

1.0 AIM CAN Protocol for ECU Communications (Type 1)

Die trijekt Steuerung sendet die Botschaften mit dem Identifier 0x770.
Die Übertragungsgeschwindigkeit ist einstellbar auf 500Kbits/s oder 1MBit/s.

CAN Asynchronous Messaging

AIM technique is referred to as Asynchronous messaging, basically the whole stream of parameters (all 35 Bytes) is split up into the packets 8 bytes in length, these are sequentially inserted into CAN messages and in a given order. The packets of data do not contain a specific identifier, they are just in a predefined order. At the receiving node the device looks for the Header information (this is a constant contained in the datastream), when this is seen the device knows that next message is the start of the datastream and all subsequent CAN messages will contain the given parameters in the predefined order.

In this way the CAN system is simply a carrier for seemingly highly variable data under a single base addresses and the software handlers at either end know how to breakdown and reassemble these separate packets of data into a continuous and complete datastream.

Byte	Signal	Units	Scaling
0:1	RPM	RPM	1RPM
2:3	reserved		
4:5	Reserved		
6:7	Reserved		
8:9	Water Temperature	Deg C	0.1Deg C
10:11	Reserved		
12:13	Battery Voltage	Volts	0.01Volts Calculation: CAN-Value * 5 / 1024
14:15	Throttle Angle	%	0.1%
16:17	Manifold Pressure	mBar	1mBar
18:19	Air Charge Temperature	Deg C	0.1Deg C
20:21	Reserved		
22:23	Lambda	Lambda	0.001 La
24:25	Fuel Temperature	Deg C	0.1Deg C
26:27	Reserved		
28:29	Errors	ECU-specific error flags Bit 0: lambda sensor ok Bit 1: engine temp. sensor ok Bit 2: air temp. sensor ok Bit 3: int. air press. sensor ok Bit 4: throttle pot. ok Bit 5: ext. air press. sensor ok Bit 6: airmass sensor ok	
30	Number of Data Bytes	30	
31	Marker Byte 1	FC	
32	Marker Byte 2	FB	
33	Marker Byte 3	FA	
34	Checksum		

Checksum is the sum of all bytes of the structure up to and including marker byte 3.

2.0 AIM CAN Protocol for ECU Communications (Type 2)

Der Typ 2 ist identisch mit Typ1, jedoch sind die Identifier der einzelnen Botschaften wie folgt:

Bytes	Identifier
0..7	0x0770
8..15	0x0771
16..23	0x0772
24..31	0x0773
32..34	0x0774