

CANdy

(CAN Handy Terminal)

User Handbook



Authors:	Haemyung Lee, Hyunseoung Cho		
Company:	HanilProTech		
Version	1.0e	Date:	2008-05-30
Compled:	Kyungsuk Jun		2008-05-28
Verified:	Haemyung Lee, Hyunseoung Cho		2008-05-30
Pages:	44		

Revision History

Revision	Autor	Date	Description
0.98	Haemyung Lee	2008-04-16	Last update before release.
1.0	Kyungsuk Jun	2008-05-27	Released
1.0e	Kyungsuk Jun	2008-05-28	English version created and released.

Contents

2 Brief Instruction for Configuration Data	7
3 Use Instruction of CANdy	11
3.1 Main menu	11
3.1.1 I-Generator (Interactive Generator)	11
3.1.2 Generator	14
3.1.3 Memory	16
3.1.4 Setup	16
3.1.5 Info	17
3.1.6 Download	17
3.2 Message Transfer	18
3.2.1 Periodic Message	18
3.2.2 OnEvent Message	19
3.2.3 Periodic & OnEvent Message	19
3.2.4 Message Transfer according a Sequence List	20
3.3 Message Reception	21
3.3.1 Graph Window	21
3.3.2 Remarks to Message Transmission	22
3.3.3 Attributes of Message and Signal	24
3.3.4 Change of Attributes	25
3.4 Examples	27
3.4.1 OnEvent	27
3.4.2 OnEvent & Repetition	27
3.4.3 Periodic	28
3.4.4 Periodic & OnEvent	28
3.4.5 Periodic & OnEvent with Repetition	29
3.4.6 Button Type	30
3.4.7 Signal Value Reset of Periodic & OnEvent Message	31
4 Use Instruction of CANdyEditor	33
4.1 Installation of CANdyEditor	33
4.2 Run CANdyEditor	34
4.3 Main Menu	35
4.4 Load CAN Database	35
4.5 Verification of USB Port	38
4.6 Configuration of USB and CAN Interface	39

4.7 Edit Configuration Data	41
4.8 Download of Configuration Data	42
5 Trouble Shooting	43
6 Appendix.....	44
6.1 Specification.....	44
6.2 Components to be delivered.....	44

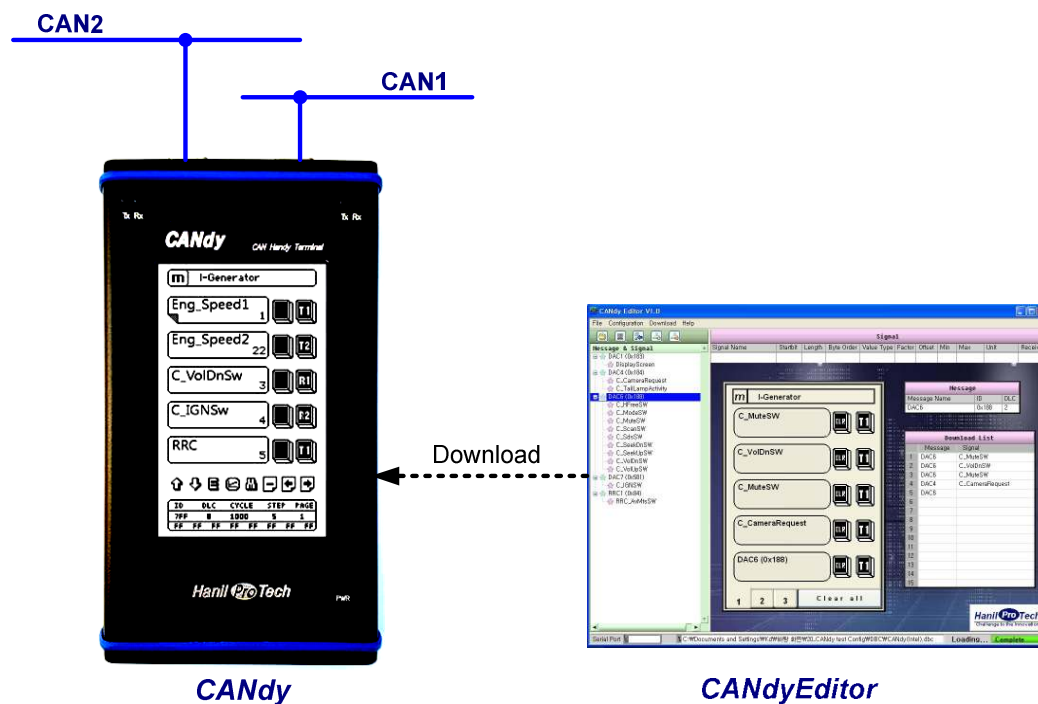
1 Introduction

CANdy is a handheld CAN terminal used to monitor the CAN communication as well as to transmit CAN messages.

CANdy is therefore predestinated for analyzing CAN messages and testing ECU functions in a real process.

CANdy provides 2 CAN interfaces **CAN1** and **CAN2**. These 2 CAN interfaces operate independently and can be chosen between high-speed CAN and fault-tolerant, low-speed CAN. The single-wire CAN can be also delivered on demand.

For configuration of CAN messages and signals for use in **CANdy**, the PC-based editor program **CANdyEditor** is provided.



CANdyEditor is used to create configuration data in comfortable way.

The configuration can be done on base of CAN database.

Up to 75 messages or signals on the whole can be defined in **CANdyEditor**.

The defined messages and signals are downloaded to **CANdy** via USB interface and stored in 5 memory slots of **CANdy**. Each memory slot contains 15 message or signal definitions.

The configuration of CAN messages and signals can be also done within **CANdy**.

CANdy supports various message send types:

- *OnEvent*
- *Periodic*
- *Periodic & OnEvent*
- *Repetition*

A short instruction for usage of **CANdy** and **CANdyEditor** are shown in

Chapter 2 : Brief Instruction for Configuration Data.

Also detail instructions are given in

Chapter 3 : Use Instruction of **CANdy**

and

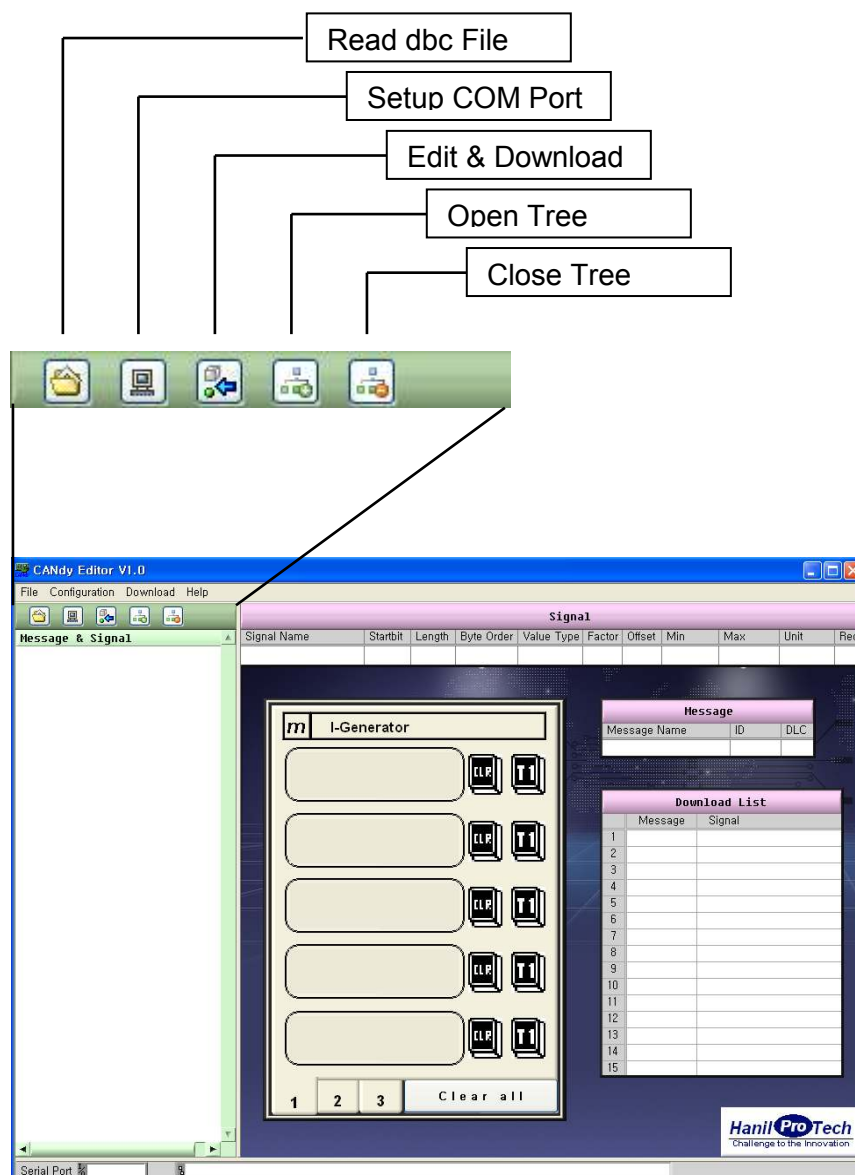
Chapter 4 : Use Instruction of **CANdyEditor.**

In **Chapter 5** some useful instructions are given in case of trouble occurrence.

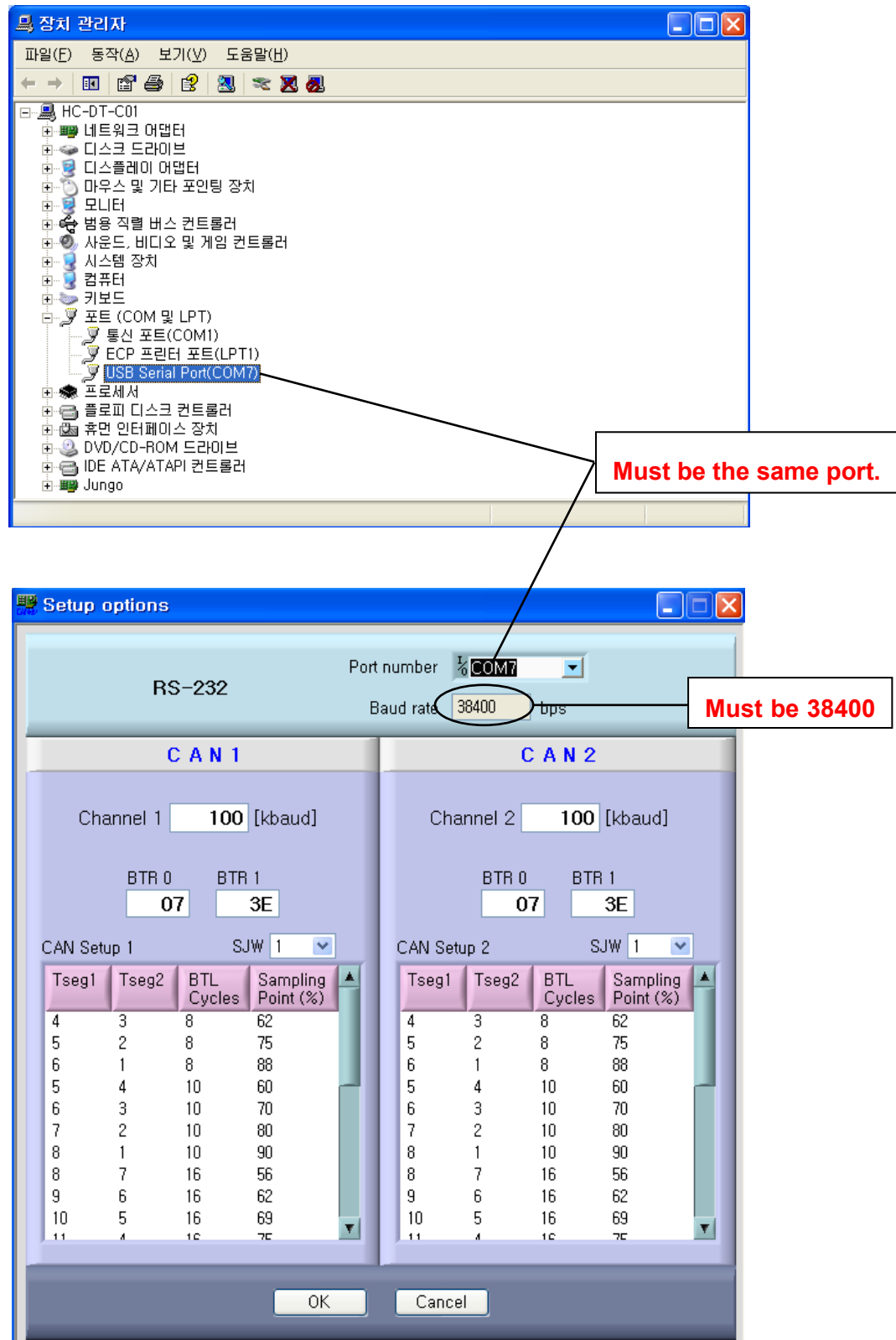
2 Brief Instruction for Configuration Data

In following the configuration procedure based on CAN database (*.dbc) and the download procedure are briefly described for easy startup with **CANdy**.

- 1) Start **CANdyEditor** (Refer to **4.1 Installation of CANdyEditor** and **4.2 Run CANdyEditor**)
- 2) Power on **CANdy**
- 3) Select [Download] in main menu of **CANdy**
- 4) Return to **CANdyEditor**

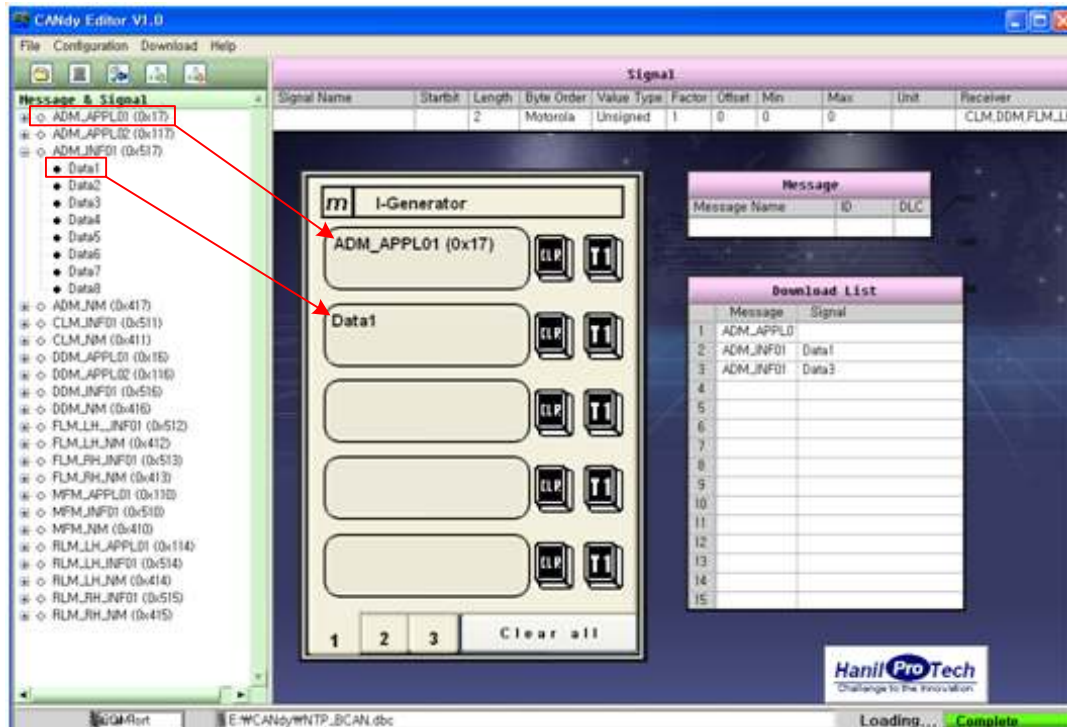


5) Installation and Setup USB port (Refer to 4.5 Verification of USB Port and 4.6 Configuration of USB and CAN Interface)

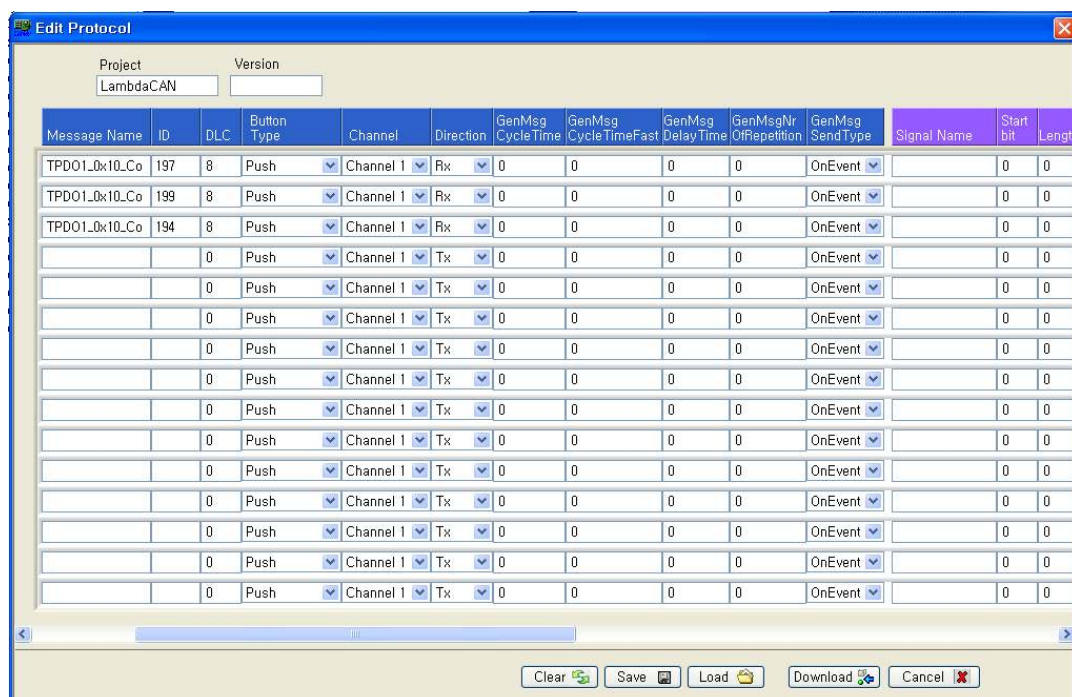


6) Open a CAN database (*.dbc) file. .

7) Drag & Drop the message or signal to a slot of I-Generator Screen.



8) Create a configuration data and edit attributes (Refer to 4.7 Edit Configuration Data)



9) Start download

The download is successful, if the following message is displayed.



Quit [OK] and exit **CANdyEditor**.

10) Return to **CANdy**

CANdy is now ready to use.

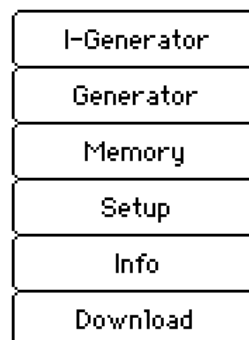
3 Use Instruction of CANdy

This chapter describes the usage of **CANdy** and in the following chapter – chapter 4 - the usage of the editor program **CANdyEditor**.

If the power is switched on, **CANdy** loads automatically the configuration data from the internal memory. Thereafter the main menu appears on the LCD,

3.1 Main menu

CANdy has 6 main menus as follow:








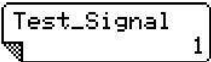









3.1.1 I-Generator (Interactive Generator)

This Menu handles with

- Reception of CAN message,
- Transfer of CAN message,
- Display of signal value,
- Graphical display of a signal value and
- Configuration of the message attributes.



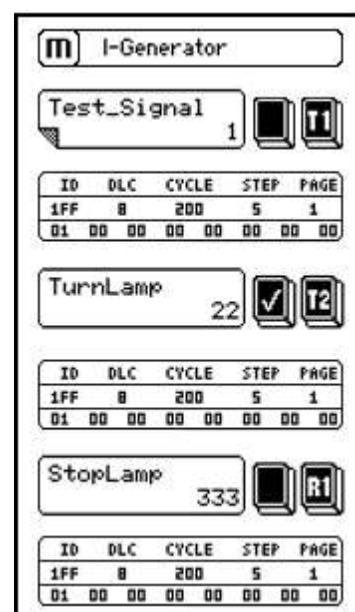
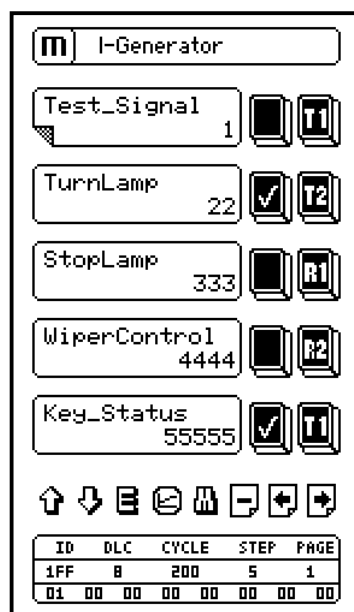
Description of Symbols

No.	Symbol	Description
1		Move to main menu.
2		Bus normal status
3		Bus Off status Push the symbol clears the Bus Off symbol.  turns to  , but the Bus Off status on the CAN bus is not influenced.
4		The Signal name according to the configuration data and the actual signal value are displayed. The folded left edge means, that the signal is selected for manipulation.
5		The symbol means the message is set for periodic transmission. Press the symbol stops to transmit message.
6		The periodic message is not activated to transmit. Press the symbol starts to transmit the message periodically.
7		CAN channel 1 is configured as a transmit port. Press the symbol starts to transmit CAN message, if it is defined as an OnEvent message.
8		CAN channel 2 is configured as a transmit port. Press the symbol starts to transmit CAN message, if it is defined as an OnEvent message.
9		CAN channel 1 is set as a receiving port.
10		CAN channel 2 is set as a receiving port.
11		Select the button increases the send value by the predefined step.
12		Select the button decreases the send value by the predefined step.
13		Enter into the edit mode (s. 3.3.3 Attributes of Message and Signal)

14		Enter into the graphic mode. The received signal is displayed on LCD graphically.
15		Change the display mode between “5+1” (5 signal+1 info-box) display mode ¹⁾ and “3+3” (3 signal+3 info-box) display mode ²⁾ .
16		Delete the message or signal value.
17		Move to the previous page.
18		Move to the next page
19		Info box displays ID, DLC, cyclic time, increment step, page no. and 8 bytes CAN data.

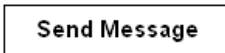
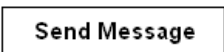
¹⁾ “5+1” display mode

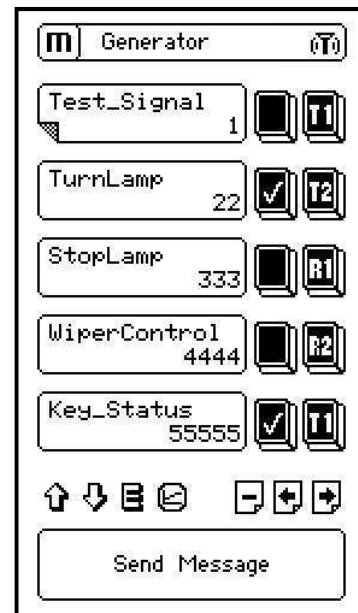
²⁾ “3+3” display mode








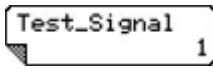



3.1.2 Generator












Generator has the same function as the **I-Generator**, but the display mode “3+3” (Refer to **3.1.1 I-Generator(Interactive Generator)**) is not supported.

Every time select the  button the message, marked with a folded left edge, is transferred to the CAN bus. Afterwards the next message is sequentially selected as a successor to be sent at next selection of .



Description of the Symbols:

No.	Symbol	Description
1		Move to main menu.
2		Bus normal status
3		Bus Off status Press the symbol clears the bus off symbol on display  →  , but the Bus off status on the CAN bus is not influenced.
4		The Signal name according the configuration and the actual signal value are displayed. The folded left edge means, that the signal is selected for manipulation.
5		The symbol means that the message is sent periodically. Press the symbol stops to send message.
6		The periodic message is not activated to transmit. Press the symbol starts to send the message periodically.
7		CAN channel 1 is set as a transmit port. Press the symbol starts to transmit CAN message, if it is configured as an OnEvent message.

8		CAN channel 2 is set as a transmit port. Press the symbol starts to transmit CAN message, if it is configured as an OnEvent message.
9		CAN channel 1 is set as a receive port.
10		CAN channel 2 is set as a receive port.
11		Increases the send value by a predefined step.
12		Decrease the send value by a predefined step.
13		Enter into the edit mode (s. s. 3.3.3 Attributes of Message and Signal)
14		Enter into the graphic mode – received signal is displayed graphically.
15		Delete message and signal value of the selected signal.
16		Move to the previous page.
17		Move to the next page
18		By each selection of the [Send Message] button, the selected message, marked with a folded left edge, is sent.

Remarks:

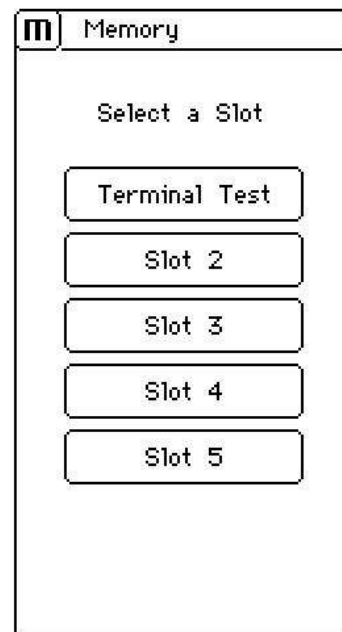
- 1) Define and use different signal values with a same CAN message, e.g. 0(Off) and 1(On) , if an alternately changing values are required.

3.1.3 Memory

CANdy has 5 memory slots for message and signal configuration.

In each memory slot up to 15 message and signal configurations can be stored.

- For every memory slot a own name can be given.
- By selecting the memory slot sub-menus [SAVE] [LOAD] [DELETE] appears on display.
- With [SAVE] the selected configuration is saved and is ready to use.
- The configuration data is loaded automatically at power on of **CANdy**.

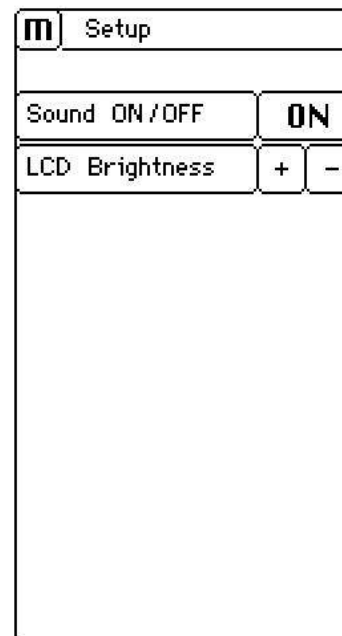


The screenshot shows the 'Memory' menu with a title bar containing an 'm' icon and the word 'Memory'. Below the title bar, the text 'Select a Slot' is displayed. There are five buttons stacked vertically: 'Terminal Test', 'Slot 2', 'Slot 3', 'Slot 4', and 'Slot 5'.

3.1.4 Setup

In the setup menu the buzzer can be set to on and off.

The brightness of the display can be also adjusted. The new configurations are stored automatically.

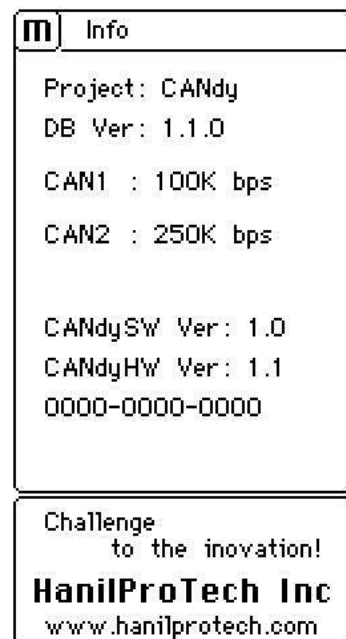


The screenshot shows the 'Setup' menu with a title bar containing an 'm' icon and the word 'Setup'. Below the title bar, there are two rows of controls. The first row has a label 'Sound ON/OFF' and a button labeled 'ON'. The second row has a label 'LCD Brightness' and two buttons labeled '+' and '-'. Below these controls is a large empty rectangular area.

3.1.5 Info

The configuration information of the selected project are displayed:

- project name
- version of DB
- communication speed of CAN interface
- software version
- serial number



3.1.6 Download

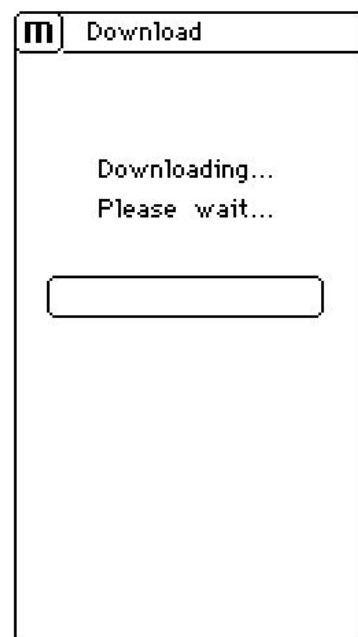
The **Download** menu is used to download configuration data from [CANdyEditor](#).

Remarks:

The connection to PC must be done before

selecting the menu

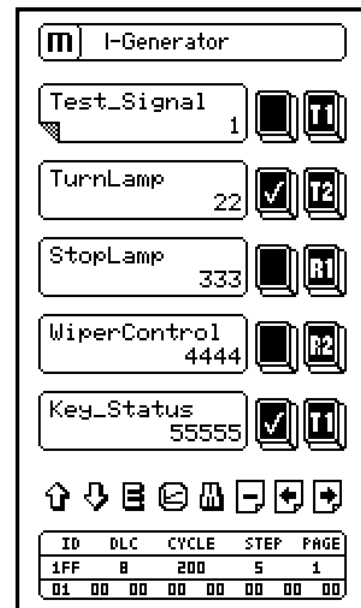
Download



3.2 Message Transfer

CAN message can be transferred using menus either





- **I-Generator**
- or
- **Generator.**



3.2.1 Periodic Message

Attributes:

Message	SendType	Periodic
Message	CycleTime	> 10 ms

- Selecting the  button, the symbol turns to  and **CANdy** starts to send cyclically the CAN message.
- Selecting the  button again, the message transfer stops. The symbol turns to .

Remarks:



- 1) At sending 15 periodic messages in whole, the cyclic time must not below 20 ms, otherwise the cyclic time can't be kept exactly.

- 2) If periodic time of 10ms must be kept, the number of CAN messages must not exceed 6 periodic messages.

3.2.2 OnEvent Message

Attributes:







Message	SendType	OnEvent
---------	----------	---------

- Pushing the  or  button, the selected message is sent just once.

3.2.3 Periodic & OnEvent Message







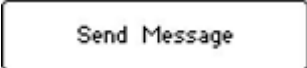
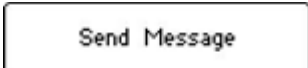
Attributes:

Message	SendType	Periodic
Signal	SendType	OnEvent

- Pushing the  button, the symbol turns to  and the selected message is started to send periodically,
- Pushing the  button again, the message transfer is stopped. The symbol turns to .
- Selecting the  or  button, the selected message is sent once.

3.2.4 Message Transfer according a Sequence List

For sending same CAN message with different signal value according a sequence list, **Generator** can be used:



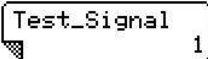


- Pushing the  button the symbol turns to  and the selected message is started to send periodically,
- Pushing the  button again the message transfer is stopped. The symbol turns to .
- Selecting the  or  button the selected message is sent once.
- At each selection of the  button, a message, which is defined as transfer message, is sent out, and a next transfer message is selected to be transferred by next push of the  button. The messages, which are defined as a receive messages, are skipped and the next transfer message is positioned to ready to send.



3.3 Message Reception

Attributes:

Message	Direction	Rx
Signal	StartBit	The signal information can be defined.
Signal	Length	The signal information can be defined.

- The signal value can be monitored based on the signal configuration.
- For change the direction to reception, turn the button to  or .
-  shows, that the selected slot has received signal value 1 for the signal "Test_Signal"
- In the receiving mode  or  no periodic transmission can be activated.

3.3.1 Graph Window

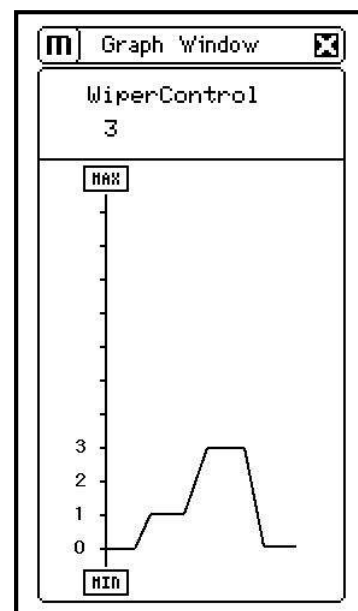
The graph window shows a trend of the selected signal.

The graph is updated at every reception of a new value.

At the upper part of the graph window, signal name and signal value are displayed.

The scale of the y-axis is calculated from the maximum and minimum value of the configuration data. In case of incorrect declaration the scale is calculated automatically.

The maximum and minimum value can be modified by selecting [MAX] or [MIN].



3.3.2 Remarks to Message Transmission

Signals with

- Signal length ≤ 16 bit value of Intel or Motorola format can be transmitted.

Signals of

- 1) Signed Type
- 2) Float Type

can't be transmitted. If these signals are defined wrongly as transmission, the transmission mode changes automatically to receive mode.

Signals with

- Signal length ≤ 16 bit signed type
- Signal length ≤ 16 bit unsigned type
- Signal length = 32 bit Intel float type

can be received.

All signals with

- signal length > 16 bit, except 32 bit Intel, float type

can't be received.

The baudrate of the CAN communication is set to 100 kbps as a default value and must be modified by **CANdyEditor**, if another baudrate is required.

It is recommended to define signal groups with different baudrate, if baudrate often changes. Every memory slot – 5 in all - can be configured with different baudrate.

If a message with a specified data should be transmitted, the signal length of message must be set to 0. Otherwise - if signal length is not 0 – the signal value influences the message data.

Example:

The message has signal attributes:

signal length = 1,

Startbit = 0,

Sendvalue = 0,

and a 4-bytes data **FF FF FF FF** should be transmitted.

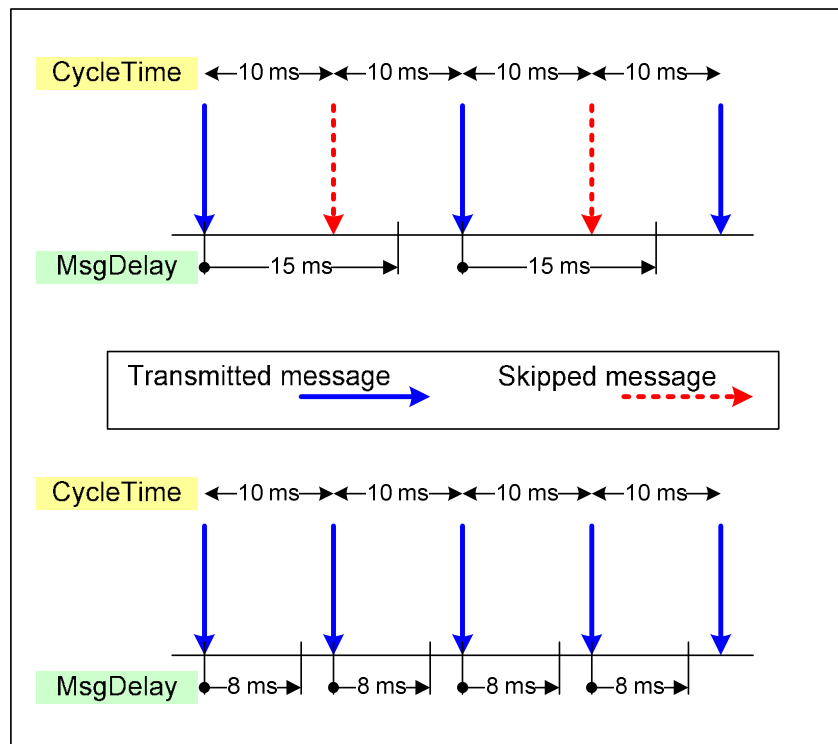
If the signal length isn't set to 0, then **FD FF FF FF** is sent instead of **FF FF FF FF** as initially specified. Therefore the signal length must set to 0.

It is recommended to delete the configuration data and initialize new.

The elapsed time **MsgDelay** between 2 message transmissions with same ID must be smaller than **CycleTime** and **RepeatTime**. Otherwise the following message transmission within **MsgDelay** is ignored.

Example:

A periodic message with CycleTime = 10 ms, MsgDelay = 15 ms is transmitted in every 20 ms. The second message between first and third is ignored, because no message transmission takes place within the MsgDelay (15 ms). In case of CycleTime > MsgDelay, every cyclic message is sent without loss.



3.3.3 Attributes of Message and Signal

1) Attributes of Message

No.	Parameter	Description
1	Message Name	Message name defined in CANdyEditor
2	ID	CAN Identifier: 0x001 ~ 0x7FF
3	DLC	Data Length Code: 1 ~ 8
4	Data	1 ~ 8 byte CAN message data
5	SendType	Periodic : Cyclic transmission OnEvent : Transmission driven by Event
6	CycleTime	Periodic message ($\geq 10\text{ms}$)
7	MsgDelay	Minimal delay time between 2 message transmissions with same ID. Remarks: If a message is defined as a Periodic & OnEvent message and message transmission is activated by OnEvent , then the transmission of the message is delayed, until MsgDelay elapses.
8	Repetition	Number of repeated message transmission including first message transmission.
9	RepeatTime	Delay time between 2 repeated message transmissions.. Remarks: RepeatTime \geq MsgDelay
10	ButtonType	Transmission condition of OnEvent message Push : Transmission at push of the button Push & Release : Transmission at push and release of the button. When the button is released, signal value is set to 0. This is useful to switch on and off.
11	Channel	CAN channel – CAN1, CAN2
12	Direction	Transmission Tx , Reception Rx

Edit

Message Name : HPT_01

ID: 0x1FF

DLC: 8

DATA: 01 00 00 00 00 00 00

SendType: Periodic

CycleTime: 200

MsgDelay: 20

Repetition: 3

RepeatTime: 40

ButtonType: Push Rel

Channel: CAN1

Direction: Tx

2) Attributes of Signal

No.	Parameter	Description
1	Name	Signal name defined in CANdyEditor
2	StartBit	Start bit (0 ~ 63) of the signal.
3	Length	Bit length of the signal (1~16) Exceeding of bits causes automatically to Float Rx
4	SendType	Signal and message send type (Refer to 3.2 Message Transfer)
5	Factor	Factor value to be multiplied, if Physical is selected as a display type.
6	Offset	Offset value to be added, if Physical is selected as a display type.
7	ValueType	Signal type <ul style="list-style-type: none"> - Signed (only for Rx) - Unsigned (for Tx and Rx) - Float (for Rx of Intel format)
8	SendValue	Signal value to be sent.
9	ResetValue	Initial value of the send signal in case of Periodic & OnEvent ,
10	Step	Increment value
11	ByteOrder	Byte order of the signal <ul style="list-style-type: none"> • INTEL • MOTOROLA
12	DisplayType	Display type of received data: <ul style="list-style-type: none"> • Raw • Physical

Edit

Signal Name : TestSignal

StartBit: 0

Length: 2

SendType: ONEVENT

Factor: 1.0

Offset: 0.0

ValueType: UNSIGNED

SendValue: 1


ResetValue: RESET

Step: 1

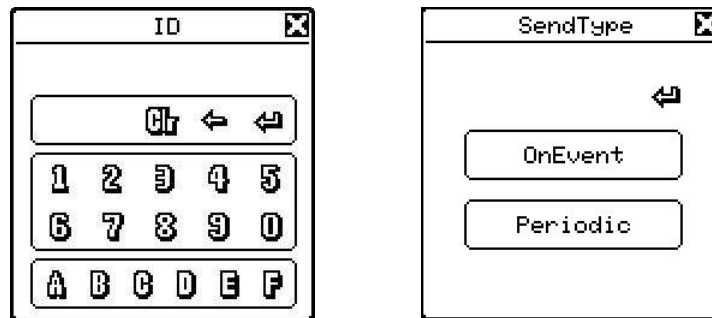
ByteOrder: INTEL


DisplayType: RAW


3.3.4 Change of Attributes

Select  in **I-Generator** or **Generator** to change attributes of message and signal.

Example: Modification of the **ID** and **Send type**



After entering ID using numeric keys and selecting send type, the return button  is activated. Otherwise the change is not saved.

With the exit button  it returns to the previous menu.


3.4 Examples

3.4.1 OnEvent

Message	SendType	OnEvent
Message	Repetition	1
Signal	SendType	None or On Event

Time	Chn	ID	Name	Dir	DLC	Data
17.515890	1	7ff		Rx	8	00 00 00 00 00 00 00 00
5.974280	1	7ff		Rx	8	00 00 00 00 00 00 00 00
1.982730	1	7ff		Rx	8	00 00 00 00 00 00 00 00

Tx Button Pushed


- Every time the transmit button  is pushed, the **OnEvent** message is sent.

3.4.2 OnEvent & Repetition

Message	SendType	OnEvent
Message	Repetition	3 (1 & 2 Repetition)
Message	RepeatTime	40 ms
Signal	SendType	None or On Event

Time	Chn	ID	Name	Dir	DLC	Data
115.170260	1	7ff		Rx	8	00 00 00 00 00 00 00 00
0.039780	1	7ff		Rx	8	00 00 00 00 00 00 00 00
0.040080	1	7ff		Rx	8	00 00 00 00 00 00 00 00
4.190200	1	7ff		Rx	8	00 00 00 00 00 00 00 00
0.039780	1	7ff		Rx	8	00 00 00 00 00 00 00 00
0.040080	1	7ff		Rx	8	00 00 00 00 00 00 00 00
6.224000	1	7ff		Rx	8	00 00 00 00 00 00 00 00
0.039820	1	7ff		Rx	8	00 00 00 00 00 00 00 00
0.040070	1	7ff		Rx	8	00 00 00 00 00 00 00 00

Tx Button Pushed


- If the transmit button  is pushed, the **OnEvent** message is sent once and repeated twice with the **RepeatTime** (40 ms) repetition time interval.

3.4.3 Periodic

Message	SendType	Periodic
Message	CycleTime	100 ms
Signal	SendType	None

Time	Chn	ID	Name	Dir	DLC	Data
201.427680	1	7ff		Rx	8	00 00 00 00 00 00 00 00
0.100180	1	7ff		Rx	8	00 00 00 00 00 00 00 00
0.100190	1	7ff		Rx	8	00 00 00 00 00 00 00 00
0.100190	1	7ff		Rx	8	00 00 00 00 00 00 00 00
0.100190	1	7ff		Rx	8	00 00 00 00 00 00 00 00
0.100180	1	7ff		Rx	8	00 00 00 00 00 00 00 00
0.100190	1	7ff		Rx	8	00 00 00 00 00 00 00 00
0.100190	1	7ff		Rx	8	00 00 00 00 00 00 00 00
0.100190	1	7ff		Rx	8	00 00 00 00 00 00 00 00
0.100190	1	7ff		Rx	8	00 00 00 00 00 00 00 00
0.100190	1	7ff		Rx	8	00 00 00 00 00 00 00 00

Periodic
Button
Pushed

- If the cyclic transmit button is set to , a **Periodic** message is sent cyclically in every **CycleTime** (100 ms).

3.4.4 Periodic & OnEvent



Message	SendType	Periodic
Message	CycleTime	100 ms
Message	MsgDelay	20 ms
Message	Repetition	1
Signal	SendType	OnEvent

Time	Chn	ID	Name	Dir	DLC	Data
0.100180	1	7ff		Rx	8	00 00 00 00 00 00 00 00
0.100190	1	7ff		Rx	8	00 00 00 00 00 00 00 00
0.100200	1	7ff		Rx	8	00 00 00 00 00 00 00 00
0.100180	1	7ff		Rx	8	00 00 00 00 00 00 00 00
0.090430	1	7ff		Rx	8	00 00 00 00 00 00 00 00
0.109950	1	7ff		Rx	8	00 00 00 00 00 00 00 00
0.100200	1	7ff		Rx	8	00 00 00 00 00 00 00 00
0.100180	1	7ff		Rx	8	00 00 00 00 00 00 00 00
0.100190	1	7ff		Rx	8	00 00 00 00 00 00 00 00
0.100190	1	7ff		Rx	8	00 00 00 00 00 00 00 00
0.100180	1	7ff		Rx	8	00 00 00 00 00 00 00 00
0.100190	1	7ff		Rx	8	00 00 00 00 00 00 00 00
0.040410	1	7ff		Rx	8	00 00 00 00 00 00 00 00
0.059780	1	7ff		Rx	8	00 00 00 00 00 00 00 00

Periodic
Button
Pushed























Tx Button
Pushed



Tx Button
Pushed

- If the cyclic transmit button is set to , a **Periodic** message is sent cyclically with the **CycleTime** (100 ms).
- At pressing the transmit button  the **OnEvent** message is sent with **MessageDelay** (20ms).

3.4.5 Periodic & OnEvent with Repetition

Message	SendType	Periodic
Message	CycleTime	100 ms
Message	MsgDelay	20 ms
Message	Repetition	3 (1 + 2 repetition)
Message	RepeatTime	40 ms
Signal	SendType	OnEvent


Time	Chn	ID	Name	Dir	DLC	Data	
 0.100180	1	7ff		Rx	8	00 00 00 00 00 00 00 00	 <div>Periodic Button Pushed</div>
 0.100200	1	7ff		Rx	8	00 00 00 00 00 00 00 00	
 0.100190	1	7ff		Rx	8	00 00 00 00 00 00 00 00	
 0.100190	1	7ff		Rx	8	00 00 00 00 00 00 00 00	
 0.020240	1	7ff		Rx	8	00 00 00 00 00 00 00 00	 <div>Tx Button Pushed</div>
 0.039930	1	7ff		Rx	8	00 00 00 00 00 00 00 00	
 0.040090	1	7ff		Rx	8	00 00 00 00 00 00 00 00	
 0.100110	1	7ff		Rx	8	00 00 00 00 00 00 00 00	 <div>Tx Button Pushed</div>
 0.100190	1	7ff		Rx	8	00 00 00 00 00 00 00 00	
 0.100180	1	7ff		Rx	8	00 00 00 00 00 00 00 00	
 0.100200	1	7ff		Rx	8	00 00 00 00 00 00 00 00	
 0.100190	1	7ff		Rx	8	00 00 00 00 00 00 00 00	
 0.052470	1	7ff		Rx	8	00 00 00 00 00 00 00 00	
 0.039770	1	7ff		Rx	8	00 00 00 00 00 00 00 00	
 0.040070	1	7ff		Rx	8	00 00 00 00 00 00 00 00	
 0.068060	1	7ff		Rx	8	00 00 00 00 00 00 00 00	
 0.100200	1	7ff		Rx	8	00 00 00 00 00 00 00 00	
 0.100190	1	7ff		Rx	8	00 00 00 00 00 00 00 00	
 0.100180	1	7ff		Rx	8	00 00 00 00 00 00 00 00	

- If the periodic transmit button is set to , a **Periodic** message is send periodically with **CycleTime** (100 ms).
- At pressing the transmit button  an **OnEvent** signals is sent once and then 2-times repeated with **RepeatTime** (40 ms).

3.4.6 Button Type

Push Button



Message	SendType	OnEvent
Message	ButtonType	Push
Message	Repetition	1
Signal	StartBit	0
Signal	Length	1
Signal	SendValue	1

- An **OnEvent** message is sent by press the transmit button .
- At release of transmit button there is no message transmission.

Time	Chn	ID	Name	Dir	DLC	Data	
3422.113010	1	7ff		Rx	8	01 00 00 00 00 00 00 00	Tx Button Pushed
11.131960	1	7ff		Rx	8	01 00 00 00 00 00 00 00	

Push & Release Button

Message	SendType	OnEvent
Message	ButtonType	Push & Release
Message	Repetition	1
Signal	StartBit	0
Signal	Length	1
Signal	SendValue	1

- An **OnEvent** message is send **SendValue** at push the transmit button  and the reset value at release  button.

Time	Chn	ID	Name	Dir	DLC	Data	
3512.216900	1	7ff		Rx	8	01 00 00 00 00 00 00 00	Tx Button Pushed
0.270500	1	7ff		Rx	8	00 00 00 00 00 00 00 00	Tx Button Released

3.4.7 Signal Value Reset of Periodic & OnEvent Message

Signal Value Reset

Message	SendType	Periodic
Message	CycleTime	100 ms
Message	Repetition	3
Message	RepeatTime	40
Signal	SendType	OnEvent
Signal	StartBit	0
Signal	Length	1
Signal	SendValue	1
Signal	Value Reset	Reset

- A **Periodic** message combined with an **OnEvent** signal and activated **ValueReset** causes to send the signal value 0 periodically, if no **OnEvent** condition occurs.

With an occurrence of an **OnEvent** condition, in the upper example signal value 1 is sent 3-times with 40 ms repeat time.

- To use **Signal Value Reset** the attributes of signal must be configured.

Time	Chn	ID	Name	Dir	DLC	Data
0.100200	1	7ff		Rx	8	00 00 00 00 00 00 00 00
0.100190	1	7ff		Rx	8	00 00 00 00 00 00 00 00
0.100180	1	7ff		Rx	8	00 00 00 00 00 00 00 00
0.100190	1	7ff		Rx	8	00 00 00 00 00 00 00 00
0.100200	1	7ff		Rx	8	00 00 00 00 00 00 00 00
0.020240	1	7ff		Rx	8	01 00 00 00 00 00 00 00
0.039940	1	7ff		Rx	8	01 00 00 00 00 00 00 00
0.040090	1	7ff		Rx	8	01 00 00 00 00 00 00 00
0.100110	1	7ff		Rx	8	00 00 00 00 00 00 00 00
0.100180	1	7ff		Rx	8	00 00 00 00 00 00 00 00
0.100190	1	7ff		Rx	8	00 00 00 00 00 00 00 00

Tx Button Pushed

Signal Value No Reset

Message	SendType	Periodic
Message	CycleTime	100 ms
Message	Repetition	3
Message	RepeatTime	40
Signal	SendType	OnEvent
Signal	StartBit	0

Signal	Length	1
Signal	SendValue	1
Signal	Value Reset	No Reset

- A **Periodic** message combined with an **OnEvent** signal and non-activated **SendValue** causes to send the signal value 1 periodically, if no **OnEvent** condition occurs.

With an occurrence of an **OnEvent** condition, in the upper example the signal value 1 is sent 3-times with 40 ms repeat time.

Time	Chn	ID	Name	Dir	DLC	Data
0.100190	1	7ff		Rx	8	01 00 00 00 00 00 00 00
0.100190	1	7ff		Rx	8	01 00 00 00 00 00 00 00
0.100180	1	7ff		Rx	8	01 00 00 00 00 00 00 00
0.100190	1	7ff		Rx	8	01 00 00 00 00 00 00 00
0.100180	1	7ff		Rx	8	01 00 00 00 00 00 00 00
0.100200	1	7ff		Rx	8	01 00 00 00 00 00 00 00
0.025450	1	7ff		Rx	8	01 00 00 00 00 00 00 00
0.039730	1	7ff		Rx	8	01 00 00 00 00 00 00 00
0.040080	1	7ff		Rx	8	01 00 00 00 00 00 00 00
0.095120	1	7ff		Rx	8	01 00 00 00 00 00 00 00
0.100180	1	7ff		Rx	8	01 00 00 00 00 00 00 00
0.100200	1	7ff		Rx	8	01 00 00 00 00 00 00 00
0.100190	1	7ff		Rx	8	01 00 00 00 00 00 00 00

Tx Button Pushed

4 Use Instruction of **CANdyEditor**

CANdyEditor is a PC-based editor program, developed to create configuration data for use in **CANdy**.

Based on CAN database (*.dbc) the user can define messages and signals to be sent and received with **CANdy**.

Characteristics:

- 1) Comfortable configuration by GUI(Graphic User Interface)
 - 1) Tree view and window
 - 2) Selection of options by Drag & Drop
- 2) Messages and signals can be defined easily on base of CAN database (*.dbc).

Requirements of the PC:

1	OS	MS Windows XP
2	RAM	> 256 MB
3	Monitor	1024 x 768 Resolution
4	HDD	> 160 MB

4.1 Installation of **CANdyEditor**

The installation of **CANdyEditor** can be done with execution of **setup.exe**.

이름	크기	종류
bin		파일 폴더
license		파일 폴더
supportfiles		파일 폴더
nidist.id	1KB	ID 파일
setup.exe	2,352KB	응용 프로그램
setup.ini	11KB	구성 설정

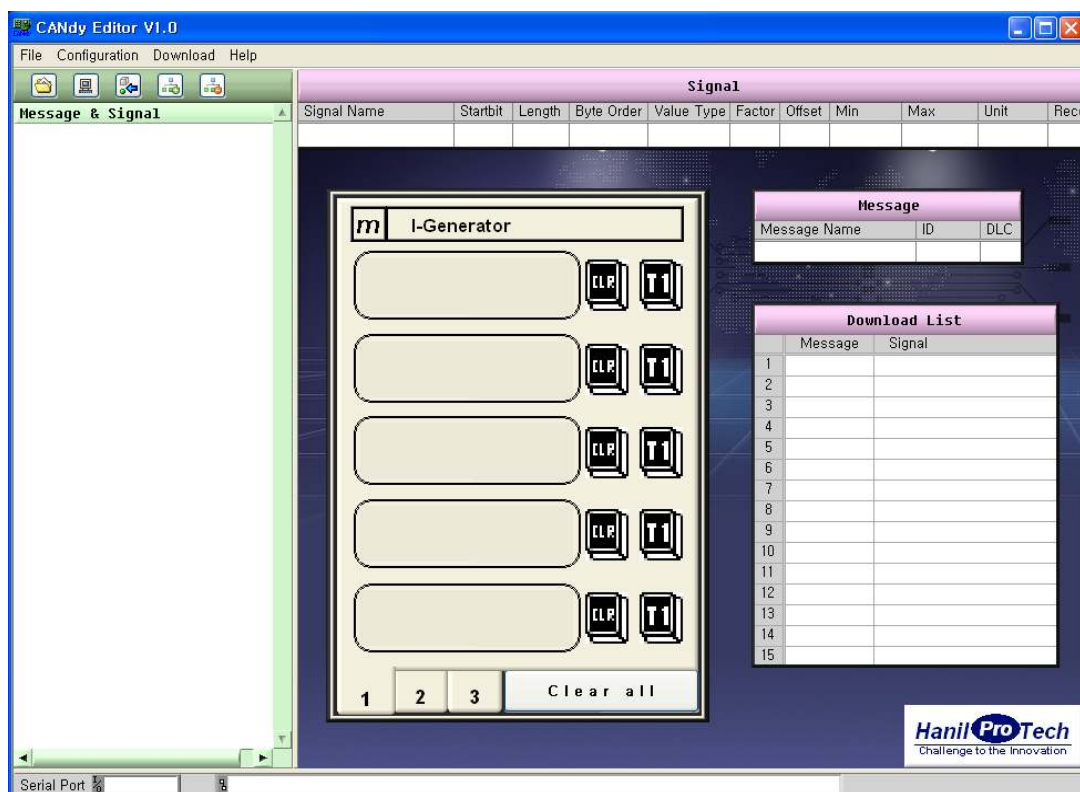
And follow the instruction of the program.



4.2 Run CANdyEditor

Execute **CandyEditor.exe**.

The following panel will appear on the PC monitor.








4.3 Main Menu

The main menu consists of 4 function groups:


Main Menu	Sub-menu	Description
[File]	[New]	Create a new configuration data
	[Open DB]	Open a CAN database file (*.dbc)
	[Quit]	Exit of program
[Configuration]	[Setup]	Configure USB and CAN interfaces
[Download]	[Download]	Download the configuration data to CANdy
[Help]	[Help]	Describes diverse help instructions
	[Information]	Information about CANdyEditor

Icons



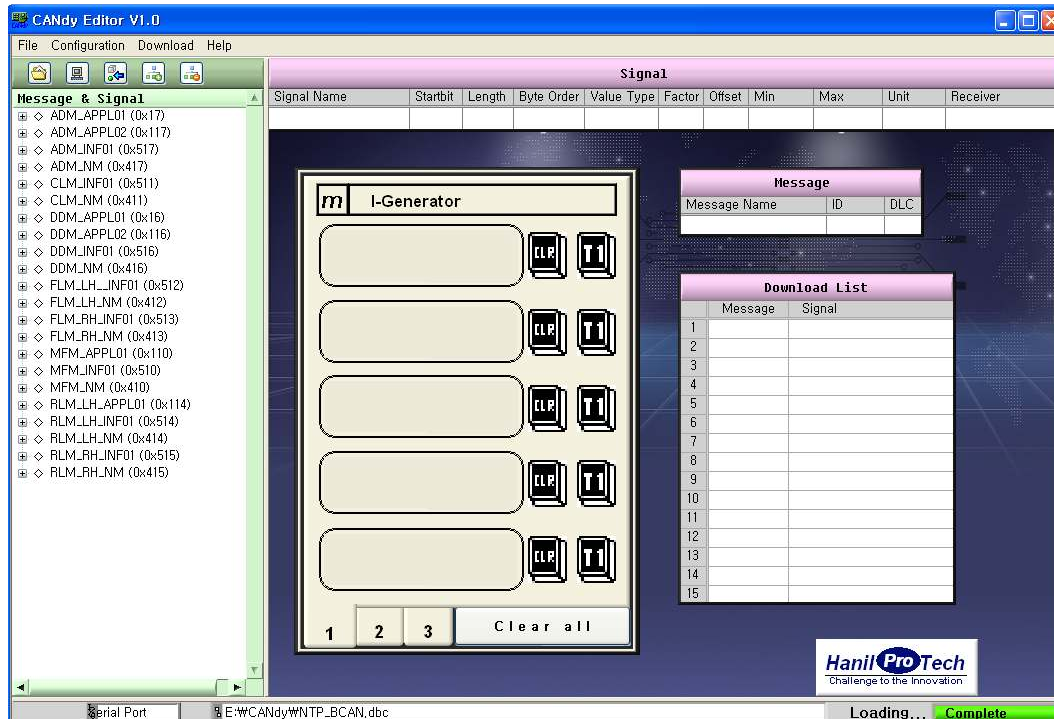
	Open a CAN database file (*.dbc)
	Configure USB and CAN interfaces
	Download the configuration data (message and signal definitions) to CANdy
	Open the tree on the left side of the window
	Close the tree

4.4 Load CAN Database

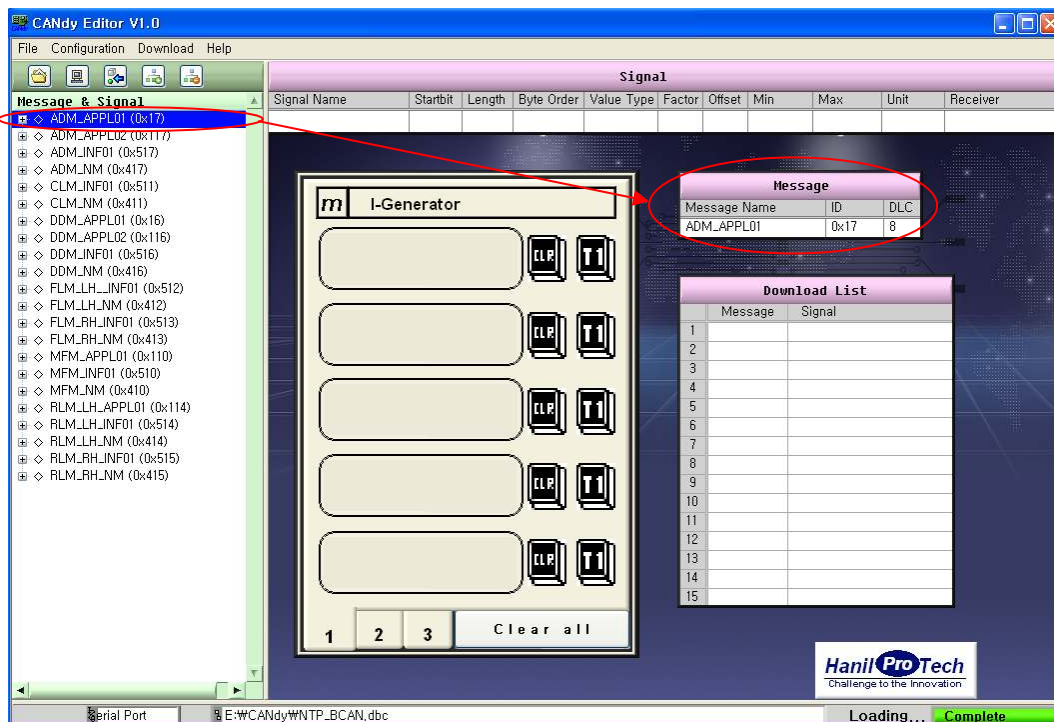
Open a CAN database (*.dbc) selecting the icon  or inserte in main menu **[File]** and **[Open DB]**.

File	
New	Ctrl+N
Open DB	Ctrl+O
Quit	Ctrl+Q

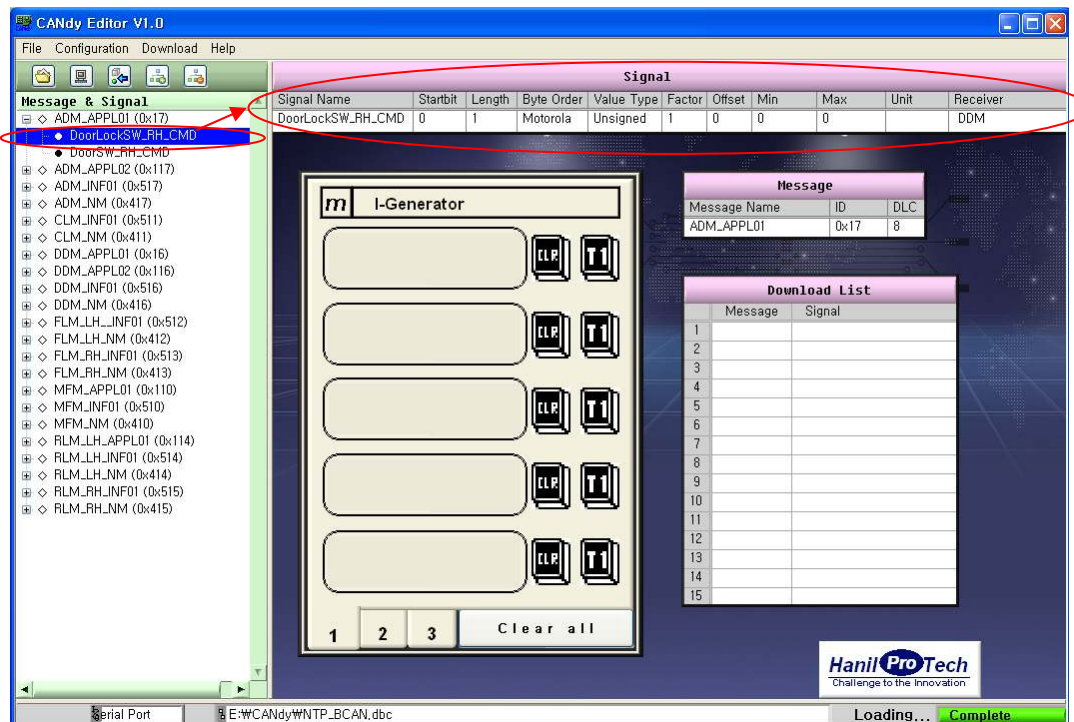
The imported CAN messages and signals are displayed in the Message & Signal window below the icons.



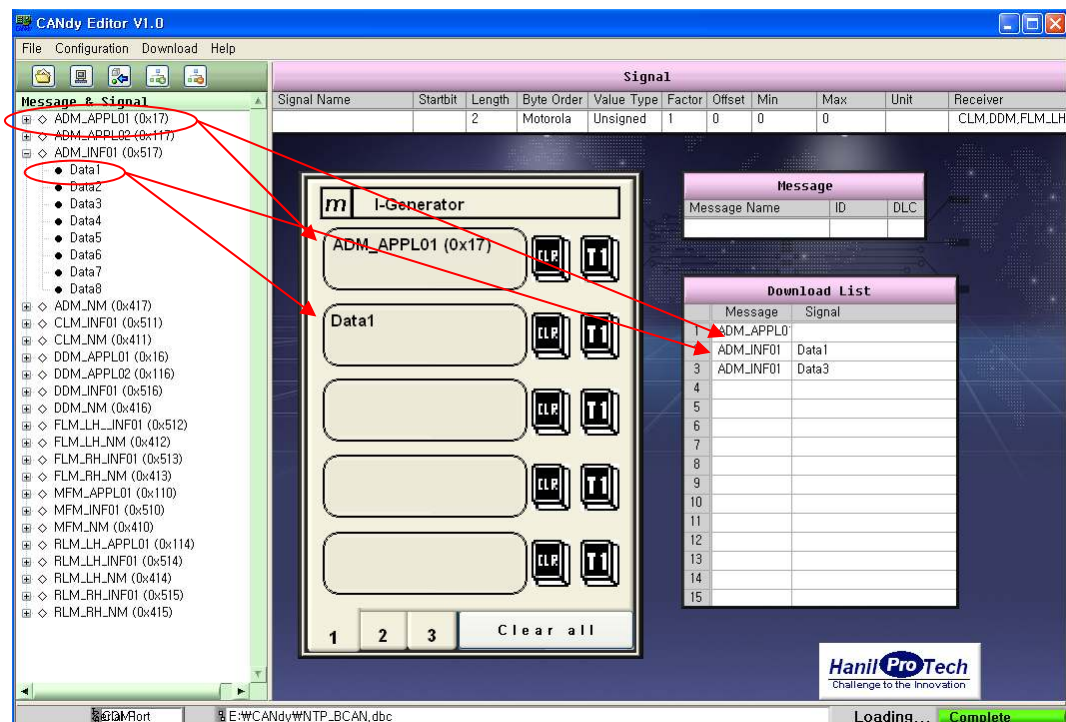
Select message or signal by mouse click. The message ID and length DLC are displayed separately in the message window.



The parameters of the selected signal are displayed also separately in signal window.



The selection of CAN messages and signals for **CANDy** can be done by “Drag & Drop” placing in slot of i-Generator.



Up to 5 messages and/or signals can be selected and placed in slots by “Drug & Drop”.

Repeating the procedure, up to 15 messages and/or signals can be defined.

With the [CLR] button at the right hand the configuration data can be deleted.

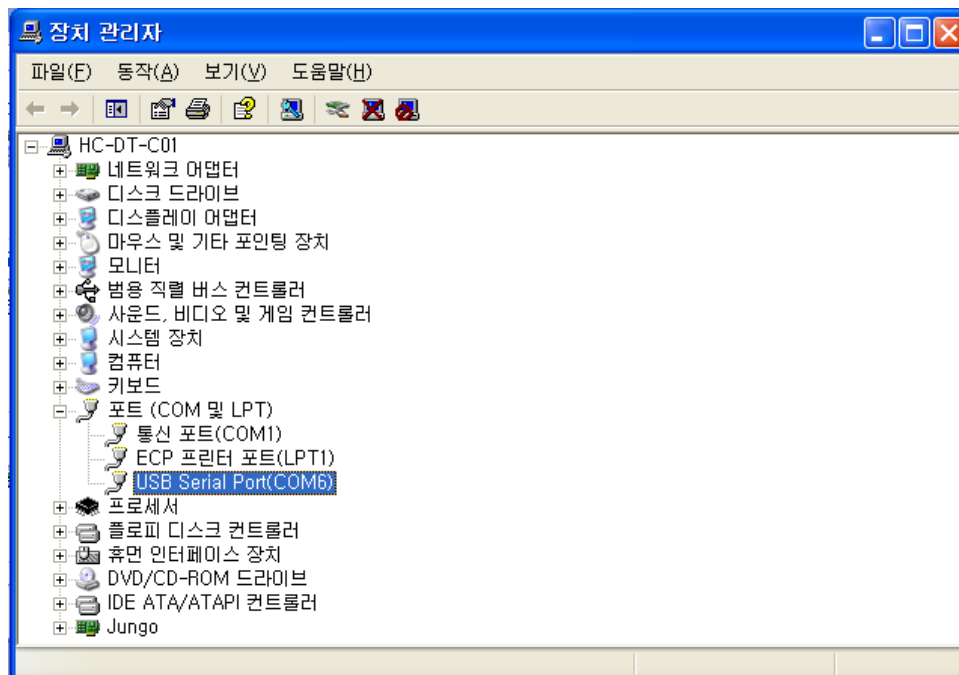
With the [Clear all] button all 15 configuration data in slots can be deleted.

Example.:

“T1” means that the corresponding signal/message is set on channel “#1” and for “transmission”.

4.5 Verification of USB Port

The USB port for communication with **CANdy** must be verified by the hardware status as shown below in the control panel of the PC..



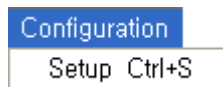
If PC doesn't recognize the USB interface of **CANdy**, which is connected to PC and powered on, automatically, a new USB driver must be installed.

To install a USB driver, use <**USBdriver**> folder in CD delivered with **CANdy**.

이름	크기	종류	수정된 날짜
CDM 2.00.00 Release Info...	42KB	Microsoft Office ...	2006-09-27 오후 ...
FTBUSUI.dll	104KB	응용 프로그램 확장	2006-05-24 오전 ...
ftcserco.dll	20KB	응용 프로그램 확장	2006-05-18 오전 ...
FTD2XX.dll	172KB	응용 프로그램 확장	2006-05-24 오전 ...
FTD2XX.H	20KB	H 파일	2005-12-06 오전 ...
FTD2XX.lib	18KB	LIB 파일	2006-05-24 오전 ...
ftdibus.cat	10KB	보안 카탈로그	2006-06-02 오후 ...
FTDIBUS.INF	3KB	설치 정보	2006-05-24 오전 ...
FTDIBUS.sys	47KB	시스템 파일	2006-05-18 오전 ...
ftdiport.cat	10KB	보안 카탈로그	2006-06-02 오후 ...
FTDIPOINT.INF	4KB	설치 정보	2006-05-24 오전 ...
ftdiun2k.ini	1KB	구성 설정	2003-04-10 오후 ...
FTDIUNIN.exe	184KB	응용 프로그램	2006-05-24 오전 ...
FTLang.dll	100KB	응용 프로그램 확장	2006-05-24 오전 ...
ftser2k.sys	60KB	시스템 파일	2006-05-18 오전 ...
ftserui2.dll	33KB	응용 프로그램 확장	2006-05-19 오전 ...

The USB driver can be also downloaded from www.ftdichip.com.

Verify the USB port in the main menu [**Configuration**] and [**Setup**].



The port no. in the setup options window must be same as the USB interface recognized by PC.

4.6 Configuration of USB and CAN Interface

The configuration of the USB and CAN interface must be done before downloading the configuration data to **CANdy**.

Use Icon  or insert main menu [**Configuration**] and [**Setup**] to setup the USB and CAN parameters.

USB parameters:

- Port Number: COM port configured at the installation.
- Baudrate: 38400 bps.

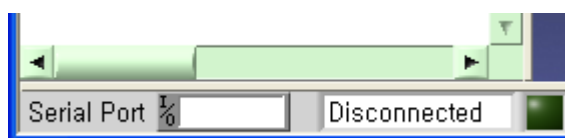
CAN1 and CAN2 parameters:

- Communication speed,
- Sampling Point,
- BTL
- BTR(Bus Timing Resister) 0, 1 are calculated.



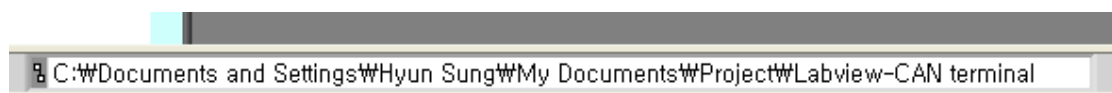
Status Bar

The status bar below of the window shows the condition of the serial communication port.



Remark: Before download configuration data, **CANdy** must be set to “download” mode.


Route of the database file

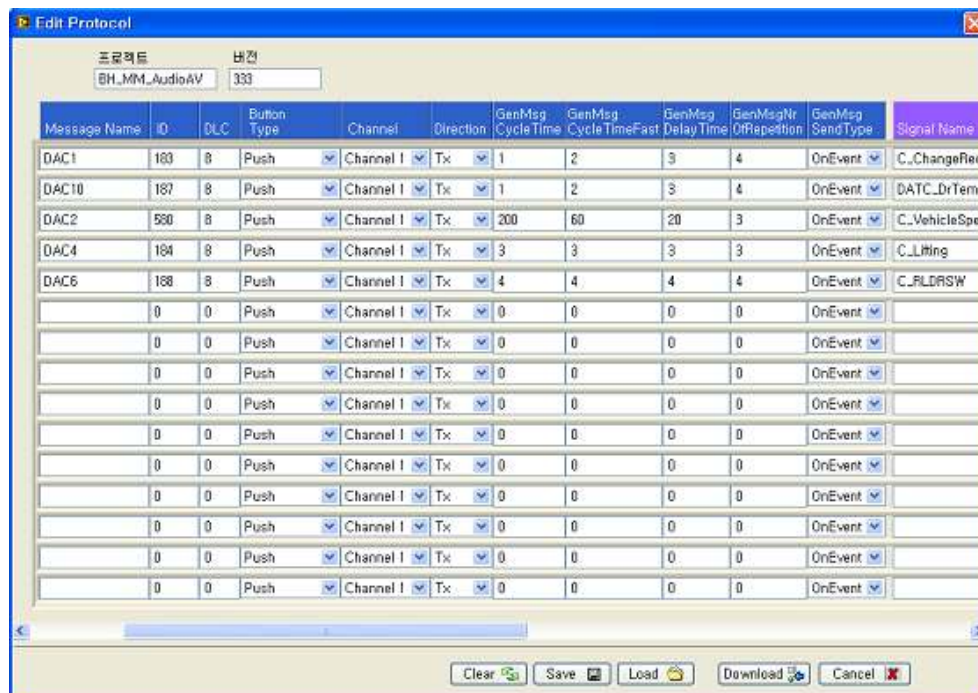


Loading bar



4.7 Edit Configuration Data

Open the editor window choosing the icon .



All contents and attributes can be changed manually. By a double click of the left mouse key the edited content is accepted.

Default setting:

1) Messages

	Object	Default	Selectable Attribute
M1	Button Type	Push	Push & Release
M2	Channel	1	2
M3	Direction	Tx	Rx
M4	Message Send Type	Periodic	OnEvent

2) Signals

	Object	Default	Selectable Attribute
S1	Display_phy	Raw data	Physical value
S2	ValueReset	Off	On
S3	Step	1	
S4	GenSigSendType	OnEvent	None

The modified configuration data can be saved using load button. The file format is thereby *.dbx and differs from the dbc file.

4.8 Download of Configuration Data

- 1) Connect the PC and **CANdy** by USB interface and check in menu window left below whether the status of the USB port is connected.

- 2) In setup option the serial port and baudrate are defined.

- 3) Configuration data (messages and signals) are downloaded to **CANdy**

choosing the icon  or insert in main menu **[Download]** and **[Download]**.



- 4) The configuration data is transferred and saved to use in **CANdy**.

5 Trouble Shooting

Q : **CANdy** doesn't transmit CAN message (LED for transmission doesn't light up).

A :

- Check the CAN cable to be connected proper to a CAN bus.
- Check the CAN parameter (CAN parameters(transmission rate, Sampling Point, Message transmission type, Cycle Time, etc.).
- Messages of periodic type - for example - can't be sent by send command of OnEvent.

Cyclic message can't be also sent, if the CycleTime is set to 0.

Q: The configuration data can't be downloaded to **CANdy**.

A:

- Check the USB cable.
- Check, whether the baudrate of serial communication is set to 38400 bps.
- Try again in the download process:
 - 1) Reset **CANdy**
 - 2) Set **CANdy** to download mode
 - 3) Check, whether the status bar of the USB port is on "Connected"
If not, bring the serial port to "connected" mode.
 - 4) From CANdy the download process can be executed.

6 Appendix

6.1 Specification

MCU: MB90F594G/ Fujitsu

Power Supply: 9-24 VDC

CAN Interface:

2 Channel

Transceiver (TJA1054A, PCA82C250/Philips)

Baudrate 100k bps (Default)

Serial Interface:

USB (Rs232 option)

LCD : 128 * 240 Touch Screen

6.2 Components to be delivered

CANdy

CANdyEditor

USB Cable

CAN cable (2set)

Power Supply 12VDC/1 A

User Manual