

## User Manual

Revision 1.005  
English

### **CAN / USB – Converter, for CAN Analysis use**

(Order Code: HD67390-U-D1)

### **CAN / Ethernet – Converter, for CAN Analysis use**

(Order Code: HD67390-E-D1)

for Website information:

[www.adfweb.com?Product=HD67390](http://www.adfweb.com?Product=HD67390)

for Price information:

**Mr Tuấn 0915 980 479 / 0357988660**

#### **Benefits and Main Features:**

- ▶ CAN Analyzer Advanced 2.0A, 2.0B (11 and 29 bit identifier);
- ▶ Free updating to lifetime;
- ▶ HW filter for CAN packet;
- ▶ MAX baud rate 1Mb;
- ▶ Industrial temperature range -40°C / 85°C (-40°F / 185°F)



HD67390-U-D1



HD67390-E-A1

For others similar products:

#### **CAN Analyzer**

See also the following link:

[www.adfweb.com?Product=HD67316](http://www.adfweb.com?Product=HD67316)

Do you have an your customer protocol?

See the following links:

[www.adfweb.com?Product=HD67003](http://www.adfweb.com?Product=HD67003)

Do you need to choose a device? do you want help?

Ask it to the following link:

[www.adfweb.com?Cmd=helpme](http://www.adfweb.com?Cmd=helpme)

**INDEX:**

	Page
INDEX	2
UPDATED DOCUMENTATION	2
REVISION LIST	2
WARNING	2
TRADEMARKS	2
SECURITY ALERT	3
CONNECTION SCHEME	4
INTRODUCTION	6
POWER SUPPLY	6
USB	7
ETHERNET	7
LEDS	7
UPDATE NEW FIRMWARE	9
CAN	12
PROTOCOL	13
MECHANICAL DIMENSIONS	25
ORDERING INFORMATIONS	26
ACCESSORIES	26
DISCLAIMER	27
OTHER REGULATIONS AND STANDARDS	27
WARRANTIES AND TECHNICAL SUPPORT	28
RETURN POLICY	28
PRODUCTS AND RELATED DOCUMENTS	28

**UPDATED DOCUMENTATION:**

Dear customer, we thank you for your attention and we remind you that you need to check that the following document is:

- Updated
- Related to the product you own

To obtain the most recently updated document, note the "document code" that appears at the top right-hand corner of each page of this document.

With this "Document Code" go to web page [www.adfweb.com/download/](http://www.adfweb.com/download/) and search for the corresponding code on the page. Click on the proper "Document Code" and download the updates.

To obtain the updated documentation for the product that you own, note the "Document Code" (Abbreviated DC on the product's box) and download the updated from our web site [www.adfweb.com/download/](http://www.adfweb.com/download/)

**REVISION LIST:**

Revision	Date	Author	Chapter	Description
1.002	05/12/2011	FI	All	Revision
1.003	10/04/2012	FI	All	Revision
1.004	07/02/2013	Nt	All	Added new chapters
1.005	30/05/2013	FI	All	Added USB driver link

**WARNING:**

ADFweb.com reserves the right to change information in this manual about our product without warning.  
ADFweb.com is not responsible for any error this manual may contain.

**TRADEMARKS:**

All trademarks mentioned in this document belong to their respective owners.

**SECURITY ALERT:****GENERAL INFORMATION**

To ensure safe operation, the device must be operated according to the instructions in the manual. When using the device are required for each individual application, legal and safety regulation. The same applies also when using accessories.

**INTENDED USE**

Machines and systems must be designed so the faulty conditions do not lead to a dangerous situation for the operator (i.e. independent limit switches, mechanical interlocks, etc.).

**QUALIFIED PERSONNEL**

The device can be used only by qualified personnel, strictly in accordance with the specifications.

Qualified personnel are persons who are familiar with the installation, assembly, commissioning and operation of this equipment and who have appropriate qualifications for their job.

**RESIDUAL RISKS**

The device is state of the art and is safe. The instrument can represent a potential hazard if they are inappropriately installed and operated by personnel untrained. These instructions refer to residual risks with the following symbol:



This symbol indicates that non-observance of the safety instructions is danger for people to serious injury or death and / or the possibility of damage.

**CE CONFORMITY**

The declaration is made by us. You can send an email to [support@adfweb.com](mailto:support@adfweb.com) or give us a call if you need it.

**CONNECTION SCHEME:**

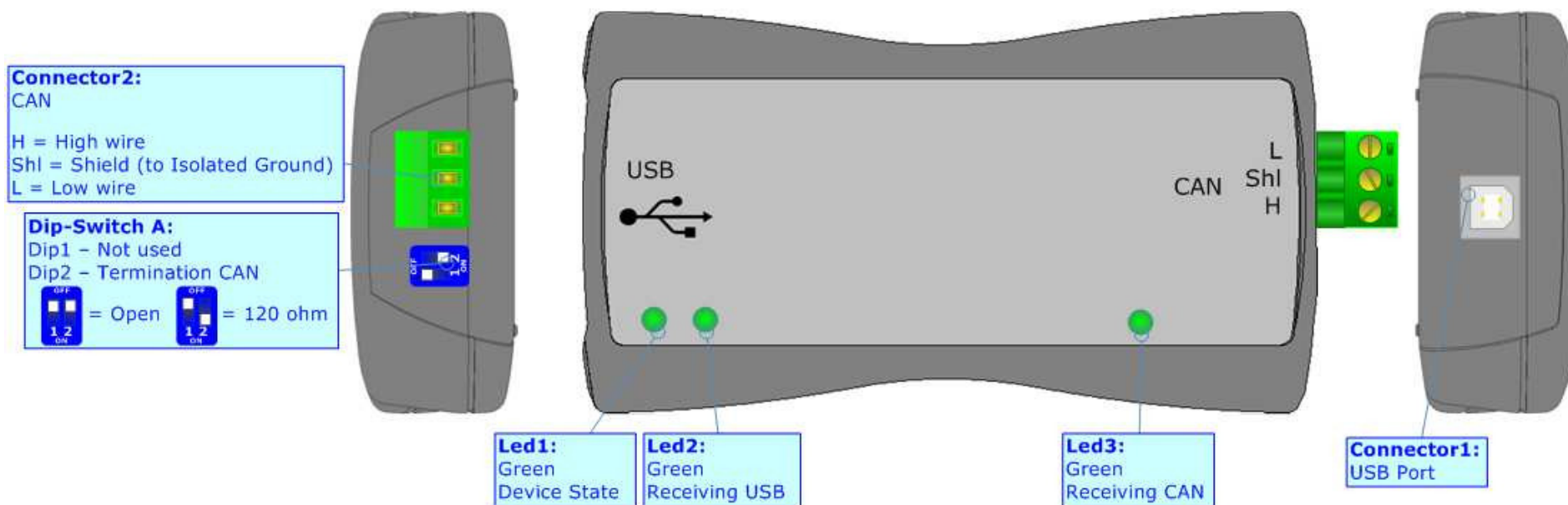


Figure 1: Connection scheme for HD67390-U-D1

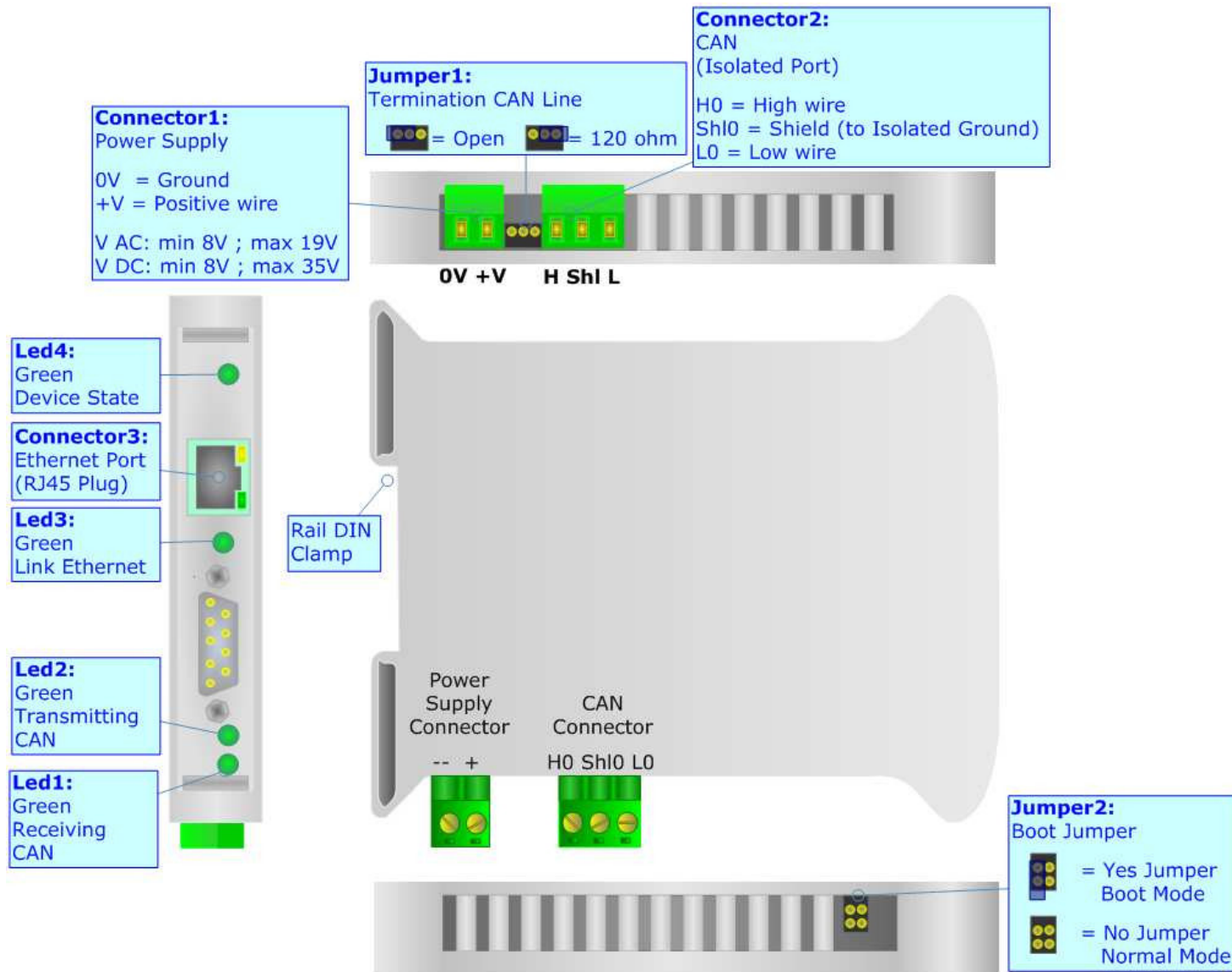


Figure 2: Connection scheme for HD67390-E-A1

**INTRODUCTION:**

The CANbus from/to USB/Ethernet is a powerful, flexible and economic instrument for communicate through USB/Ethernet with systems based in CAN/CANopen/J1939/DeviceNet/NMEA2000.


The instrument is composed of the following: module hardware with a USB/Ethernet interface that connects to a personal computer and a CAN terminal that connects to the line.

The software must be created following the information in this document. For the USB version the Port used is a COM port; for the Ethernet version you have to use the UDP protocol.

**POWER SUPPLY:**

The HD67390-U-D1 device don't need an external power supply for work. The alimentation is given by USB port.

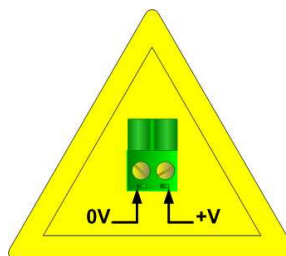
The HD67390-E-A1 can be powered at 8...19V AC and 8...35V DC. For more details see the two tables below.

VAC 		VDC 	
Vmin	Vmax	Vmin	Vmax
8V	19V	8V	35V

Consumption at 24V DC:

Device	W/VA
HD67390-E-A1	3.5

**Caution: Not reverse the polarity power**



HD67390-E-A1



**USB:**

The USB connector (Connector1) of HD67390-U-D1 is a Type-B Female. So the cable must be a Type-B Male.

The driver of USB can be downloaded from here: [www.adfweb.com/download/filefold/SW67119.zip](http://www.adfweb.com/download/filefold/SW67119.zip).

**ETHERNET:**

The Ethernet connection must be made using Connector3 of HD67390-E-A1 with at least a Category 5E cable. The maximum length of the cable should not exceed 100m. The cable has to conform to the T568 norms relative to connections in cat.5 up to 100 Mbps. To connect the device to an Hub/Switch is recommended the use of a straight cable, to connect the device to a PC/PLC/other is recommended the use of a cross cable.

**LEDS:**

The device HD67390-U-D1 has got three LEDs that are used to give information of the functioning status. The various meanings of the LEDs are described in the table below.

LED	Normal Mode	Boot Mode
1: Device State (Green)	Blink slowly (~2Hz)	Blink quickly
2: Receiving USB (Blue)	<b>Blink slowly:</b> No frame is received <b>Blink quickly:</b> Frame received on USB	Blink quickly
3: Receiving CAN (Blue)	<b>Blink slowly:</b> No frame is received <b>Blink quickly:</b> Frame received on CAN	Blink quickly

The device HD67390-E-A1 has got four LEDs that are used to give information of the functioning status. The various meanings of the LEDs are described in the table below.

LED	Normal Mode	Boot Mode
1: Receiving CAN (Green)	<b>Blink slowly (~2Hz):</b> No frame is received <b>Blink quickly:</b> A frame is received	Blink quickly
2: Transmitting CAN (Green)	Blink slowly (~2Hz)	Blink quickly
3: Link Ethernet (Green)	<b>ON:</b> Ethernet Cable connected <b>OFF:</b> Ethernet Cable disconnected	<b>ON:</b> Ethernet Cable connected <b>OFF:</b> Ethernet Cable disconnected
4: Device State (Green)	Blink slowly (~2Hz)	Blink quickly



## UPDATE NEW FIRMWARE:

### ON HD67390-U-D1:

For update the firmware version of HD67390-U-D1 it is necessary to download from this link [www.adfweb.com/download/filefold/USB\\_SW67316\\_Update.zip](http://www.adfweb.com/download/filefold/USB_SW67316_Update.zip) the files that are necessary for do the operation.

After extracting the archive you have to launch the executable "USB\_SW67316\_Update.exe" file. The window that appears is shown in Fig. 3.

The steps to do are:

- ➔ Connect the device to an USB port and press the "Update COMport List" button;
- ➔ Select the COM port where the device is attached (USB port is virtualized like a serial);
- ➔ Press the "Update Device" button and select the file "FW\_USB\_SW67316.sim".

When in the log field appears this "Update made with success" sentence the updating is finished correctly.

If doesn't appears, try to repeat the operations.

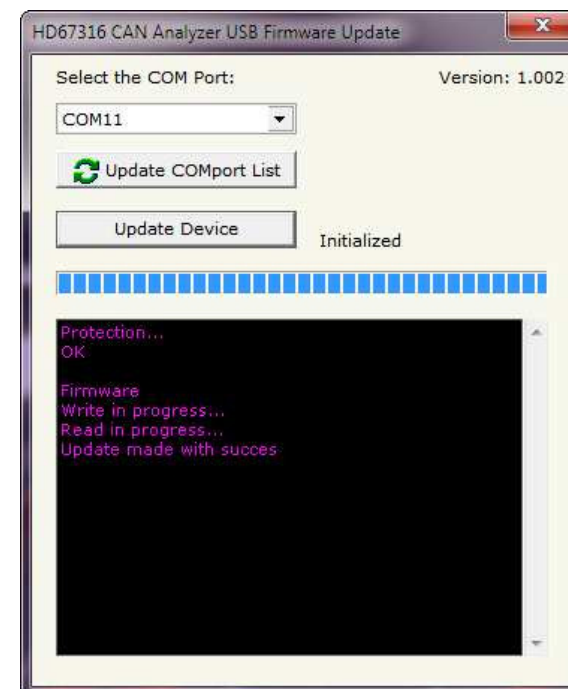


Figure 3: HD67316 CAN Analyzer USB Firmware Update

**ON HD67390-E-A1:**

For update the firmware version or changing the Ethernet parameters of HD673190-E-A1 it is necessary to download from this link [www.adfweb.com/download/filefold/SW67216.zip](http://www.adfweb.com/download/filefold/SW67216.zip) the files that are necessary for do the operation.

You have to install the SW67216 program, launch it (Fig.4 appears) and follow these instructions:

- Connect the Ethernet cable to the Analyzer and feed it;
- Go to "Function" → "SetPort", select "Ethernet Type (HD67316-E-A1)" and press the "OK" button;
- Go to "Ethernet" → "Configuration" and select or "By Serial" if you don't know the settings of Ethernet parameters; or "By Ethernet" if you know the IP Address and the Port.



Figure 4: Main window of SW67216

**USE OF "BY SERIAL"**

The window (Fig. 5) is divided in two sections, one for the data of Ethernet and the other for the actual update.

The parameters that are necessary to set are: "IP Address", "SubNet Mask", "Default Gateway", "Port".

After selecting the correct COM port you have to select the operations you want to do. You can select only "Firmware", only "Project" or both of them. Then press the "UPDATE" button.

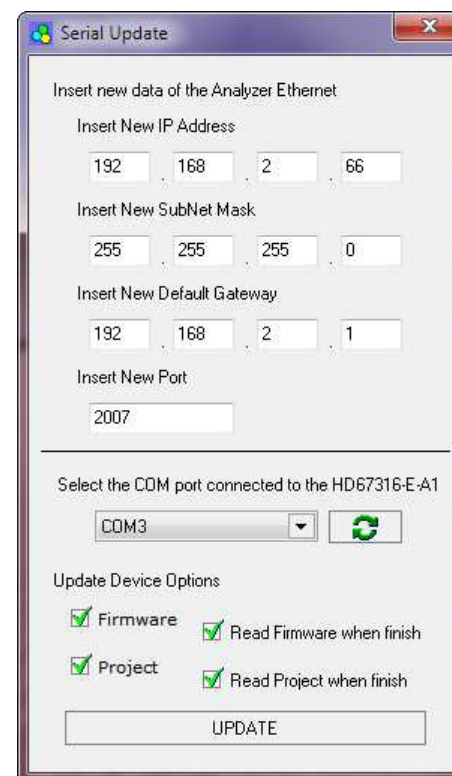
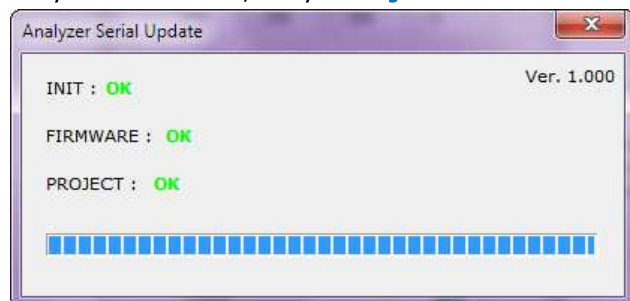


Figure 5: Update "By Serial" window



When appears Fig. 6 the update will be completed with success.

Figure 6: "Update Device" window

USE OF "BY ETHERNET"

The window (Fig. 7) is divided in two sections, one for the data of Ethernet and the other for the actual update.

The parameters that are necessary to set are: "IP Address", "SubNet Mask", "Default Gateway", "Port".

After inserting the actual "IP Address" and the actual "Port" you have to select the operations you want to do. You can select only "Firmware", only "Project" or both of them. Then press the "UPDATE" button.

When appears Fig. 8 the update will be completed with success.

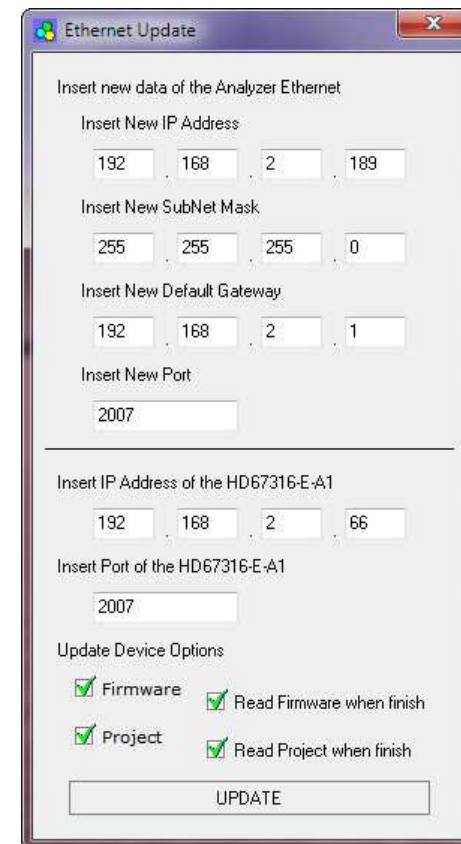


Figure 7: Update "By Ethernet" window

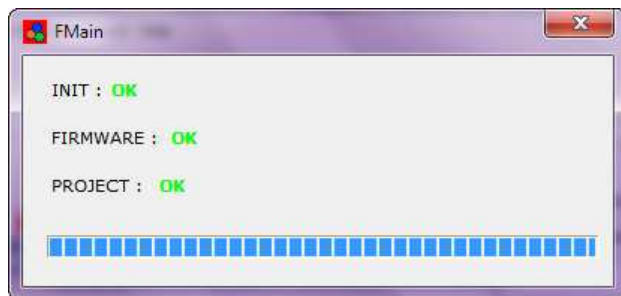


Figure 8: "Update Device" window

**CAN:**



The termination of CAN line, with a 120Ω resistor, in the HD67390-U-D1 is made by putting the “Dip2” of “Dip-Switch A” at “ON” position. The termination of CAN line, with a 120Ω resistor, in the HD67390-E-A1 is made by inserting a jumper like in the figure above.

Characteristics of the cable:

<b>DC parameter:</b>	Impedance	70 Ohm/m
<b>AC parameters:</b>	Impedance	120 Ohm/m
	Delay	5 ns/m
<b>Length</b>	<b>Baud Rate [bps]</b>	<b>Length MAX [m]</b>
	10 K	5000
	20 K	2500
	50 K	1000
	100 K	650
	125 K	500
	250 K	250
	500 K	100
	800 K	50
	1000 K	25

## PROTOCOL:

### Managements frames:

- **"READSTATUS"** = Read the firmware version of CAN Analyzer. The device returns a string with the actually version. The last version is **"CAN-ANALYZER VER P.901.02\r\n"**;
- **"GO TO BOOT"** = Put the device in Boot Mode. This state is used for update the firmware version;
- **"ENABLE INFO MESSAGE"** = Enable the sending, every ½ second, of a Info frame message. The device returns this string: **"ENABLED INFO MESSAGE \r\n"**;
- **"DISABLE INFO MESSAGE"** = Enable the sending, every ½ second, of a Info frame message. The device returns this string: **"DISABLED INFO MESSAGE\r\n"**;
- **"ENABLE MEDIA MODE"** = The Media is made counting the frames received from the starting of the device. The device returns this string: **"ENABLED MEDIA MODE\r\n"**;
- **"DISABLE MEDIA MODE"** = The Media is made watching how many frames are received in ½ second. The device returns this string: **"DISABLED MEDIA MODE\r\n"**;
- **"ENABLE PASSIVE MODE"** = The device is put on "Passive Mode". In this state it doesn't send any packet on CAN network. The device returns this string: **"ENABLED PASSIVE MODE\r\n"**;
- **"DISABLE PASSIVE MODE"** = The device is put out of "Passive Mode". In this state it sends packet on CAN network. The device returns this string: **"DISABLED PASSIVE MODE\r\n"**;
- **"ENABLE BIN MODE"** = The send and receive of a CAN packet is made in "Binary Mode". Also the Info Message is in "Binary Mode". All other frames are in ASCII. The HD67390-U-D1 returns this string: **"ENABLED BIN MODE SET\r\n"**. The HD67390-E-A1 returns this string: **"ENABLED BIN MODE \r\n"**;
- 
- **"DISABLE BIN MODE"** = The send and receive of a CAN packet is made in "ASCII Mode". Also the Info Message is in "ASCII Mode". All other frames are in ASCII. The device returns this string: **"DISABLED BIN MODE\r\n"**;
- **"ENABLE CAN RXTX"** = The send and receive of a CAN packet is enabled. The device returns this string: **"ENABLED DEVICE\r\n"**;
- **"DISABLE CAN RXTX"** = The send and receive of a CAN packet is disabled. The device returns this string: **"DISABLED DEVICE\r\n"**;
- **"BAUDRATE CAN="** = Set the Baudrate of CAN port. The available values are:
  - o 16k bps → 3E80 **"BAUDRATE\_CAN=00003E80"** and the confirmation is: **"BAUDRATE=16000\r\n"**;
  - o 20k bps → 4E20 **"BAUDRATE\_CAN=00004E20"** and the confirmation is: **"BAUDRATE=20000\r\n"**;
  - o 32k bps → 7D00 **"BAUDRATE\_CAN=00007D00"** and the confirmation is: **"BAUDRATE=32000\r\n"**;
  - o 40k bps → 9C40 **"BAUDRATE\_CAN=00009C40"** and the confirmation is: **"BAUDRATE=40000\r\n"**;
  - o 50k bps → C350 **"BAUDRATE\_CAN=0000C350"** and the confirmation is: **"BAUDRATE=50000\r\n"**;
  - o 80k bps → 13880 **"BAUDRATE\_CAN=00013880"** and the confirmation is: **"BAUDRATE=80000\r\n"**;

Electronic Devices

- o 100k bps → 186A0 "BAUDRATE\_CAN=000186A0" and the confirmation is: "BAUDRATE=100000\r\n";
- o 125k bps → 1E848 "BAUDRATE\_CAN=0001E848" and the confirmation is: "BAUDRATE=125000\r\n";
- o 160k bps → 27100 "BAUDRATE\_CAN=00027100" and the confirmation is: "BAUDRATE=160000\r\n";
- o 200k bps → 30D40 "BAUDRATE\_CAN=00030D40" and the confirmation is: "BAUDRATE=200000\r\n";
  
- o 250k bps → 3D090 "BAUDRATE\_CAN=0003D090" and the confirmation is: "BAUDRATE=250000\r\n";
- o 320k bps → 4E200 "BAUDRATE\_CAN=0004E200" and the confirmation is: "BAUDRATE=320000\r\n";
- o 400k bps → 61A80 "BAUDRATE\_CAN=00061A80" and the confirmation is: "BAUDRATE=400000\r\n";
- o 500k bps → 7A120 "BAUDRATE\_CAN=0007A120" and the confirmation is: "BAUDRATE=500000\r\n";
- o 666k bps → A2990 "BAUDRATE\_CAN=000A2990" and the confirmation is: "BAUDRATE=666000\r\n";
- o 800k bps → C3500 "BAUDRATE\_CAN=000C3500" and the confirmation is: "BAUDRATE=800000\r\n";
- o 1000k bps → F4240 "BAUDRATE\_CAN=000F4240" and the confirmation is: "BAUDRATE=1000000\r\n";
- "**MAPPA11=**" = Set the filter for the 2.0A functioning mode (11bit). The filter must be written in hex format (so it is necessary to use numbers from 0 to F). Every byte is used for mask 8 Cob-ID. The confirmation is: "**MAPPA11 IMPOSTATA\r\n**".

Here some examples:

- o "**MAPPA11=01**" the Analyzer shows (send to USB) only the frame with Cob-ID=0x0. So in this case:

<b>Hex</b>	01							
<b>Binary</b>	0	0	0	0	0	0	0	1
<b>Cob-ID</b>	7	6	5	4	3	2	1	0

- o "**MAPPA11=1F**" the Analyzer shows only the Cob-ID 0x0, 0x1, 0x2, 0x3, 0x4;
- o "**MAPPA11=10**" the Analyzer shows only the Cob-ID 0x4;
- o "**MAPPA11=11**" the Analyzer shows only the Cob-ID 0x0, 0x4;
- o "**MAPPA11=0201**" the Analyzer shows only the Cob-ID 0x1, 0x8. So in this case:

<b>Hex</b>	02							
<b>Binary</b>	0	0	0	0	0	0	1	0
<b>Cob-ID</b>	7	6	5	4	3	2	1	0

<b>Hex</b>	01							
<b>Binary</b>	0	0	0	0	0	0	0	1
<b>Cob-ID</b>	F	E	D	C	B	A	9	8

- o "**MAPPA11=030F18**" the Analyzer shows only the Cob-ID 0x0, 0x1, 0x8, 0x9, 0xA, 0xB, 0x13, 0x14. So in this case:

<b>Hex</b>	03							
<b>Binary</b>	0	0	0	0	0	0	1	1
<b>Cob-ID</b>	7	6	5	4	3	2	1	0

<b>Hex</b>	0F							
<b>Binary</b>	0	0	0	0	1	1	1	1
<b>Cob-ID</b>	F	E	D	C	B	A	9	8



<b>Hex</b>	18							
<b>Binary</b>	0	0	0	1	1	0	0	0
<b>Cob-ID</b>	17	16	15	14	13	12	11	10

- **"MAPPA29="** = Set the filter for the 2.0B functioning mode (29bit). There are two main types of filters: "Cob-Id Filter" or "Mask Filter". The "Cob-Id Filter" can be "Positive" or "Negative". The first four values after **"MAPPA29="** can assume three values **"MAPPA29=02000000"** for set the "Mask Filter", **"MAPPA29=01000000"** for "Positive Cob-Id Filter" or **"MAPPA29=01FFFFFF"** for "Negative Cob-Id Filter". The confirmation is: **"MAPPA29 IMPOSTATA\r\n"**.
  - o Case **"MAPPA29=02000000"**: It is possible to set up to 10 filters. It is possible to write 0,1,X. The maximum length of the string is 29 characters. You have to create two 32-bit numbers following these instruction:

1° number (Mask1): where val is the array that contain the string of 0,1,x.

```
Mask1=0; //32bit value
j=0;
for (i=len;i>0;i--) {
    if (val[i]!='0') Mask1=Mask1 | (1<<j);
    j++;
}
```

2° number (Mask2): where val is the array that contain the string of 0,1,x.

```
Mask2=0; //32bit value
j=0;
for (i=len;i>0;i--) {
    if (val[i]='1') Mask1=Mask1 | (1<<j);
    j++;
}
```

Here some examples:

- Mask written: 11000111111101100101000000001  
 Mask1 after decoding: 11000111111101100101000000001 [18FECA01]  
 Mask2 after decoding: 11000111111101100101000000001 [18FECA01]  
**"MAPPA29=0200000018FECA0118FECA01"**

Cob-Id Received	Result
18FECA01	Passed
18FECA00	Not passed
00FEDE71	Not passed

- Mask written: 1100011111101100101000000xx1  
 Mask1 after decoding: 1100011111101100101000000111 [18FECA07]  
 Mask2 after decoding: 1100011111101100101000000001 [18FECA01]  
 Mask written: 11000000000001100101000000001  
 Mask1 after decoding: 11000000000001100101000000001 [1800CA01]  
 Mask2 after decoding: 11000000000001100101000000001 [1800CA01]  
**"MAPPA29=0200000018FECA0718FECA011800CA011800CA01"**

Cob-Id Received	Result
18FECA01	Passed
18FECA03	Passed
1800CA01	passed
18FECA06	Not passed
1800CA00	Not passed

- Mask written: 11x001111111011001010000000x1  
 Mask1 after decoding: 1110011111101100101000000011 [1CFECA03]  
 Mask2 after decoding: 1100011111101100101000000001 [18FECA01]  
**"MAPPA29=020000001CFECA0318FECA01"**

Cob-Id Received	Result
18FECA01	Passed
18FECA03	Passed
1CFECA01	Passed
1CFECA03	Passed
18FECA02	Not passed
16FECA01	Not passed

- Case **"MAPPA29=01FFFFFF"**: It is possible to set up to 63 filters. In this case only the Cob-Ids written are sent to USB port. The Cob-Id is written in hexadecimal format (from 0 to 01FFFFFF). Here some examples:
  - **"MAPPA29=01FFFFFF01FECA00"**

Cob-Id Received	Result
01FECA00	Passed
01FECA01	Not passed



- **"MAPPA29=01FFFFFF01FECA0000FEDE71"**

Cob-Id Received	Result
01FECA00	Passed
00FEDE71	Passed
01FECA01	Not passed
00FEDE70	Not passed
101	Not passed

- Case **"MAPPA29=01000000"**: It is possible to set up to 63 filters. In this case all the Cob-ID except the ones written are sent to the USB port. The Cob-Id is written in hexadecimal format (from 0 to 01FFFFFF). Here some examples:

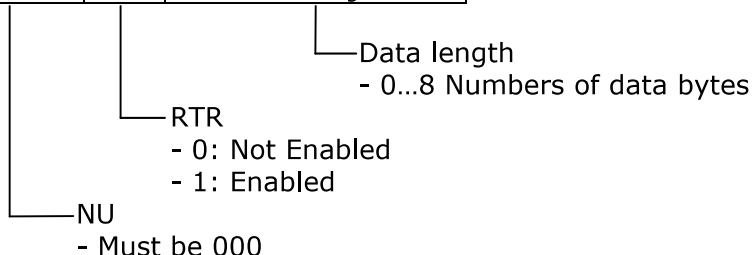
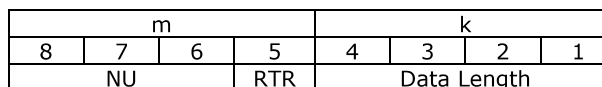
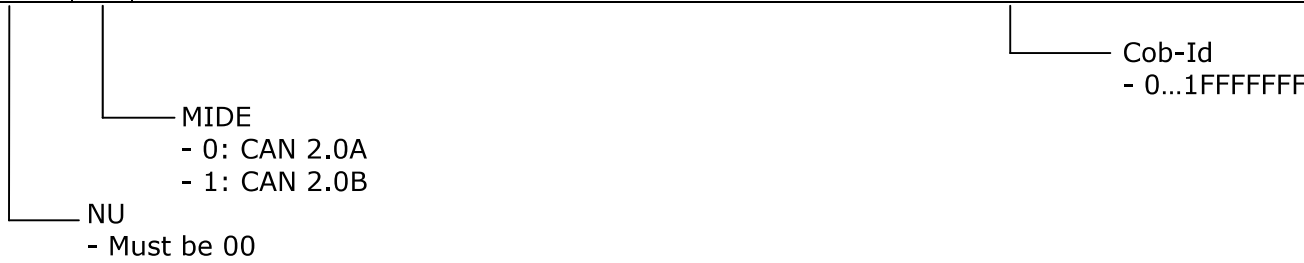
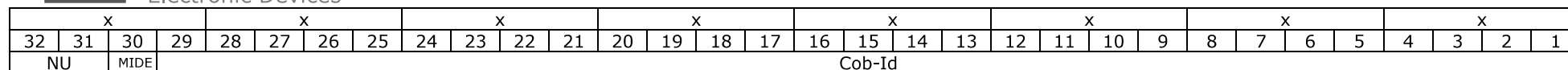
- **"MAPPA29=0100000001FECA00"**

Cob-Id Received	Result
01FECA01	Passed
01FECA00	Not passed

- **"MAPPA29=0100000001FECA0000FEDE71"**

Cob-Id Received	Result
01FECA01	Passed
00FEDE70	Passed
101	passed
01FECA00	Not passed
00FEDE71	Not passed

- **"SEND\_PACKET=xxxxxxxxxyyyyyyyzzzzzzmk"** = Sends a CAN frame. Instead of "xxxxxxxx" is necessary to insert the Cob-ID (from 0x00000000 to 0x1FFFFFFF). If the CAN type is 2.0B there is the necessity to sum 0x20000000 to the Cob-ID otherwise (CAN 2.0A) this isn't necessary. If CAN 2.0A is used, the original Cob-Id must be shifted at left side of 18 positions. The "yyyyyy" are the four most significative bytes of data and the "zzzzzz" are the four less significative bytes of data. The order of bytes is reversed. The "m" is used for the RTR bit. If 0 the RTR isn't enabled, if 1 the RTR is enabled. The "k" is the number of bytes of data and can assume a value from 0 to 8. If the frame is successfully transmitted a frame is returned for confirm that. This packet format depends of the mode selected (Binary or ASCII).



Here some examples:

- CAN Type: 2.0A; Cob-Id: 0x181; Data: 0102030405060708; RTR: 0; Data Length: 8  
**"SEND\_PACKET=06040000040302010807060508"**
- CAN Type: 2.0A; Cob-Id: 0x03A; Data: 15263748; RTR: 0; Data Length: 4  
**"SEND\_PACKET=00E80000483726150000000004"**
- CAN Type: 2.0B; Cob-Id: 0x18CAFE88; Data: 010203040506; RTR: 0; Data Length: 6  
**"SEND\_PACKET=38CAFE88040302010000060506"**
- CAN Type: 2.0B; Cob-Id: 0x18CAFE88; Data: 44; RTR: 0; Data Length: 1  
**"SEND\_PACKET=38000088000000440000000001"**
- CAN Type: 2.0B; Cob-Id: 0x18CAFE88; Data: 010203040506; RTR: 1; Data Length: 6  
**"SEND\_PACKET=38000088040302010000060516"**

The confirmation frame format is the same of "Data Packet Received" (which is explained below).

- **Data Packet Received:** This packet contain all the information of a received CAN frame.

ASCII MODE FUNCTIONING:

"PR=xxxxxxxx zzzzzzzzwwwwwww qk rrrrrrrr\r\n"

PR=	xxxxxxxx		zz	zz	zz	zz	ww	ww	ww	ww		q	k	rrrrrrrr	\r	\n
Header	Cob-Id + MIDE	Space	MSB-3	MSB-2	MSB-1	MSB	LSB	MSB-6	MSB-5	MSB-4	Space	RTR	Length	Time	13	10

- o PR= : Header;
- o xxxxxxxx : COB-ID + MIDE

x			x				x					x				x				x														
32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1			
NU		MIDE																													Cob-Id			

- **Cob-Id:**
  - 0...1FFFFFFF
- **MIDE:**
  - 0 = CAN 2.0A
  - 1 = CAN 2.0B
- **NU:**
  - Must be 00
- o zzzzzzzz : the four most significant bytes of data. The order of bytes is reversed.
- o wwwwww : the four less significant bytes of data. The order of bytes is reversed.
- o q : RTR
  - 0 = Not Enabled
  - 1 = Enabled
- o k : Data Length
  - 0...8 = Number of bytes of Data
- o rrrrrrrr : Tenths of milliseconds since the device was turned on
- o \r\n: Tail (13 10 "0x0D 0x0A")

Here some examples:

- "PR=38FECA08 0403020108070605 08 00000005\r\n" the decoding of frame received is the follow: CAN Type: 2.0B; Cob-Id: 0x18FECA08; Data: 0102030405060708; RTR: 0; Data Length: 8; Time: 5 tenth of ms
- "PR=06040000 1413121100001615 06 0000000A\r\n" the decoding of frame received is the follow: CAN Type: 2.0A; Cob-Id: 0x181; Data: 111213141516; RTR: 0; Data Length: 6; Time: 10 tenth of ms

**BINARY MODE FUNCTIONING:**

The length of the frame is variable. It depends of the type of CAN (2.0A or 2.0B) and the number of data bytes. The minimum length is 10 bytes and the maximum is 20 bytes.

MSB	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	LSB
0x01	0xPP	0xWW	0xWW	0xWW	0xWW	0xOO	0xOO	0xOO	0xOO	0xRR	0xRR	0xRR	0xRR	0xMM	0xMM	0xMM	0xMM	0xYY	0xLL
Header	Info	Cob-Id				Data								Time				Check-Sum	

MSB	...	...	...	...	...	...	...	...	LSB
0x01	0xPP	0xWW	0xWW	0xMM	0xMM	0xMM	0xMM	0xYY	0xLL
Header	Info	Cob-Id		Time				Check-Sum	

- 0xPP: Info

<b>Bit</b>	<b>8</b>	<b>7</b>	<b>6</b>	<b>5</b>	<b>4</b>	<b>3</b>	<b>2</b>	<b>1</b>
<b>Sense</b>		MIDE	Packet type	RTR	Length			
	Info							

- **MIDE:**  
0 = CAN 2.0A  
1 = CAN 2.0B
- **RTR:**  
0 = Not Enabled  
1 = Enabled
- **Length:**  
0...8 = Number of bytes of Data

- 0xWWWWWWWW: Cob-Id of CAN frame. The value can be from 0x000 to 0x7FF in the case of CAN 2.0A or from 0x00000000 to 0x1FFFFFFF in the case of CAN2.0B.
- 0xOOOOOOOO: the four less significant bytes of data. The order of bytes is reversed.
- 0xRRRRRRRR: the four most significant bytes of data. The order of bytes is reversed.
- 0xMMMMMMMM: Tenths of milliseconds since the device was turned on.

- Check-Sum: Is the sum of all bytes of this frame except the 0xYY 0xLL (0xYYLL = 0x01+0xPP+0xWW+0xWW+0xWW+0xWW+0xOO+0xOO+0xOO+0xOO+0xRR+0xRR+0xRR+0xRR+0xMM+0xMM+0xMM+0xMM).

Here some examples:

- "0x01 0x48 0x18 0xFE 0xCA 0x08 0x08 0x07 0x06 0x05 0x04 0x03 0x02 0x01 0x00 0x00 0x00 0x05 0x02 0x5A" the decoding of frame received is the follow: CAN Type: 2.0B; Cob-Id: 0x18FECA08; Data: 0102030405060708; RTR: 0; Data Length: 8; Time: 5 tenth of ms
- "0x01 0x06 0x01 0x81 0x16 0x15 0x14 0x13 0x12 0x11 0x00 0x00 0x00 0x0A 0x01 0x08" the decoding of frame received is the follow: CAN Type: 2.0A; Cob-Id: 0x181; Data: 111213141516; RTR: 0; Data Length: 6; Time: 10 tenth of ms

- **Info Packet Received:** This packet contain the information of the status of CAN.

ASCII MODE FUNCTIONING:

"PI=xxxxxxx yyyyyyy zzzzzzz aaaaaaa bbbbbbb\r\n"

PI=	xxxxxxx		yyyyyyy		zzzzzzz		aaaaaaa		bbbbbbb	\r	\n
Header	# of Received packet	Space	# of Discarded packet	Space	# of CAN error	Space	CAN Status	Space	Bus Load	13	10

- PI= : Header;
- xxxxxxx: Number of CAN frame received (hexadecimal format)
- yyyyyyy: Number of CAN frame discarded (filtered) (hexadecimal format)
- zzzzzzz: Number of CAN frame with error(hexadecimal format)
- aaaaaaa: CAN Status

Bit	32...17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
Sense	////////	OVSLY	TBSY	RBSY	BERR	FERR	AERR	SERR	CERR	TSTP	TOVF	WAKEUP	SLEEP	BOFF	ERRP	WARN	ERRA

- **ERRA:** Error Active Mode  
 0 = CAN controller is not in Error Active Mode.  
 1 = CAN controller is in Error Active Mode.  
 This flag is set depending on TEC and REC counter values. It is set when node is neither in Error Passive Mode nor in Bus Off Mode.
- **WARN:** Warning Limit  
 0 = CAN controller Warning Limit is not reached.  
 1 = CAN controller Warning Limit is reached.  
 This flag is set depending on TEC and REC counter values. It is set when at least one of the counter values exceeds 96.

- **ERRP:** Error Passive Mode  
 0 = CAN controller is not in Error Passive Mode.  
 1 = CAN controller is in Error Passive Mode.  
 This flag is set depending on TEC and REC counters values.  
 A node is error passive when TEC counter is greater or equal to 128 (decimal) or when the REC counter is greater or equal to 128 (decimal).
- **BOFF:** Bus Off Mode  
 0 = CAN controller is not in Bus Off Mode.  
 1 = CAN controller is in Bus Off Mode.  
 This flag is set depending on TEC counter value. A node is bus off when TEC counter is greater or equal to 256 (decimal).
- **SLEEP:** CAN controller in Low power Mode  
 0 = CAN controller is not in low power mode.  
 1 = CAN controller is in low power mode.
- **WAKEUP:** CAN controller is not in Low power Mode  
 0 = CAN controller is in low power mode.  
 1 = CAN controller is not in low power mode.  
 When a WAKEUP event occurs, the CAN controller is synchronized with the bus activity. Messages can be transmitted or received. The CAN controller clock must be available when a WAKEUP event occurs.
- **TOVF:** Timer Overflow  
 0 = The timer has not rolled-over FFFFh to 0000h.  
 1 = The timer rolls-over FFFFh to 0000h.
- **TSTP** Timestamp  
 0 = No bus activity has been detected.  
 1 = A start of frame or an end of frame has been detected (according to the TEOF field in the CAN\_MR register).
- **CERR:** Mailbox CRC Error  
 0 = No CRC error occurred during a previous transfer.  
 1 = A CRC error occurred during a previous transfer.  
 A CRC error has been detected during last reception.
- **SERR:** Mailbox Stuffing Error  
 0 = No stuffing error occurred during a previous transfer.  
 1 = A stuffing error occurred during a previous transfer.  
 A form error results from the detection of more than five consecutive bit with the same polarity.
- **AERR:** Acknowledgment Error  
 0 = No acknowledgment error occurred during a previous transfer.  
 1 = An acknowledgment error occurred during a previous transfer.

- An acknowledgment error is detected when no detection of the dominant bit in the acknowledge slot occurs.
- **FERR:** Form Error
    - 0 = No form error occurred during a previous transfer
    - 1 = A form error occurred during a previous transfer

A form error results from violations on one or more of the fixed form of the following bit fields:

    - CRC delimiter
    - ACK delimiter
    - End of frame
    - Error delimiter
    - Overload delimiter
  - **BERR:** Bit Error
    - 0 = No bit error occurred during a previous transfer.
    - 1 = A bit error occurred during a previous transfer.

A bit error is set when the bit value monitored on the line is different from the bit value sent.
  - **RBSY:** Receiver busy
    - 0 = CAN receiver is not receiving a frame.
    - 1 = CAN receiver is receiving a frame.

Receiver busy. This status bit is set by hardware while CAN receiver is acquiring or monitoring a frame (remote, data, overload or error frame).
  - **TBSY:** Transmitter busy
    - 0 = CAN transmitter is not transmitting a frame.
    - 1 = CAN transmitter is transmitting a frame.

Transmitter busy. This status bit is set by hardware while CAN transmitter is generating a frame (remote, data, overload or error frame).
  - **OVLSY:** Overload busy
    - 0 = CAN transmitter is not transmitting an overload frame.
    - 1 = CAN transmitter is transmitting a overload frame.
- bbbbbbbb: CAN Bus Load
    - This number can assume values from 0 (0%) to 200 (100%) of bandwidth occupation.
  - \r\n: Tail (13 10 "0x0D 0x0A")

**BINARY MODE FUNCTIONING:**

In this mode the Info Frame that the Analyzer sends is packed into a 21-bytes frame that have the following structure:

**"0x01 0x20 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x21"**

MSB	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	LSB
0x01	0x20	0xRR	0xRR	0xRR	0xRR	0xTT	0xTT	0xTT	0xTT	0xFF	0xFF	0xFF	0xFF	0xSS	0xSS	0xSS	0xSS	0xMM	0xYY	0xLL
Header		# of Received packet				# of Discarded packet				# of CAN error				CAN Status				Bus Load	Check-Sum	

- o 0x01 0x20: Header
- o 0xRRRRRRRR: Number of CAN frame received (hexadecimal format)
- o 0xTTTTTTTT: Number of CAN frame discarded (filtered) (hexadecimal format)
- o 0xFFFFFFFF: Number of CAN frame with error(hexadecimal format)
- o 0xSSSSSSSS: CAN Status
- o 0xMM: CAN Bus Load  
This number can assume values from 0 (0%) to 200 (100%) of bandwidth occupation.
- o Check-Sum: Is the sum of all bytes of this frame except the 0xYY 0xLL (0xYYLL = 0x01+0x02+0xRR+0xRR+0xRR+0xRR+0xTT+0xTT+0xTT+0xTT+0xFF+0xFF+0xFF+0xFF+0xSS+0xSS+0xSS+0xSS+0xMM).

For the USB device, after install the Driver, on your program you must open a Serial Port with these settings:

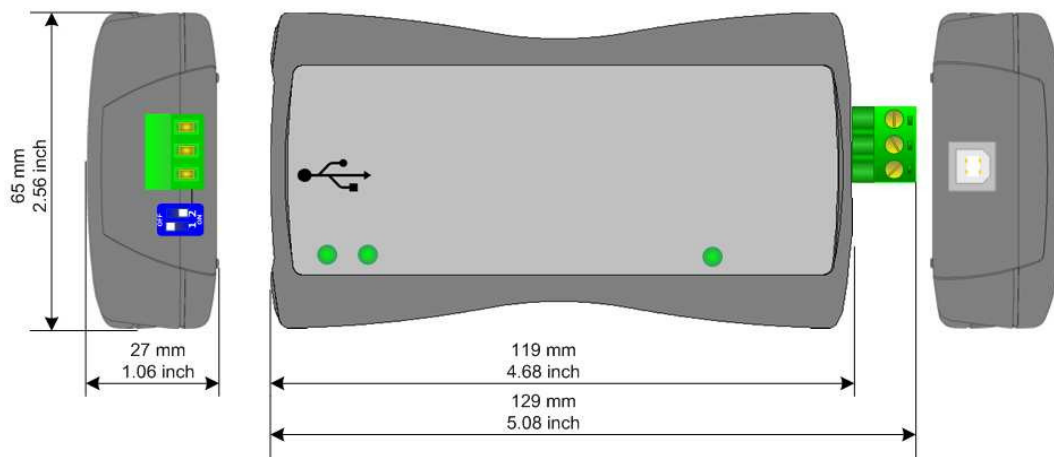
- Baud rate: 3000000 bps;
- Parity: None;
- Data: 8 bit
- Stop-Bit: 1 bit;
- Flow Control: No.

For the Ethernet device, you have to use the UDP protocol.

For both devices the minimum frames that must be sent are: "**ENABLE CAN RXTX**" "**BAUDRATE CAN=**", "**MAPPA11=**", "**MAPPA29=**".

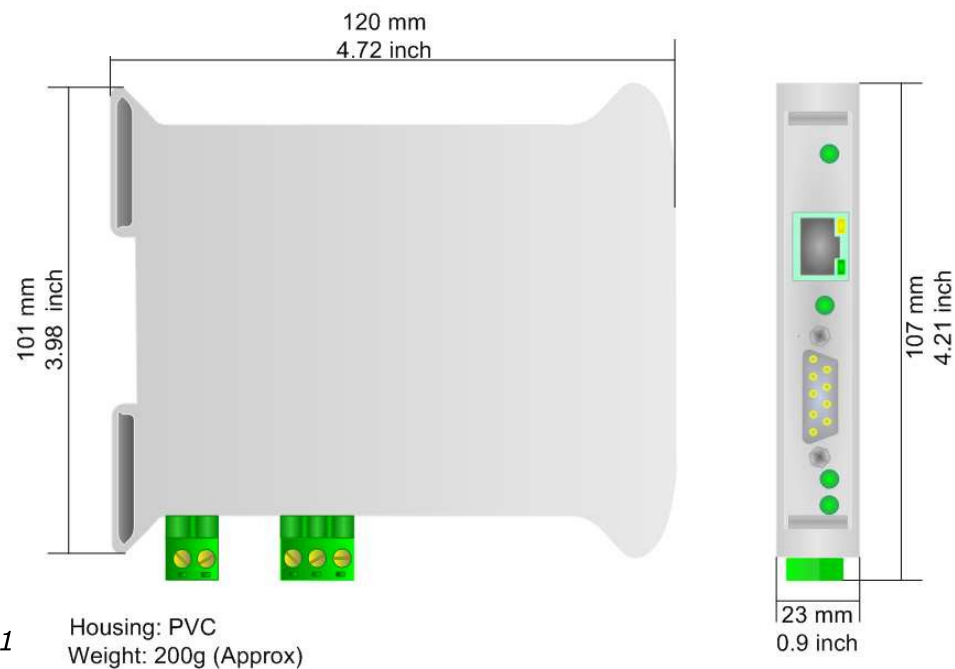


**MECHANICAL DIMENSIONS:**



Housing: PVC  
Weight: 200g (Approx)

Figure 9: Mechanical dimensions scheme for HD67390-U-D1



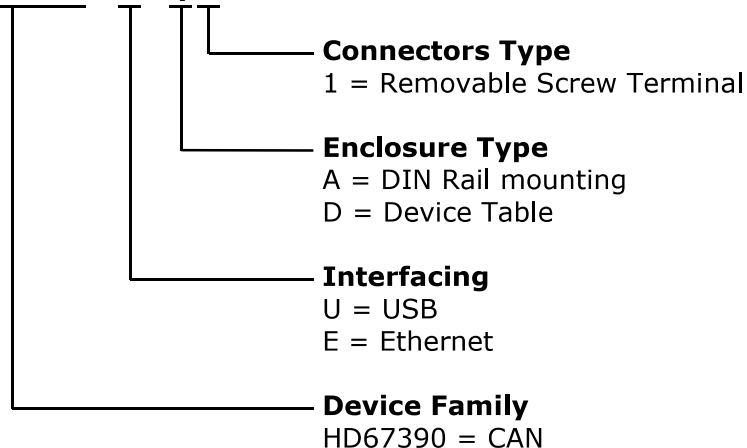
Housing: PVC  
Weight: 200g (Approx)

Figure 10: Mechanical dimensions scheme for HD67390-E-A1

**ORDERING INFORMATIONS:**

The ordering part number is formed by a valid combination of the following:

**HD67390 - x - y z**



Order Code: **HD67390-U-D1** - CAN / USB – Converter, for CAN Analysis use (device table)

Order Code: **HD67390-E-A1** - CAN / Ethernet – Converter, for CAN Analysis use (35mm DIN Rail mounting)

**ACCESSORIES:**

Order Code: **AC34001** - Rail DIN - Power Supply 220/240V AC 50/60Hz – 12 V AC

Order Code: **AC34002** - Rail DIN - Power Supply 110V AC 50/60Hz – 12 V AC

## DISCLAIMER

All technical content within this document can be modified without notice. The content of the document content is a recurring audit. For losses due to fire, earthquake, third party access or other accidents, or intentional or accidental abuse, misuse, or use under abnormal conditions repairs are charged to the user. ADFweb.com S.r.l. will not be liable for accidental loss of use or inability to use this product, such as loss of business income. ADFweb.com S.r.l. shall not be liable for consequences of improper use.

## OTHER REGULATIONS AND STANDARDS

### WEEE INFORMATION



Disposal of old electrical and electronic equipment (as in the European Union and other European countries with separate collection systems).

— This symbol on the product or on its packaging indicates that this product may not be treated as household rubbish. Instead, it should be taken to an applicable collection point for the recycling of electrical and electronic equipment. If the product is disposed correctly, you will help prevent potential negative environmental factors and human health, which could otherwise be caused by inappropriate disposal. The recycling of materials will help to conserve natural resources. For more information about recycling this product, please contact your local city office, your household waste disposal service or the shop where you purchased the product.

### RESTRICTION OF HAZARDOUS SUBSTANCES DIRECTIVE



The device respects the 2002/95/EC Directive on the restriction of the use of certain hazardous substances in electrical and electronic equipment (commonly referred to as Restriction of Hazardous Substances Directive or RoHS).

### CE MARKING



The product conforms with the essential requirements of the applicable EC directives.



### WARRANTIES AND TECHNICAL SUPPORT:

For fast and easy technical support for your ADFweb.com SRL products, consult our internet support at [www.adfweb.com](http://www.adfweb.com). Otherwise contact us at the address [support@adfweb.com](mailto:support@adfweb.com)

### RETURN POLICY:

If while using your product you have any problem and you wish to exchange or repair it, please do the following:

- 1) Obtain a Product Return Number (PRN) from our internet support at [www.adfweb.com](http://www.adfweb.com). Together with the request, you need to provide detailed information about the problem.
- 2) Send the product to the address provided with the PRN, having prepaid the shipping costs (shipment costs billed to us will not be accepted).

If the product is within the warranty of twelve months, it will be repaired or exchanged and returned within three weeks. If the product is no longer under warranty, you will receive a repair estimate.

### PRODUCTS AND RELATED DOCUMENTS:

Part	Description	URL
HD67121	Gateway CANopen / Canopen	<a href="http://www.adfweb.com?product=HD67121">www.adfweb.com?product=HD67121</a>
HD67001	Gateway CANopen / Modbus – RTU Master	<a href="http://www.adfweb.com?product=HD67001">www.adfweb.com?product=HD67001</a>
HD67504 HD67505	Gateway CANopen / Modbus – Ethernet TCP	<a href="http://www.adfweb.com?product=HD67504">www.adfweb.com?product=HD67504</a>
HD67134	Gateway CANopen / DeviceNet	<a href="http://www.adfweb.com?product=HD67134">www.adfweb.com?product=HD67134</a>
HD67117	CAN bus Repeater	<a href="http://www.adfweb.com?product=HD67117">www.adfweb.com?product=HD67117</a>
HD67316	CAN bus Analyzer	<a href="http://www.adfweb.com?product=HD67316">www.adfweb.com?product=HD67316</a>