

# IBus

[Start](#)[Up](#)[IBusCars](#)[IBus Devices & Operat](#)[BM buttons](#)[Steering Wheel Buttor](#)

## Links from this page:

[IBusCars](#)[IBus Devices & Operations](#)[General](#)[BM buttons](#)[Steering Wheel Buttons](#)

Last changed 31 Mars 2009 (still needs to be tided up)

## General

Serial communications on the bus are 9600 bps, 8 data bits, Even parity, 1 stop bit.

Electrical interface is single line 12V with open collector devices connected to it. Any device can start sending when the bus is idle, but if it discovers that line is pulled low without doing it itself, it has to abort. This is how priority is solved as the first byte sent is its own ID.



The structure of an Ibus packet is the following :

<i>Source Device ID</i>	The device which needs to send a message to another device
<i>Length</i>	The length of the packet whithout Source ID and length it-self.
<i>Destination Device ID</i>	The device which must receive the message
<i>Data</i>	The message to send to Destination ID  The first byte is a command byte or message id, and the following bytes attributes to that command or request.
<i>XOR CheckSum</i>	This byte is used to check the integrity of the message. The receiver will compare that value with its own

A very nice [document](#) is found at the [HackTheIbus](#) group at yahoo in the file section.

I got a link to a site from HackTheIbus that seems to be a [goldmine](#). Interpret it with [AltaVista bablefish](#).

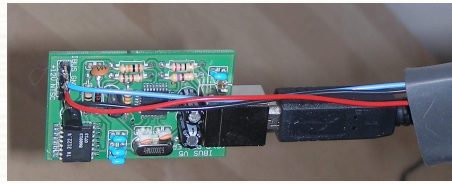
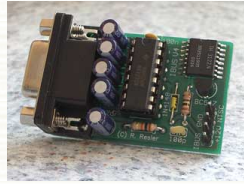
Codes can also be found at <http://www.baso.no> and in hacktheibus database.

IBus Wiki [here](#).

---

## IBus Interface

Interface of steering wheel button etc with IBus interface.



[Link to Resler](#)

---

## IBus Analyser

IBA can analyze IBus traffic and send commands that you enter once or periodically. [Download](#)

A simple terminal emulator like [realtterm](#) can be useful as development and debugging tool.

---

## IBusComM

An exe file created with Visual Basic that reads IBus messages. [Download.](#)

- 1) IBusComM will send key press sequences to window applications in response to IBus messages.
- 2) IBusComM can control RTS that is used by CIB to select video source,
- 3) send IBus messages itself.
- 4) and send windows messages (haven't tried that yet)

***Everything is controlled by an ini file. I have managed to get IBusComM to send a 'poll' message when it gets an 'I'm alive' message, pull RTS high, and to interpret turning of the left bordmonitor nob as a command to send F9 and F10 to windows media player! (decrease and increase volume)***

***Thank you [Franck Touanen](#) for writing and sharing this code!***

Eventually I think that I have to write some logic into this app. Pressing one button might require certain actions when running NAV and another when the mediaplayer is active. I'm not sure that IBusCom can handle that as is. Can RoadRunner do that for instance? Maybe CARMes and what have you has already a solution to that. We'll see when I get there.

For the time beeing its quite enough to know (ruffly) what IBusComM can do, and that it keeps to Bordmonitor out of sleep and selects proper video input!

[My IBus service installation](#)

---

## IBus Messages

## Bordmonitor Button Messages

### [Bordmonitor Buttons](#)

## Steering wheel buttons

### [Steering wheel Buttons](#)

## Specific Messages that I do use

***The sideview mirror selector switch on the left door armrest will send these messages: (Thanks saft6luck) I will use this message to turn on rearview camera***

9B 04 51    6D 40 E3 // right  
9B 04 51    6D 80 23 // left

*saft6luck: "For my LH driven car the 0x9B is the left and 0x51 is the right mirror module. Btw. I duno if this changes for the RH driven car. The mirrors are controlled by these modules and the buttons are read in by the 0x9B module in my case."*

***When key is inserted into the ignition lock these messages are sent. I will use this to detect ignition on.***

44 05 BF 74 04 01 8F // in  
44 05 BF 74 00 FF 75 // out

> GT --> IKE : Ignition status request (Jochen)

Answer:  
80 04 BF 11 03 29

80 = IKE  
04 = length  
BF = Global  
11 = Ignition status  
03 = Data  
29 = checksum

Data: (bit mapped)  
Bit 1 = KL\_R (Pos1\_Acc)  
Bit 2 = KL\_15 (Pos2\_On)  
Bit 3 = KL\_50 (Pos3\_Start)

***When gear is put in reverse and ignition is on these messages are sent: (Thanks saft6luck) I will use this to turn on rearview camera***

Note that the four red marked bytes are really a bitvector. Only the bold one bit is significant for these messages.

80 0A BF 13 02 10 00 00 00 00 38 0C // in  
80 0A BF 13 02 00 00 00 00 00 38 1C //out

handbrake on  
oil pressure low  
brake pads worn  
transmission emergency program  
gearbox not in P  
motor running  
vehicle driving  
reverse not plausible  
Gear: R  
Gear: 1  
Gear: 2  
Gear: D  
alarm horn on  
immobiliser on  
Aux heating on  
Aux ventillation on  
Temp deg F

---

***When a button is pressed on the remote key (#1) these messages are sent. I will use this to start the PC as soon as possible.***

00 04 BF 72 22 EB // unlock in  
00 04 BF 72 12 DB // lock in  
00 04 BF 72 42 8B // boot in  
00 04 BF 72 02 CB // out

---

***LED control messages (Thanks, Priie and saft6luck) I will use this to indicate System Status***





Ex.  
C8h 04h E7h 2Bh DB1 crc

DB1 is controlling the LEDs:

DB1: bit  
76 54 32 10  
00 gg yy rr

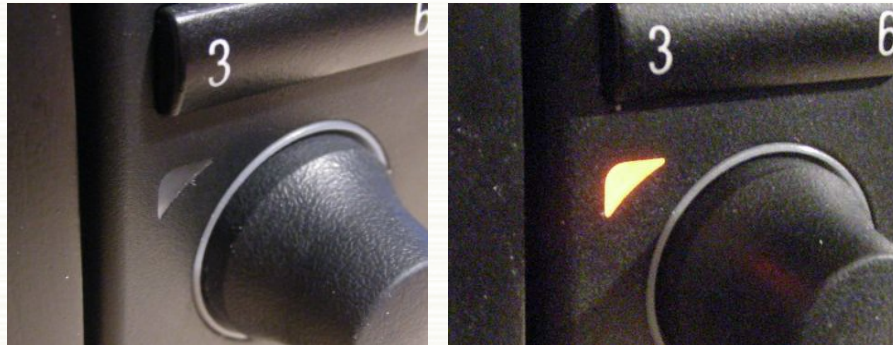
00: off  
01: on  
11: on/flashing

*The LEDs are for the telephone connection - all three will blink for the Bluetooth pairing, I don't know if they blink for hardwired phone, the right LED is red, and blinks if there is NO cell coverage on the connected phone. The middle LED is green and lights up when you have active connection on the phone.*

*Indicator lamps for phone\* mode  
Yellow: Call is being routed through alternate network  
Green: Call is connected  
Red: Phone not available  
Flashes: Phone is not registered with a service provider  
Please consult the separate manual for instructions on operating the telephone. [http://www.e38.org/BMW\\_nav\\_owners\\_manual2004.pdf](http://www.e38.org/BMW_nav_owners_manual2004.pdf)*

---

**Turning on an off LED next to left knob (Thank you Saft6luck). I will use this to indicate PC up and running**



*"For the LED on the left button: it is indicating the status of the radio (on/off), at least for my LHD 16:9 board monitor.*

*You can use message 0x4A, e.g. 0x68 LL 0xF0 0x4A 0xFF CS to switch it on and DB1=0x00 to switch it off."*

---

**Speed (Thank you, Richard Gaunt) I will use this to create PWM for the GPS**

80 05 BF 18 ss rr cc

ss = speed / 2 [km/h] (512km/h max)  
rr = revs / 100 rpm

---

**Set Time (thanks enforcer) I might use this to set car time from PC/GPS time.**

GT telling IKE to set the time:

3B 06 80 40 01 0C 3B cc  
GT -> IKE : On-board computer set data: Set Time = 12:59  
40 = OBC Set data  
01 = Time  
0C = hours in hex

3B = minutes in hex

GT telling IKE to set the date:

3B 07 80 40 02 1B 05 08 cc

GT --> IKE : On-board computer set data: Set Date = 27/05/08

40 = OBC Set data

02 = Date

1B = day in hex

05 = month in hex

08 = year in hex

' 15h is an answer to 14h or is used to save new settings

' 80 07 BF 15 00 04 00 42 6B

' IKE --> GLO : Country coding status, Data="00 04 00 42"

' first byte = vehicle + language

' Only onboard network (not for vehicle identification!) for following vehicles:

' 0000 ---- E38/E39H (or E53H) since start of E38/E39H

' 0011 ---- E39B (or E53B,E52) E39B=PU98

' 0100 ---- E46 before PU98 (Softwarevers. 7)

' 0110 ---- E46 after PU98 (Softwarevers. 11 und 12)

' 0101 ---- blocked, not used

' 1010 ---- E83/E85

' 1011 ---- R50 from PU00

' 1111 ---- E46 after PU98 (from Softwarevers. 13)

' 0010 ---- E52 (from BG 2 B2)

' 0001 ---- E53 (not used, see E38, E39H, E39B)

' 1010 ---- free

' 0111 ---- blocked

' 1000 ---- RR01

' 1100 ---- R55

' 1110 ---- E65

' 1001 ---- free

' ---- 0000 Deutschland (A,CH usw) D 0 deutsch

' ---- 0001 engl. UK GB 1 English UK

' ---- 0010 engl. US USA 2 English US

' ---- 0011 Italien I 3 Italian

' ---- 0100 Spanien E 4 Spanish

' ---- 0101 engl. Japan J 5 English UK

' ---- 0110 French F 6 French

' ---- 0111 CDN CDN 7 English US

' ---- 1000 AUS/Golf/ZA AUS/GOLF 8 englisch UK

' Warning: Bits 0-3 in DB1 are not country codes, but are language codes

Thanks NAVCoder/Jochen

80 07 BF 15 F1 00 00 00 DC

IKE --> GLO : Country coding status: Ntwk=E46 Lang=GB\_English Clock=24h OutTemp=°C AvgSpd=km/h Limit=km Dist=km ArrTime=24h Consump=l/100\_km Motor=Petrol

### **Set units on IKE**

To change country coding, first tell the IKE what it should be  
The IKE then does the job of telling everyone else.

Here is an E46 that just changed km to miles:

3B 07 80 15 F1 70 10 00 38

GT --> IKE : Country coding status: Ntwk=E46 Lang=GB\_English Clock=24h OutTemp=°C

AvgSpd=mph Limit=mls Dist=mls ArrTime=12h Motor=Petrol

So it uses the same country coding status, but the GT sends the info to the IKE  
80 05 BF 15 01 00 cc

change OBC temp to C, 24H time, MPG

80 05 BF 15 01 01 cc	change OBC temp to C, 12H time, KM/L
80 05 BF 15 01 02 cc	change OBC temp to F, 24H time, L/100km
80 05 BF 15 01 03 cc	change OBC temp to F, 12H time, US MPG
80 05 BF 15 01 04 cc	change OBC temp to C, 24H time, MPG
80 05 BF 15 01 05 cc	change OBC temp to C, 12H time, KM/L
80 05 BF 15 01 06 cc	change OBC temp to F, 24H time, L/100KM
80 05 BF 15 01 07 cc	change OBC temp to F, 12H time, US MPG
80 05 BF 15 01 08 cc	change OBC temp to C, 24H time, MPG
80 05 BF 15 01 09 cc	change OBC temp to C, 12H time, KM/L

country coding (0x15) has 4 x databytes

A complete message would be 80 07 BF 15 00 04 00 42 6B

DB1

Upper nibble: vehicle network type

Lower nibble: language coding, decimal:

0=DE

1=GB

2=US

3=IT

4=ES

5=JP

6=FR

7=CDN

8=AUS/Golf/ZA

9=NL

10=RU

DB2:

bit 0 time and date, 0=24h, 1=12h

bit 1 temperature 0=°C, 1=°F

bit 4 avg speed 0=km/h, 1=mph

bit 5 limit 0=km, 1=miles

bit 6 distance 0=km, 1=miles

bit 7 arrival time 0=24h, 1=12h

DB3:

bits 0...3 control the units for the consumption values

0000 0000 l/100 km 0x00

0000 0101 mpg 0x05

0000 1111 km/l 0x0F

I think this is 2 x different unit values: one for Cons1 and one for Cons2

but nav applies same settings to Consumpt1 and Consumpt2

eg:

xxxx xx00 Consumpt.1 L/100

xxxx xx01 Consumpt.1 MPG

xxxx xx11 Consumpt.1 KM/L

xxxx 00xx Consumpt.2 L/100

xxxx 01xx Consumpt.2 MPG

xxxx 11xx Consumpt.2 KM/L

DB4:

bit 0 motor type 0=petrol, 1=Diesel

### **Lights (Thanks warfield)**

[ID:TURN\_LIGHTS\_OFF]

00 04 BF 76 00 cc

[ID:FLASH\_WARN]

00 04 bf 76 02 cc

[ID:FLASH\_LOW]

00 04 bf 76 04 cc

[ID:FLASH\_LOW\_WARN]

00 04 bf 76 06 cc

[ID:FLASH\_HI]

00 04 bf 76 08 cc

[ID:FLASH\_HI\_WARN]

00 04 bf 76 0A cc

[ID:FLASH\_LOW\_HI]

00 04 bf 76 0C

[ID:FLASH\_LOW\_HI\_WARN]

00 04 bf 76 0E cc

[ID:FLASH\_LOW\_SMALL]

80 04 BF 11 03 cc

[ID:FLASH\_TEST1]

00 04 bf 76 11 cc

Testsequence (Do not try! Will set IO status from diagnostic to entire car)

0="3F0BBF0C0000000021100006"

1="3F0BBF0C0000000000000006"

2="3F0BBF0C0000000021100006"

3="3F0BBF0C0000000000000006"

4="3F0BBF0C0000000058200006"

5="3F0BBF0C0000000000000006"

6="3F0BBF0C0000000058200006"

7="3F0BBF0C0000000000000006"

### **Window and door message 7A**

00 05 BF 7A 51 20 B1

3F 06 00 0C 01 31 01 04 driver side mirror fold in

3F 06 00 0C 01 30 01 05 driver side mirror fold out

3F 06 00 0C 02 31 01 07 passenger side mirror fold in

3F 06 00 0C 02 30 01 06 passenger side mirror fold out



# How to start up a 16:9 bordmonitor

## *Cut from hacktheibus messages*

Commands including crc

```
GT --> BMBT: RGB control: LCD_on TV
// on          3B 05 F0 4F 11 11 81
GT --> BMBT: RGB control: LCD_off TV
// off         3B 05 F0 4F 01 01 81
IKE --> GLO : Ignition status: Pos1_Acc Pos2_On Pos3_Start
// poll        80 04 BF 11 17 3D
```

Boot block sent from my Bordmonitor at boot

```
BMBT --> RAD : CD status: Mode0_Function_02 Pause CD 1, Track 0 CDs_Loaded:
P1
// To Radio    F0 0B 68 39 00 02 00 01 00 01 00 00 A8
BMBT --> RAD : CD status: Mode11_Function_02 Pause CD 0, Track 0
CDs_Loaded: P1
// To Radio    F0 0B 68 39 0B 02 10 01 00 00 00 00 B2
// To LCM      F0 04 BF 02 70 39
BMBT --> IKE : Ignition status request
// To IKE      F0 03 80 10 63
BMBT --> LKM : Light dimmer status request
// To NAV locat F0 03 D0 5D 7E
BMBT --> GM : Doors/flaps status request
// Broadcast   F0 03 00 79 8A
```

"I'm alive" message from Bordmonitor every 10'th second. Possibly a handshake with poll.

```
// To Radio    F0 03 68 01 9A

// Radio polls CD with    68 03 18 01 72
// Reply is              18 04 FF 02 00 E1
```

The Bordmonitor is shutdown when ignition off message is received.

```
// 44 05 BF 74 00 FF 75
```

After ignition is applied my BM starts to send: F0 BM 03 ...68 RADIO 01 'status request' CS

-> It is checking for the radio.

When I remove the radio from kbus the LED turns off. -> If the radio LED would be on the BM will switch it off when the radio doesn't answer to the message.-> The ignition message is not related to this BUT the 16:9 BM likes to get the ignition status message regularly. In my logs it looks like it even asks for it (ID 10h) when the status of the ignition line changes.