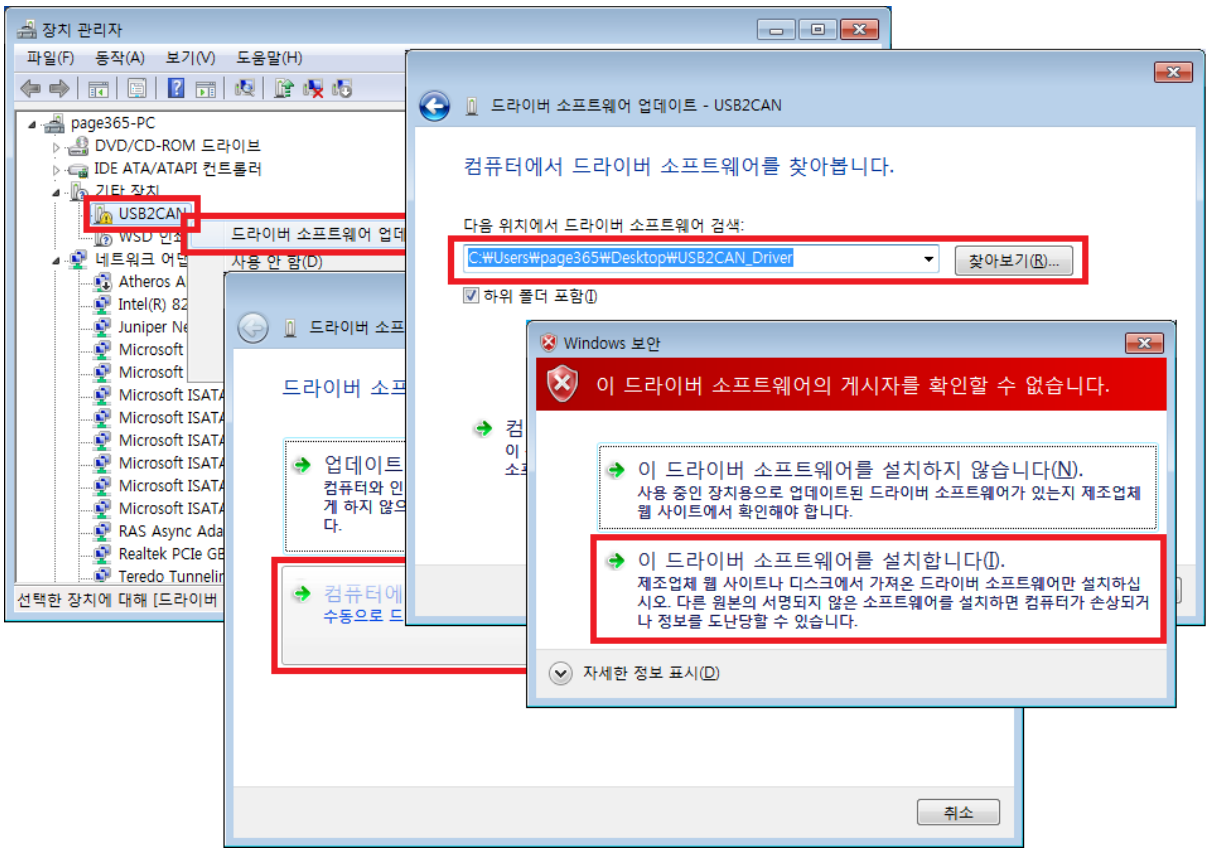
USB2CAN-FIFO 컨버터를 처음으로 호스트 PC에 연결하면, 올바른 장치 드라이버를 찾지 못하여 장치 드라이버가 제대로 설치되지 않습니다. 이때 다운로드 받은 장치 드라이버의 경로를 지정하

여 장치 드라이버를 수동으로 설치하여야 합니다. 만일 한 번 이상 장치 드라이버를 설치하였다면 다음 연결에서는 장치 드라이버를 새로 설치하지 않아도 됩니다.

다음 절에서 USB2CAN-FIFO와 USB2CAN-VCP 컨버터의 장치 드라이버를 설치하는 과정에 대하여 각각 설명합니다.

5.2 USB2CAN-FIFO 드라이버 설치

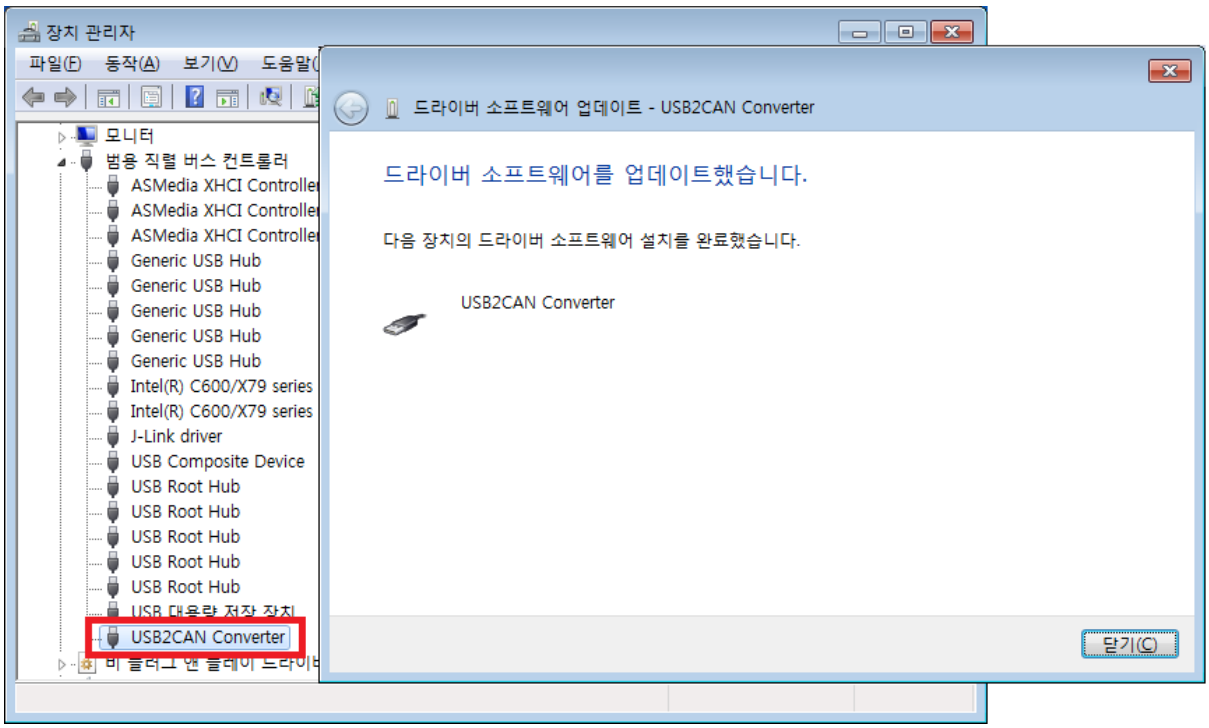
제어판에서 장치 관리자를 오픈합니다. 여기서 기타 장치 그룹에 가면 USB2CAN(VCP)에 느낌표 아이콘이 표시되어 있습니다. USB2CAN(VCP) 항목에 마우스 커서를 위치시키고 마우스 오른버튼을 눌러 팝업 메뉴를 실행합니다. 그리고 드라이버 소프트웨어 업데이트 메뉴를 선택합니다.



드라이버 소프트웨어 업데이트 창에서 '컴퓨터에서 드라이버 소프트웨어 찾아보기(R)'을 클릭합니다. 그리고 [찾아보기(R)...] 버튼을 눌러 다운로드한 드라이버의 압축을 푼 폴더를 지정합니다. 그리고 [다음] 버튼을 누르면 드라이버 소프트웨어의 설치 여부를 물어봅니다. 이때 '이 드라이버 소프트웨어를 설치합니다.'를 선택합니다.

그러면 다음과 같이 '드라이버 소프트웨어를 업데이트했습니다.'를 표시하는 메시지 창이 뜨고, 장

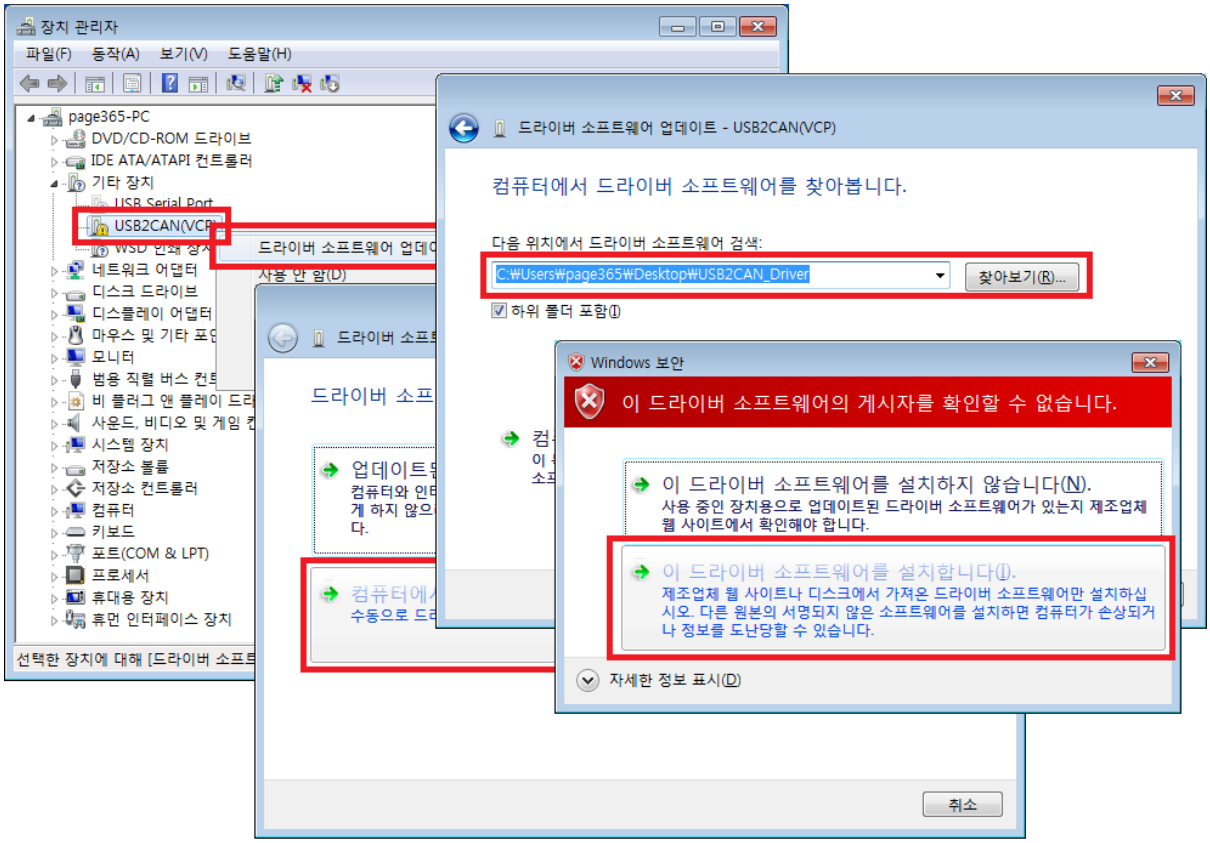
치 관리자의 범용 직렬 버스 컨트롤러에 USB2CAN Converter가 설치된 것을 볼 수 있습니다.



※ 장치 드라이버를 설치할 때는 USB2CAN-FIFO 컨버터가 호스트 PC에 연결되어 있어야 합니다.

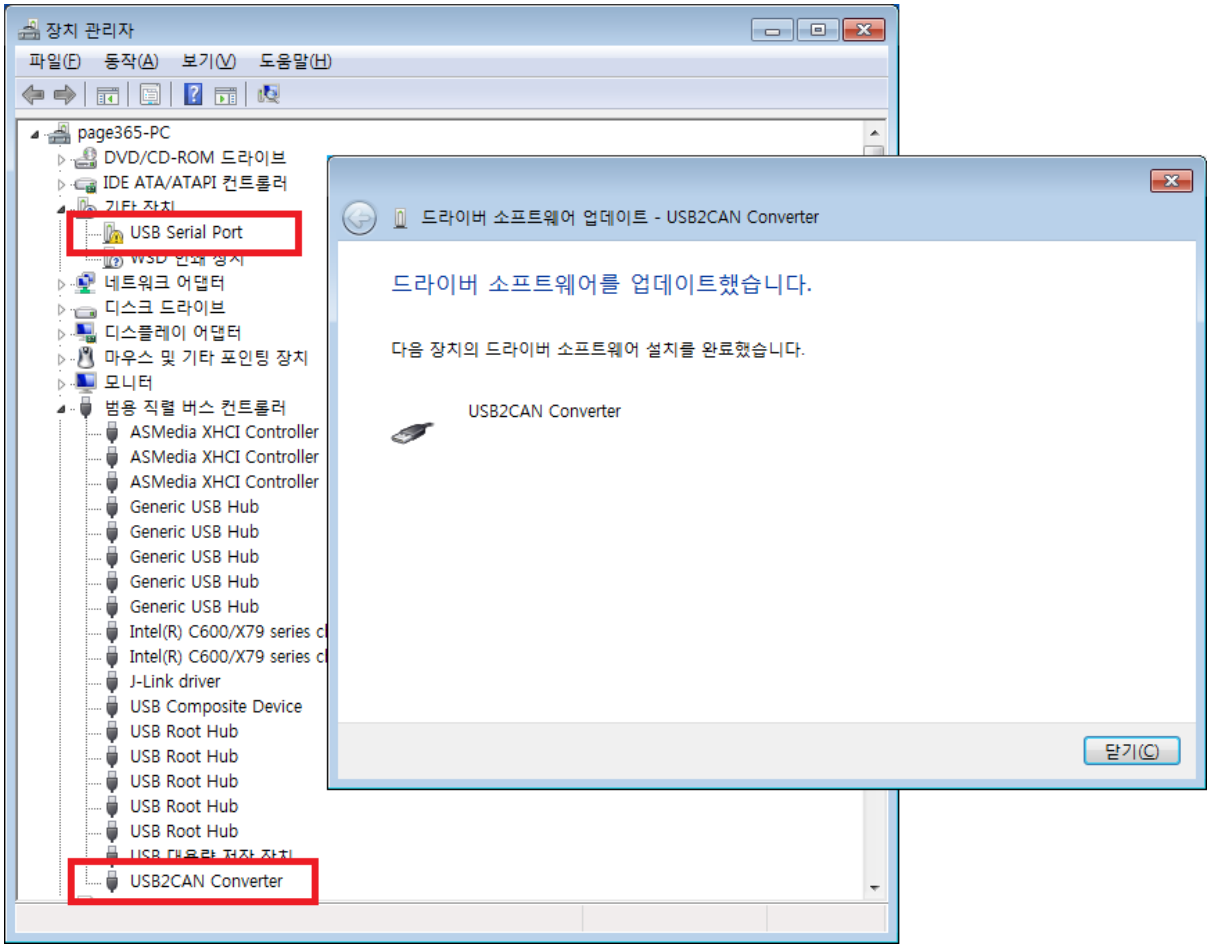
5.3 USB2CAN-VCP 드라이버 설치

제어판에서 장치 관리자를 오픈합니다. 여기서 기타 장치 그룹에 가면 USB2CAN(VCP)에 느낌표 아이콘이 표시되어 있습니다. USB2CAN(VCP) 항목에 마우스 커서를 위치시키고 마우스 오른버튼을 눌러 팝업 메뉴를 실행합니다. 그리고 드라이버 소프트웨어 업데이트 메뉴를 선택합니다.



드라이버 소프트웨어 업데이트 창에서 '컴퓨터에서 드라이버 소프트웨어 찾아보기(R)'을 클릭합니다. 그리고 [찾아보기(R)...] 버튼을 눌러 다운로드한 드라이버의 압축을 푼 폴더를 지정합니다. 그리고 [다음] 버튼을 누르면 드라이버 소프트웨어의 설치 여부를 물어봅니다. 이때 '이 드라이버 소프트웨어를 설치합니다.'를 선택합니다.

그러면 다음과 같이 '드라이버 소프트웨어를 업데이트했습니다.'를 표시하는 메시지 창이 뜨고, 장치 관리자의 범용 직렬 버스 컨트롤러에 USB2CAN Converter가 설치된 것을 볼 수 있습니다.



이후, 상기와 같은 과정으로 다시 한번 'USB Serial Port'의 장치 드라이버를 설치합니다.

최종적으로 다음과 같이 USB2CAN의 가상 시리얼 포트가 설치됨을 확인할 수 있습니다.



※ 장치 드라이버를 설치할 때는 USB2CAN-FIFO 컨버터가 호스트 PC에 연결되어 있어야 합니다.

관련 자료

아래 홈페이지에서 MoonWalker USB2CAN 관련 문서와 자료를 확인하실 수 있습니다.

- 엔티렉스:
  http://www.ntrexgo.com/

- 디바이스마트:
  http://www.devicemart.co.kr/

- MoonWalker USB2CAN (FIFO) 판매페이지:
  http://www.devicemart.co.kr/goods/view.php?seq=1111719

- MoonWalker USB2CAN (VCP) 판매페이지:
  http://www.devicemart.co.kr/goods/view.php?seq=1111711

- MoonWalker 소개:
  http://www.ntrexgo.com/moonwalker-%EC%86%8C%EA%B0%9C

- MoonWalker 판매페이지:
  http://devicemart.co.kr/goods/brand.php?seq=1475

- MoonWalekr UI Utility:
  http://www.ntrexgo.com/archives/19482

- MoonWalker 자료:
  http://www.ntrexgo.com/archives/category/ntrex-lab/moonwalker

- MoonWalker 예제:
  http://www.ntrexgo.com/archives/category/ntrex-lab/moonwalker_applications

- MoonWalker 액세서리:
  http://www.ntrexgo.com/archives/category/moonwalker_accessory

문서 변경 이력

Data	Version	Charges
2014.03.25	1.00	- 첫 출시
2014.07.28	1.10	- 목차 순서 편집 - 포트 설정과 연결 내용 추가 - 장치 드라이버 장 분리 및 내용 추가

제품의 보증

1. 본 제품은 엄정한 품질관리 및 검사과정을 거쳐서 만들어 진 제품입니다.
2. 제품 구입 후 6개월 이내에 제품 고장 발생 시에 무상으로 A/S를 해드립니다.
3. 정상적인 사용 상태에서 고장이 발생하였을 경우 보증기간 동안은 무상으로 A/S를 해드립니다.
4. 제품 보증기간이 경과한 후에 고장이 발생할 경우 유상으로 A/S를 해드립니다.
5. 보증기간 이내라 하더라도 본 보증 이내의 유상 서비스 안내에 해당되는 경우 서비스 따라 유상으로 A/S를 해드립니다.
6. 오용, 남용 및 인가되지 않은 인력에 의한 수리, 부적절한 보관상태 자연 재해로 인한 파손은 유상으로 A/S를 해드립니다.
7. 고객 변심 또는 구매 후 7일 이후에는 반품이 되지 않습니다.

회 사 명	(주)엔티렉스
본 사 주 소	인천 남구 주안동 5-38 (주)엔티렉스
전 화 번 호	070 - 7019 - 8887
팩 스 번 호	02 - 6008 - 4953
E – Mail	기술문의 - lab@ntrex.co.kr 영업문의 - stock@ntrex.co.kr
홈 페 이 지	www.ntrexgo.com www.devicemart.co.kr