

CAN Module – MPC2C

CAN > CAN interface



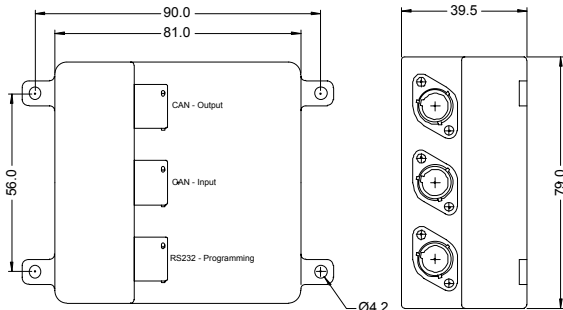
Description

This user programmable CAN interface has been designed to link multiple different CAN systems to a common high speed CAN network within the logger network. Each MPC2C device is manufacturer programmed to suit the CAN speeds of all common vehicle networks.

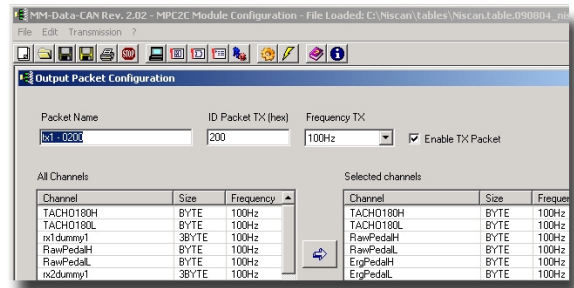
All loggers utilise the CAN2-A format with a standardised internal structure to enable simple connection to all of our own system modules and devices. For this reason, nearly all external CAN streams (regardless of speed) are fed through an MPC2C device before entering the logger.



Dimensions



MM Data CAN software



Technical data

Electrical / Mechanical

Supply..... 10 > 18 V dc
 Consumption (at no tx)..... < 50 mA
 Temperature..... -40 > + 85 °C
 Over voltage 50 V
 Shock 50 g
 @..... 10 ms
 Weight 180 g
 Container IP67

Inputs

RS232 Programming
 Format..... 38400 8N1
 CAN 250 / 500 Kbps or 1 Mbps
 Termination (default) Closed

CAN – Output

Type 2.0 A
 Speed..... 1Mbps
 Range 0x200 > 0x2FC
 Termination Open

Electrical connections

CAN out	AS0 06 05 SA HE
1	CAN P
2	CAN N
3	GND
4	not used
5	12v

CAN in	AS0 06 05 SN HE
1	CAN P
2	CAN N
3	GND
4	not used
5	not used

RS232	AS0 06 05 SD HE
1	TX
2	RX
3	GND
4	not used
5	not used

Ordering information

Part No.	Description	Order Code
	CAN to CAN interface	MPC2C

CAN input speed to be defined at the time of ordering – 128 / 250 / 500 kBsec

For further details please contact	
MM Competition Systems Ltd Hyjuniper, Long Lane Shaw, Berkshire, RG14 2TA England	Tel: +44 (0)8707 444666 Fax: +44 (0)8707 444888 www.mmcompsys.com mail@mmcompsys.com

CAN interface – Overview.

The CAN lines used within the MM equipment are primarily used as expansion routes to enable system components to communicate internally. To enable trouble free operation of these components, the CAN architecture is very strictly controlled to the following rules.

- o CAN2-A
- o 1Mbps (Mega Bit per Second)
- o All Word or larger channels are aligned with an 'even' offset with the packet.
- o A maximum of 64 packets per CAN line using ID ranges from 0x200 to 0x02FC (by default)
- o All Word or larger channels are internally processed considering the LSB (Least Significant Byte) as the last byte of the channel. (Little Endian)

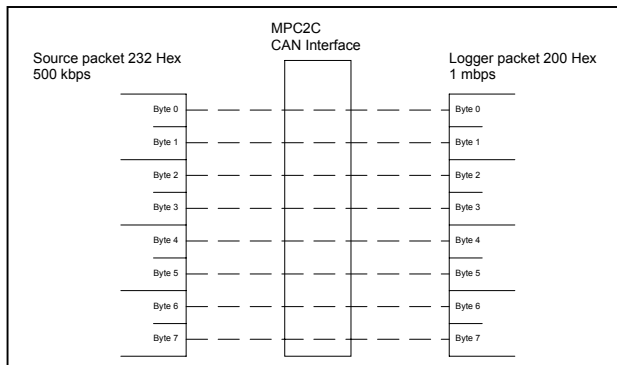
This CAN structure may not be usable as a direct interface point of many CAN systems for one or more of the following reasons.

- o CAN speed may be wrong
- o Words, or Longs may be aligned with an uneven packet offset
- o The LSB may be the first byte of a channel
- o Packet ID may be outside our normal range.
- o The vehicle may have multiple CAN lines and differing speeds

The MPC2C CAN interface module and the M-CAN software have been developed to enable any CAN system to be integrated. The following 2 diagrams are of simple conversions. This process is continued for all input packets, with every type of input packet construction being supported by the M-CAN software.

In this simple example a CAN packet needs to be converted for the following reasons

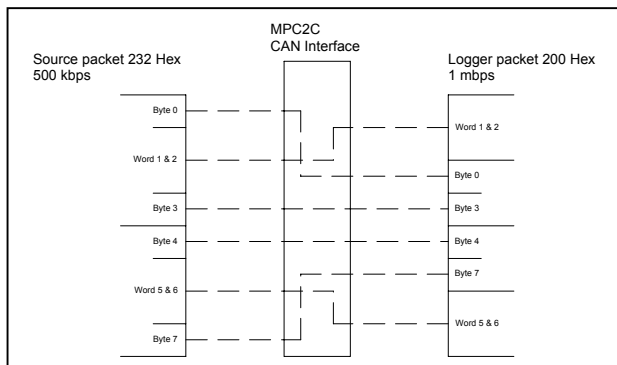
- The speed is different
- The ID is out of normal range



The 8 individual bytes are simply moved directly into a new 1Mbps CAN packet with an ID that is acceptable. This is a very simple conversion.

In this second example a CAN packet needs to be converted for the following reasons

- The speed is different
- The ID is out of normal range
- There are 2 Words with un-even packet offset alignment



The mixture of Bytes and Words are manipulated by the MPC2C to create a new Can packet that is acceptable. The output positions chosen are simply examples; the user is free to build the output packet in any acceptable way.

Notes:

Multiple MPC2C devices may be connected to the same output CAN line providing that the 'output ID' is chosen to avoid other devices.

The MPC2C is programmed separately from the other system components using a dedicated RS232 line.

The entire conversion table may be printed out to give the user an easy view of the path each channel has taken through the module.