I-Bus Protocol

CREDITS: The I-Bus was originally reverse-engineered by **Thomas L. Wood** for the **MPorty/Cajun MP3 Project**. **This** was his original write-up of it. **Richard Naninck** performed additional research and wrote the first general-purpose I-Bus message decoder for intercepting Video Module messages. **Ed Anuff** identified the I-Bus as being based on the ISO 9141/K-Bus protocol and documented the semiconductor manufacturers who supply devices supporting it.

Since then, a Yahoo! Group has formed to discuss the technical details of the I-Bus, appropriately named **HackTheIBus**. Along with the **OpenBMW Yahoo! Group**, this represents a valuable forum for discussing issues related to I-Bus interfacing and members of both groups regularly post useful information.

IMPORTANT: This information is provided for reference purposes only. Use this information at your own risk.

BMW's I-Bus is based on ISO 9141 and K-Bus (The protocols used by OBD-II). It is basically a second K-Bus in the car for handling interfacing between the radio, CD, navigation, and telephone systems. In particular, the steering wheel controls for the radio, CD, and phone utilize the I-Bus.

The I-Bus is a single wire bus that is a white/red/yellow wire which is available at a number of different connectors inside the car such as at the CD changer connector in the rear, the phone connector in the center console, etc.

The bus' physical layer is an open collector setup pulled high (+12v) by the bus, and pulled low by the talker. This means that the idle voltage on the wire is +12v (the battery voltage, or Vbatt). This is somewhat similar to RS232 signalling, and like RS232, an interface IC is needed to turn this into a TTL digital signal that can be understood by a microcontroller. Later in this document, several off-the-shelf parts will be listed that can be used to easily drive this sort of signal.

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

An I-Bus packet contains the following bytes of data:

Source Address
Length
Destination Address
Data Bytes
XOR Checksum

Unlike most similar single wire network protocols, the I-Bus has no apparent way to identify the start of a packet, so this must accomplished in software by looking for a delay between messages.

The following device addresses have been decoded for the E39 5-series:

0x00	Broadcast
0x18	CDC CD-Player
0x3B	NAV Navigation/Videomodule
0x43	MenuScreen
0x50	MFL Multi Functional Steering Wheel Buttons
0x60	PDC Park Distance Control
0x68	RAD Radio
0x6A	DSP Digital Sound Processor
0x80	IKE Instrument Control Electronics
0xBF	LCM Light Control Module
0xC0	MID Multi-Information Display Buttons
0xC8	TEL Telephone
0xD0	Navigation Location
0xE7	OBC TextBar
0xED	Lights, Wipers, Seat Memory
0xF0	BMB Board Monitor Buttons
0xFF	Broadcast

A packet containing the following bytes would display "CD 7-04" on the Radio MID Display:

```
68 17 ff 23 c0 30 07 20 20 20 20 20 08 43 44 20 37 2d 30 34 20 20 20 20 45
S  L  D  Data                    'C''D'' ''7''-''0''4'            C
r  e  s                                                          h
c  n  t                                                          k
```

A full set of button codes is available **here**.

Below is an example of a track title displayed on an E39 MID Radio Display from an in-car MP3 player (Courtesy of **Thomas L. Wood**):
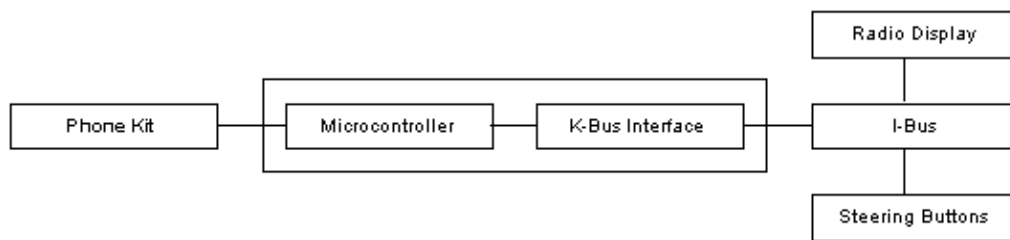
Interfacing

Typical projects as well as general experimentation call for some form of I-Bus to RS232 interface so that I-Bus packets can be captured and displayed on a laptop or other device in order to be decoded. Because the I-Bus uses the physical layer of the ISO9141/K-Bus, it is fairly easy to build a connector from off-the-shelf IC parts or to even potentially take an existing OBD-2/RS232 adapter and connect the K-Line from it to the I-Bus.

If building an adapter from scratch, utilizing an off-the-shelf K-Bus interface/transceivers should save a lot of effort. The advantage of a design using a K-Bus interface chip of some sort is that most of these chips deal with a number of issues such as bus contention and not overloading the bus that make them a safer choice. There are a number of good choices for this, which are listed with links to their manufacturers below. Another option is to attempt to use a tranceiver that is compatible with the **LIN (Local Interconnect Network)**, which is also based on ISO9141 but runs at a faster speed. Any of the listed interface/transceivers should couple very nicely with a Maxim MAX232, which handles can drive standard RS232 communications to a laptop or small embeddable computer such as the **Advantech PCM-5820**. While it is possible to interface to the I-Bus without a K-Bus tranceiver IC, BMW's own products, like the Video Module, use K-Bus chips from companies such as **ELMOS**. **Melexis** is a popular choice for obtaining these types of interfaces since they have a web store that sells single unit quantities. Many hobbyists, however, prefer to use interfaces built using transistor circuits and have had success in doing so.
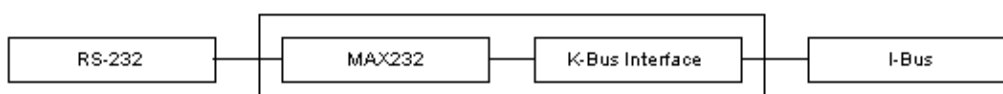
Depending on the application, for example building an adaptor to control a hands-free phone kit, connecting to a full PC would not make sense and instead some sort of low-cost microcontroller would be used. The CD-Changer adapters from companies like **BlitzSafe** and others are examples of this. Hopefully, this information will spur hobbyists to create an assortment of interface kits for controlling a number of different in-car electronics.

Useful Parts For Building Adapters

K-Bus/I-Bus Interface/Transceivers

Melexis TH8080 LIN/K-Bus Transceiver 📄

Melexis TH3122 K-Bus Transceiver with integrated Voltage Regulator 📄

ELMOS K-Bus transceiver 📄

Motorola MC33290 K-Line Serial Link Interface 📄

Motorola MC33199 Automotive ISO 9141 Serial Link Driver 📄

Atmel U6812B Single-Ended Bus Transceiver 📄

ON Semi NCV7310 Single Wire LIN Transceiver 📄

Philips TJA1020 LIN transceiver 📄

Infineon TLE 6258 Single-Wire-Transceiver 📄

RS-232 Driver/Receivers

Maxim MAX232A +5V Powered, Multichannel RS-232 Driver/Receiver 📄