

MIAAC

System CAN customisation



MIAC System CAN Customisation

The default configuration of a MIAC system uses the following (Standard Frame 11 bit) CAN IDs:

| | |
|---------------------------|-------------|
| Basic module | 0x010 (16) |
| Advanced module | 0x020 (32) |
| Serial module | 0x030 (48) |
| Industrial module | 0x040 (64) |
| Bluetooth module | 0x050 (80) |
| GPS module | 0x060 (96) |
| GSM module | 0x080 (128) |
| ZigBee coordinator module | 0x090 (144) |
| ZigBee router module | 0x0a0 (160) |
| MIAC slave | 0x0c0 (192) |
| MIAC master | 0x0f0 (240) |

These are the "Ordinal 0" addresses. Up to four of each module type can be used in a MIAC system. These are designated Ordinal values 0 to 3. The Ordinal value is defined by CAN address bits <10:9>. Hence a Basic Module set to ordinal 1 would have a CAN address of 0x010 + 0x200 (528).

All modules send their replies to the MIAC Master at CAN ID 0x0f0 (240). The MIAC System normally operates at a bit rate of 500kbs.

In order to use a MIAC System on an existing CAN network the module CAN IDs and system bit rate can be reprogrammed.

Configuring the MIAC System Modules

Firstly the hardware needs to be programmed to the new configuration. This will need to be done for each expansion modules that is to be used within the system.

Determine the bit rate for the system and the CAN address to be used by the expansion module. Use this information to create a MIAC System Flowcode program to initially setup the module.

There is a helper function that can be called from an "Insert C code" macro:

MIAC_Module_Custom_Settings (OLD_CAN_ID, NEW_CAN_ID, NEW_REPLY_ID, NEW_RATE)

As an example, if you have a CAN network that runs at 125kbs and have CAN ID addresses free between 100 and 132, you could allocate ID 100 as the MIAC Master reply address, and say use 101 as the CAN ID for an Advanced Module. To reprogram the Advanced Module with this information connect it, via a short CAN network, to a MIAC and create and download a standard MIAC System Flowcode flowchart that runs the flowing C code:

MIAC_Module_Custom_Settings(32, 101, 100, 125);

This code will send a reprogram command to the Advanced Module and change its bit rate to 125kbs, its CAN ID to 101 and its reply address to 100. The changes will take effect on the next, and subsequent, power-up. Settings are persisted in the internal EEPROM of the expansion module.

The expansion module is now ready to be installed on the existing CAN network.

Nb. A module can be re-reset to its factory default setting by either:

a) Calling: MIAC_Module_Factory_Settings(CURRENT_CAN_ID)

b) Opening the module and changing the Ordinal setting links to anything other than the zero. (Power-up the module with links set, then power-off and then set back to ordinal zero if ordinal zero is required)

Configuring the MIAC System Flowcode

Create your MIAC System Flowcode program in the usual way as outlined in the document MIAC Operation and Programming Guide (available on the Matrix website), but in order to use customised CAN system settings there are a few simple extra steps required.

a) Select MIAC(0) properties, Component Custom code, and edit [Defines]. You will see some commented out code. Un-comment the two lines, such that they appear as follows:

```
#define MX_MASTER_CAN_ID (100)    // Master MIAC CAN ID of 100, as example above
#define MX_MASTER_BIT_RATE (125) // re-configured bit rate of 125kbs
```

Click OK to save the changes.

b) For each MIAC System Module, e.g. MIAC_Advanced(0), select properties, Component Custom code, and edit [Defines]. Un-comment, or create a new line, as follows:

```
#define %dID (101)                // e.g. to set this Module CAN ID to 101
```

Click OK to save the changes.

The program is now configured to the new settings and can be compiled and downloaded to the Master MIAC in the usual way. The Master MIAC can now be installed into the host CAN network.



Matrix Multimedia Ltd.
23 Emscote Street South
Halifax
HX1 3AN

t: +44 (0)1422 252380
e: sales@matrixmultimedia.co.uk

www.matrixmultimedia.com