ACE8C_2 CBUS 8 input combination module. 12V version.

Updated 11/02/11 Now rev s

Introduction

ACE8C_2 is one of a number of layout control modules for use with the CBUS system. This is a general purpose layout control bus (LCB) using the industry standard CAN bus. For more information on CBUS, see the introductory article on the MERG website. (www.cbus.org.uk)

Please refer to the schematic ACE8C_2_sch.pdf.

ACE8C_2 is primarily a ‘producer’ module which sends events over the CBUS to consumer modules. It has 8 inputs for switches or logic levels and a change in any of these sends an ON or OFF event corresponding to which switch was changed. However, it can also learn input events so is also a CBUS ‘consumer’. Such ‘producer / consumer’ modules have been called ‘combi’ modules, hence the ‘C’ in the title.

On receipt of a learned input event, ACE8C_2 sends an output event which contains the logic levels (on or off) of the 8 inputs as the least significant byte (LSByte) of the event number (EN). This allows other producers, like control panels or a PC, to trigger an event from the ACE8C_2 which reflects the switch inputs in the event. This allows the creation of conditional events depending on the input states which may be from block occupancy detectors so allowing different routes to be set depending on the occupancy of the various tracks. Another use of this facility would be for interlocking of tracks or complete routes. A further use could be the setting of routes or events with a rotary switch or switches so any of 256 routes could be selected on the switches and then triggered by another producer, say a button on a control panel. To distinguish a triggered event from just a switch change event, a bit is set in the next byte of the EN.

The triggered output aspect of this module is still being developed. By using the ‘Sel’ switches during a learn sequence, there can be 16 possible response types to each input event. Response type 0 is a sequence of 8 events corresponding to the input states. This allows a layout state to be reflected on a control panel as if the states had actually changed e.g. for a Start of Day event. . Response type 1 allows for rotary switches and logic states to create unique events for route setting etc. Further modes have now been added to Rev s (and later) to simplify operation of a layout with a computer. See later in this document.

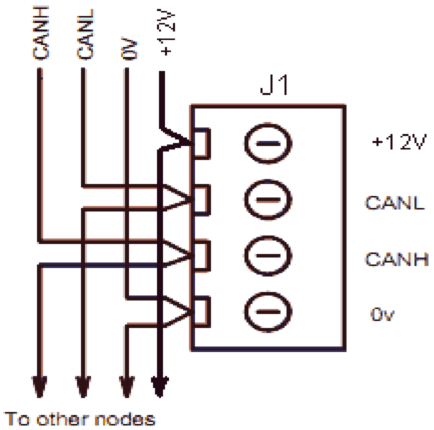
ACE8C_2 follows the SLiM (Small Layout interface Model) of CBUS which allows it to be given a Node Number (NN) by onboard jumpers. No programming is necessary. For practical reasons, the present ACE8C_2 module allows 64 node numbers to be set. However, like all other SLiM modules, it responds with its CAN-ID when interrogated by ‘nodes’ which implement the self-enumeration scheme so is compatible with the Full Model modules (FLiM). For SLiM modules, the CAN_ID is set using the Sel switches to the same value as the Node Number (NN). The firmware allows operation in either SLiM or FLiM modes and also includes a ‘bootloader’ which allows code updates via the CBUS itself. A PC based free software ‘tool’ is available for this.

Power supply.

ACE8C_2 requires an external 12 volt DC supply. This is wired to the CBUS connector pins 1 (0V) and 4 (12V). The nominal current for this module is 30mA.

The green LED (LD2) will illuminate to show the circuit is working correctly. (SLiM mode) This is not just a power on indicator but confirms correct working of the processor.

Connecting the module



The CANH and CANL wires go to all modules. They are polarity sensitive so CANH must go to CANH and CANL to CANL. These wires should ideally be a twisted pair although screening is not necessary. While it would be usual to wire the bus sequentially round the various modules, it is not essential and individual nodes can be 'star' connected if this is more convenient. The CBUS wiring should be kept separated from any DCC supply wiring to prevent possible interference.

Where a number of modules are powered off the same 12V DC supply, then their 12V input terminals should all be connected together. If separate 12V supplies are used, the 0V should be commoned.

The CAN bus requires 'termination' resistors at some point in the network. If the bus is wired sequentially round the modules, then a resistor of 120 ohms should be fitted across the bus at each end. For small layouts, it is sufficient to have resistor across the bus at one point. The value is not critical and a 68 ohm resistor will suffice.

Setting the node number.

Purely due to hardware restraints, the ACE8C_2 module allows 16 node numbers selected as a binary sequence with the DIL switch DIP 1 positions Sel 0, Sel 1, Sel 2 and Sel 3. The actual binary numbers are from 0 (all switches on) to 15 (all switches off) but the CBUS convention does not allow a node number of 0 so the actual range is 1 to 16.

A further range of node numbers is possible by using the jumpers S4 and S5. These extend the range of node numbers up to 64. The jumper S6 is used to select "push button" mode (jumper in) or toggle mode (jumper out). This applies only to SLiM operation. In the Full mode, each input can be individually selected to be ON / OFF or ON only.

Note that if a layout contains any ACE3_2 control panel modules in SLiM mode, these may occupy the first 4 node numbers so other producers like ACE8C_2 should start at 5 or above.

In Full mode, any node number can be allocated up to the maximum of 65534.

Teaching the trigger event. (SLiM mode)

The ACE8C_2 module can respond to up to 32 learned input events. If you try to set more than 32, no more will be added but the yellow LED (LD1) will flash to indicate the event stack is full.

To teach the module an event, put the 'learn' switch ON and then send the event to be learned. This can be an ON, OFF or request event. The ACE8C_2 will respond with an event sequence determined by the setting of the 4 Sel switches while in learn mode.

With the switches set to 0000, (mode 0) when triggered, the node produces 8 successive ON or OFF events corresponding to the input states. This allows a triggered sequence to set states as if the input had changed. There is a small delay between each event so the CAN bus is not fully loaded.

With the switches set to 0001, (mode 1) the response is an ON event where the LSByte of the event is the 8 inputs. Additionally, to distinguish this from input changes, bit 0 of the next higher event byte is set. There are 256 possible events.

After the learn process, turn off the learn switch and reset the Sel switches to the Node Number. The response is NOT triggered during the learn phase. To 'unlearn' the event, repeat the above but with both the learn and unlearn switches ON. All learned events can be cleared by putting the unlearn switch on and then cycling the 12V supply off and then on. Remember to turn the unlearn switch off after this procedure.

The inputs all have pullup resistors of 100K so can be connected to ON / OFF switches directly. Also they will accept logic level inputs of 5V and also may be connected to the transistor outputs of opto-isolators. Additional collector loads may be connected to the +5V supply if needed. The inputs are protected for voltages up to 12V. The 5V supply may also be used to power external circuits or devices with the proviso that the current does not exceed the capability of the device supplying the ACE8C_2 module or 1 amp which ever is the smaller.

While in SLiM mode, the mode 1 event (where the LSbyte is the 8 inputs) can be triggered by a brief press of the small PB on the PCB. This is useful for testing the action of this event without needing a trigger event externally. If the module is used for route setting with binary or BCD switches on the 8 inputs, an external PB can be substituted to 'fire' the route once the switches have been set. BCD switches are recommended to give a route selection from 0 to 99. If fewer inputs are connected to the switch, the remaining inputs may be used for interlocking or conditional route setting. Note that as you rotate the input switches, you will get 'events' as each input changes. This will be of no significance if no 'consumer' has been taught to respond to these events.

Additional modes from rev s.

Although CBUS was initially intended to use the 'Producer / Consumer' model where the ACE8C produces unique events for each switch change which are transparent to the user, this is not always advantageous if PC layout operation is desired. It is now possible to allocate known numbers to each of the 8 inputs, in either SLiM or FLiM modes but it does require a means of teaching these over the bus. (The Config. Tool, JMRI 'console' etc.)

Allocating 'device numbers' in SLiM mode.

Set the learn switch ON and switch Sel 3 to OFF (logic 1) on the DIL switch. Use switches Sel 0 to Sel 2 to select one of 8 inputs. 000 is input 1 (0) and 111 is input 8 (7). Send a 'short' event over the bus with the two LSbytes containing the desired input 'device' number and the upper two bytes of the event set to 0. Using Sel 3 gives a 'mode' range of 8 to 15, one for each input. If a number has been allocated in modes 8 to 15, any change on the inputs will be sent as a short event using the preset device number in the two LSbytes.

This does not affect the values sent in modes 0 and 1.

Using the device numbers allows a PC to know which device sent the input change and also allows the PC to 'poll' each input by number for its current state.

Remember to take the module out of learn mode and reset the DIL switches to the SLiM node number after this teaching process.

If the module is in FLiM mode, the input device numbers are allocated over the CBUS using the Config. Tool. The format of the single EV (event variable) corresponds to that on the DIL switches with bit 3 set to a 1. Also, in FLiM mode, there is a Node variable (NV) of one byte where the inputs can be set to ON / OFF or ON only on a bit by bit basis. A bit set to 0 is ON / OFF, the bit set to a 1 is ON only. This is independent of the jumper S6. These bits correspond to inputs 1 to 8.

Limitations.

The presently available CAN transceivers set a maximum number of CAN nodes on any one ‘segment’ to 110. There is no limit to the number of consumer modules so care must be taken not to overload the CAN bus. The Full Model (FLiM) scheme allows for 65536 modules which will be programmable over the CBUS itself. Here, we intend to develop modules to bridge between many CAN segments (CAN-CANs).

Notes.

The PCBs include provision both for in-circuit serial programming and debugging (ICSP) and two LEDs driven off the programming pins. Low current LEDs are recommended. The green LED indicates that the module is working in SLiM mode. It is set following the power up sequence of the processor and is not merely a power on indicator. The yellow LED will flash if the event stack is full.

The small pushbutton S1 is intended for putting the module into the Full (FLiM) mode. To do this, hold the button in for about 8 seconds. The green LED will extinguish. Release the button and the yellow LED will flash. The module now needs a node number allocating over the bus. The use of the PC based ‘Config. Tool’ is recommended for setting up modules in FLiM mode. Once in FLiM mode, the yellow LED will show steady.

The full schematic, a PCB layout which is in .PDF form and can be printed to the exact size for making masks and the PIC assembly and HEX code are available on the MERG website. These can be freely used for non-commercial purposes. Copyright to the designs is held by the authors.