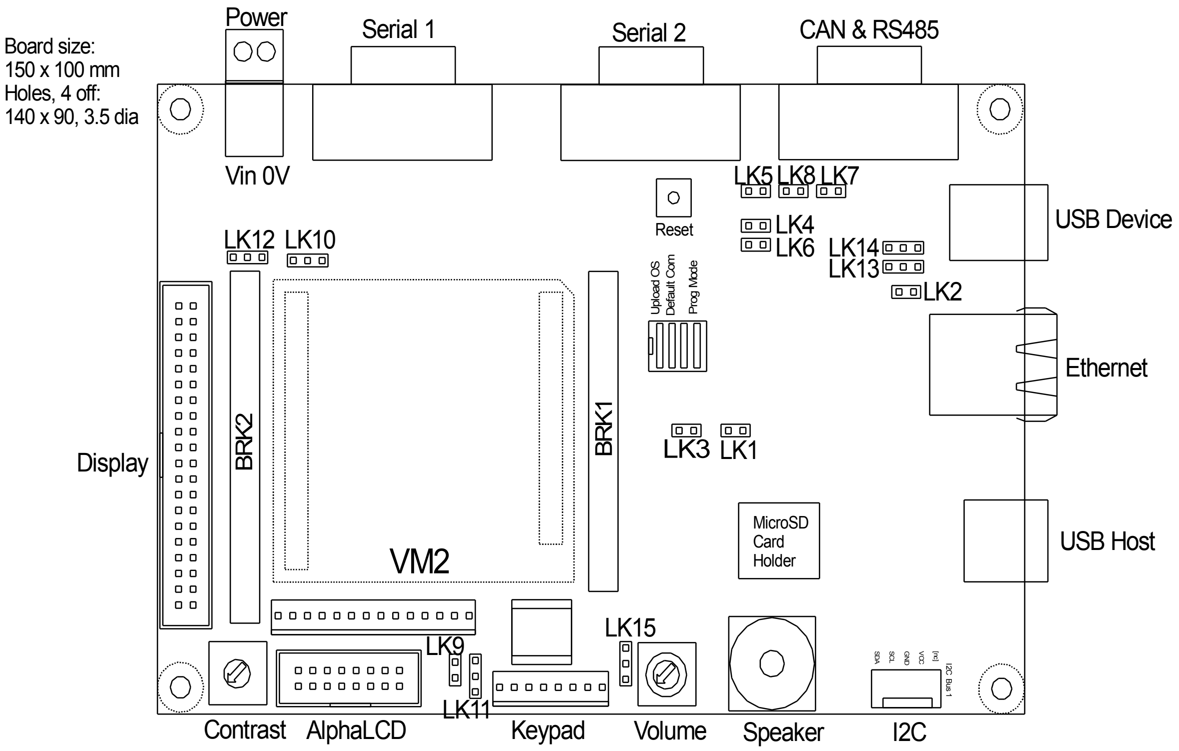
MICRO ROBOTICS

VM2 Application Board 3

Product code 5922



Application board 3 for the VM2 series of controllers is intended to implement as many interfaces as possible to allow you to explore the capabilities of the VM2 controller range.

- Graphics Display interface
- USB Host: removable storage using memory sticks, etc.
- USB Device: access to internal Flash Filing System
- Ethernet
- Memory card socket (microSD)
- 2 off RS232 serial ports
- 1 off RS485 Serial Port (half-duplex)
- I2C Bus (3.3V or 5V)
- CAN Bus
- Alphanumeric LCD interface (3.3V or 5V)
- Audio output (PWM 'beeps' or WAV sounds)
- EEPROM for non-volatile application settings or parameters
- 4 x 4 Matrix Keypad interface, or 8 General Purpose Digital I/Os
- Power inlet 12VDC (you can use 7.5 – 28 VDC with caution)
- Every VM2 pin appears on a 0.1” breakout connector.

Some of the VM2 series can't drive some interfaces; see the table of Controller capabilities on page 11.

WARNING: Users of Micro-Robotics Control Equipment should be aware of the possibility of a system failure, and must consider the implications of such failure. Micro-Robotics Ltd. can accept no responsibility for loss, injury, or damage resulting from the failure of our equipment. Use of our products in applications where their failure to perform as specified could result in injury or death is expressly forbidden.

The Old Maltings
135 Ditton Walk
Cambridge
CB5 8QB

tel: +44 (0) 1223 523100
fax: +44 (0) 1223 524242
sales@microrobotics.co.uk
www.microrobotics.co.uk

Unpacking

You should have:

- Application Board 3

What you will need

To start using the application board you will also need:

- VM2 or VM2D Controller (5900 or 5907)
- 7.5 - 12VDC, 4 Watt power supply. (You can use more than 12VDC with caution.)
- An RS232 serial lead to connect the 5902 to your host computer (you can use our product 5529)
- A PC running our *VenomIDE* development environment.

Switches

Reset button

Pressing this button shorts the VM2's Reset input to GND, which resets the VM2.

DIP switches

There is a group of four DIP switches and a single push-button switch fitted to the Application Board. These control basic modes of operation on the VM2. They are connected to mode selection pins on the VM2's JP1. Most of the time you are only likely to use the **Prog Mode** switch; all other DIP switches should be OFF.

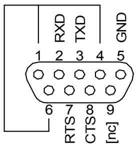
Switch number	**Label**	**Description**
4	**Prog Mode**	**ON**: The VM2 starts up in Program Mode. This gives you access to the Venom2 Command Line, so you can develop your application program. **OFF**: The VM2 starts up in Run Mode. It will run your *Venom2* application program at every Power On, or Reset.
3		*Not used*
2	*CAN(notUSB)*	This switch should be left in the OFF position unless you want to use the CAN Bus. If you use the CAN Bus then you can't use the VM2 as a USB Device.
1	Load OS	This switch puts the VM2 into *Upload OS* mode by shorting JP1 pin 4 to Vcc. This allows you to program the VM2 with a new version of the Venom2 Language using VenomIDE. See *IDE Help* in VenomIDE for details. (You can also use the USB port to load a new version of Venom – see instructions supplied with each version).

Serial Port 1 and 2: RS232

Serial ports 1 and 2 are fitted with level shifters to interface to RS232 signals, such as the COM ports on a PC.

Pin outs

Connectors: 9 way D-Type, pins



9 way D plug: View of plug

Serial 1

Pin	1	2	3	4	5	6	7	8	9
Channel		$1A	$19				$72	$73	
Signal	*	RXD	TXD	*	GND	*	RTS	CTS	N/C

Serial 2

Pin	1	2	3	4	5	6	7	8	9
Channel		$13	$12				$7B	$43	
Signal	*	RXD	TXD	*	GND	*	RTS	CTS	N/C

* - These pins are shorted together at each port.
There is a resistor position (R5) to pull SerialPort 1's CTS input high if you want the VM2 to be able to 'talk' when there is no serial lead supplying a hardware handshake signal to the Application Board. However it is usually better to overcome this problem in software by turning off hardware handshaking.

Configuration

Serial port 1 is configured for you in the default startup procedure with something like the following line.

```
  MAKE serial SerialPort(115200,1,1)
```

Serial port 4: RS485

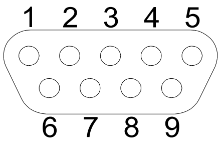
Serial port 4 is fitted with an RS485 driver in half-duplex mode. LK3 must be fitted to use this driver – it connects the driver to RX4 on the VM2.
You can’t use hardware handshaking with this hardware configuration. Instead the RTS signal for serial 4 (Channel $76) is used to control the *data direction* of the half-duplex interface. There is software support for this mode.

Termination and Pull up/down

You may also need to fit LK4 if this board is at the end of a transmission line – to terminate the line correctly, and also you may need to fit LK5 and LK6 if the line needs pulling into the passive state – this should be done only once on the whole transmission line.

Pin out

Note: the Serial 4 RS485 signals appear on the CAN connector:
Connector: 9 way D-Type, pins



Pin	1	2	3	4	5	6	7	8	9
Signal	Reserved (n/c)	CAN L	GND	**RS485 D-**	GND via LK8	GND	CANH	**RS485 D+**	Reserved (n/c)

Ethernet

Links: Fit LK1
Resources used: channel $34, SPI Bus 2, SPI ADDR.

The board has a 10-Base-T Ethernet interface.
To use this interface you must fit link LK1. This uses channel $34 (also used as an ADC input) for Ethernet interrupts.

There are network Link and Activity LEDs located on the socket itself.

Example code

See the *Venom2 Help File*.

Pin out

Standard RJ45 Ethernet pin out.

There is an EEPROM memory device associated with the Ethernet interface. This has been factory programmed with a unique MAC address that is essential for Ethernet communications. If you lose this address by over-writing it then we may charge an administration fee to replace it with a new one.

Memory Card Socket

Links: NONE
Resources used: SPI Bus 2 (4 chans), SPI ADDR (2 chans), Channel $35.

The board has a socket for microSD or microSDHC cards. The memory card is intended to provide permanent storage for the system, and should not be removed or fitted while the system is running unless you make provision for this in your application code.

Note that, on this board, power to the memory card is controlled by Channel $35. This will need to be turned on (active High) before the card can be used.
Warning: If you put a memory card in the socket and don't power it, it will interfere with the operation of the Ethernet and USB Host interfaces because it they all share SPI Bus 2.

Example code

```
   New Digital($35, %11).On ; Turn on card power.
   Make mcfs FileSystem("FSD")
```

Keypad or Digital I/O

Links: NONE
VM2 Resources used: I2CBus 1, Address 126

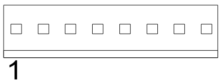
This 8-pin connector allows you to easily connect a 4 x 4 Matrix keypad. The eight pins should be connected to the rows and columns of a matrix keypad.
The interface is implemented using a PCF8574A device.
To configure it as a Keypad use the following code:

```
   MAKE kp Keypad (0, 248) ; 4x4 keypad
```

To configure it as Digital I/O you can use code like the following:

```
   Make d1 Digital(248)
   Make d2 Digital(249)
```

Keypad/Digital I/O Pint out



Pin	VM2 Channel Number (when used for Digital)	Keypad function
1	248	Column 1
2	249	Column 2
3	250	Column 3
4	251	Column 4
5	252	Row 1
6	253	Row 2
7	254	Row 3
8	255	Row 4

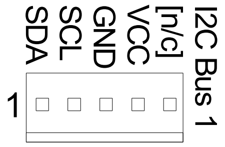
I^{2}C Bus

Links: LK10. Link 1-2 for 3.3V or 2-3 for 5.0V I2C Bus voltage.
Resources used: I2CBus 1

I2C Bus number 1 is brought out to a 5-pin connector. The pin out is printed on the board. You can select the Bus voltage using LK10. Be careful if you select 5V to make sure that every device you add to the Bus can handle 5V.

Pull up resistors

I2C Buses require pull up resistors on the bus signals. We have fitted 4K7 resistors which should be adequate for most purposes. If you need more pull up then you can fit resistors externally via the I2C Connector.



Configuration

This is done for you in the default startup procedure, so you don' t have to put it in your code:

```
   MAKE net I2CBus
```

This command shows what's connected to the I2C Bus:

```
   PRINT net ; print out a list of all devices connected to the I2C Bus.
```

EEPROMs

Links: LK10. Link 1-2 for 3.3V or 2-3 for 5.0V I2C Bus voltage.
Resources used: I2CBus 1

There are two EEPROMs fitted to the board. They are both ST24C02AM1 or equivalent, holding 256 bytes each.
The first EEPROM is at address 162. This device is intended for storing non-volatile settings and parameters for your application.

Configuration

```
   MAKE s SafeData(1, 1, 162)
```

The second EEPROM (at address 164) is intended for dedicated use by the Ethernet system. You may also use it for other purposes at your own risk – if you over-write the Ethernet MAC address then you may have to pay a small administration fee for us to issue you with a new one.

USB Host: External USB Drives

Resources used: SPI Bus 2, SPI ADDR.
You can plug a USB Mass Storage Device (such as a USB Flash Drive) into the USB Host socket. The VM2 can use this is either permanent or removable storage. Some memory sticks extend only a few millimetres beyond the USB socket and so are more suitable for permanent storage.

Configuration

```
   Make usbfs FileSystem(“USB”)
```

USB Device: VM2's Flash Filing System

Links: LK2: link. LK13 and LK14 – link pins 2-3 on both
Resources used: *CAN(notUSB)* switch set to OFF (enables USB device subsystem, disables CAN Bus)

The USB Device and CAN Bus cannot be used at the same time.

The VM2 can be a USB Drive using the Type B ('Device') USB connector on this application board. This allows you to read and write files in the VM2's Flash Filing System via USB, and also allows 'production programming' of VM2s with their finished application code.

To enable the USB system you have to do these things:

1. Set the ***CAN(notUSB)*** switch to OFF
2. Make link LK2
3. Links LK13 and 14: link pins 2 & 3 on both (Pin 1 is marked with a dot)
4. Connect the Application Board to a USB port on a PC or other Host.

When the VM2 is waiting at the **Clear RAM prompt**, or when USB access is enabled by your Venom code, the USB Host (usually your PC) will see the VM2 as a Mass Storage Device, allowing you to read and write files in the VM2's flash memory.

CAN Bus

Links: LK13 and LK14 – link pins 1-2 on both. LK 7: Bus Termination
Resources used: *CAN(notUSB)* switch set to ON (disables USB subsystem, enables CAN Bus)

The USB Device and CAN Bus cannot be used at the same time.

This board is fitted with a SN65HVD230 CAN Bus driver device allowing the VM2's CAN module to interface with CAN Bus systems.

Configuration

```
  MAKE can CANBus(1000000) ; CAN bus at 1MHz, not extended IDs
  MAKE can CANBus(125000, 1) ; CAN bus at 125KHz, extended IDs
```

CAN Waveform edge speed

This device uses a resistor to set the speed of edges on the CAN Bus waveforms. A 10K resistor is fitted, but you can change this by clipping it off, or soldering another resistor in parallel. Note: this will void the warranty.

CAN Bus Pull Up/Down and Termination

No CAN Bus pull up/down resistors are fitted to this board. They will need to be fitted externally if needed. All transmission lines should be terminated with the correct impedance at each end of the line. This is made easier on this board: LK 7 links in a termination resistor.
Only fit LK7 if the board is at one or other ends of the CAN Bus transmission line.

CAN Bus Pin Out

Connector: 9 way D-Type plug

Pin	1	2	3	4	5	6	7	8	9
Signal	Reserved (n/c)	CAN L	GND	RS485 D-	GND via LK8	GND	CANH	RS485 D+	Reserved (n/c)

Alphanumeric LCD

Links: LK11 – link pins 1-2 for 3.3V LCDs or pins 2-3 for 5V LCDs. LK9 connects the back-light to LCD_VCC.
Resources used: VM2 Channels $60-$65.
Note: VM2D can't drive an Alphanumeric LCD on this connector. You will either need a VM2 or VM2L, or you can use a VM2D to drive the LCD via the I2C Bus.

There are two connectors on board for direct connection of Hitachi and Hitachi-compatible alphanumeric LCD's. One is SIL and the other a *reverse* DIL connector. The reverse DIL connector allows a transition connector to be used on the *back* of LCD's with DIL connections.

Note that only one Alphanumeric LCD may be attached directly to the board, using either one or the other connector. Additional Alphanumeric LCDs may be connected via PCF8574 ICs on the I2C Bus.

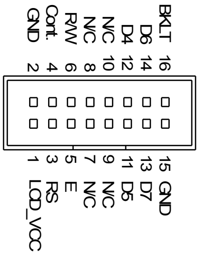
The LCD contrast voltage is set using the trim pot near the connectors. Most LCDs seem to operate with the contrast voltage at or near 0V.

Configuration

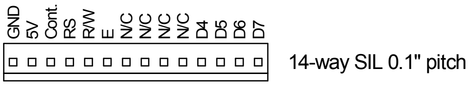
```
  MAKE lcd AlphaLCD (40,2,0) ; 40 x 2 LCD
```

DIL Pinout

Note: Pins 15 and 16 are the back-light power pins. BKLT, the +ve back-light supply is connected to LCD_VCC via LK9.



SIL Pinout



Graphic Displays

Graphics Displays of various types may be connected to this board. Some display devices have their own controller IC built in to the display, and others need an external controller.

The standard VM2 can generally drive displays with built-in display controllers that use *indirect addressing*. For TFT displays that need an external controller you can use the VM2D, which has a TFT controller built in to it.

Because just about every display connector interface is different we provide a standard connector on this board, and will provide 'display adaptor' boards to connect each type of display we support. An adaptor board carries these things:

- a connector for the display – often a fine-pitch FFC
- a back-light controller – allows on/off or dimming control of the back-light
- any extra voltage conversion devices as necessary

(If you design a custom application board for the VM2 you don't necessarily need to use this intermediate display adaptor board, though it may be a useful guard against the relatively short production lifetimes of many displays).

Configuration Example Code

See the *Venom2 Help File*, and code snippets on our website.

Audio output

Links: LK15 – link pins 1-2 for WAV sounds, or pins 2-3 for PWM sounds.
Resources used: VM2 Channels $14 or $18

This board carries a miniature loudspeaker and audio amplifier which you can use to generate beeps or play WAV files. There is a volume control located near the loudspeaker.

There are two ways to generate audio output. Firstly you can use a PWM 'square wave' generated from one of the VM2's timer outputs – using the PulseWidthOut object

```
; This is our beeper. Link LK15 pins 2-3 and turn pot up.
MAKE sounder PulseWidthOut($18,500,250,1, 100)
pwm . On ;turn it on
```

Secondly you can play certain types of WAV file, where the file is stored in the Flash File System:

```
MAKE audio Analogue($14,1)
audio.Send(file_name)
```

See the *Venom2 Help file* for more details.

Note: The speaker is too small to reproduce lower frequency sounds unaided, but the addition of a baffle of some kind improves it considerably. Given the layout of the PCB, the simplest baffle you can make is a small paper cone (25 to 50mm long), with some Blutak or similar to seal the narrow end to the speaker. The bigger the baffle the better the low frequency sound will be.

Power Supply

Caution! Supply this board with 12VDC, unless you know that attached subsystems (such as a Display Adaptor/Display Backlight) are compatible with a different supply voltage.

The board may be powered from a supply voltage range of less than 7.5V, up to 28 volts unregulated DC.
The current consumption depends on the input voltage, what is attached to the unit and whether the processor is idle, but power consumption is usually around 1 Watt when the VM2 is idle and with no external devices attached or subsystems configured, and perhaps 3 - 4 Watts with a typical configuration including graphics display.
The board has two regulators - a switch mode regulator to generate a 5V supply, and then a linear regulator to drop the 5V to 3.3V.
The main (switch mode) regulator is efficient over a wide input voltage range. There is a silicon diode before the regulator to protect against reverse polarity on the supply input.

Power Connector

2 way pluggable screw terminal. Plug provided.

Breakout connectors

The two breakout connector positions are provided at each end of the VM2, labelled BRK1 and BRK2. These are two sets of pads that exactly reproduce the connectors on the VM2, but on a 0.1” pitch. Note that the VCC and GND pads are marked with + and – signs.
You should refer to the table of channel usage on page 9 and the circuit diagram of this board before using the signals brought out here.

Table of links

This table details the function of each of the links on the board. The positions of all the links are shown on the diagram of the board.

Link	Open	Link pins 1 & 2	Link pins 2 & 3
LK1	Chan $34 free	Chan $34 used for Ethernet interrupt.	
LK2	USB pull up circuit not controlled	USB pull up circuit controlled by Dflt Comm signal	
LK3	Chan $3B is free	Chan $3B used for Serial Port 4 RX.	
LK4	RS485 not terminated	RS485 terminated	
LK5, LK6	RS485 not pulled to idle state	RS485 pulled to idle state	
LK7	CAN Bus not terminated	CAN Bus terminated	
LK8	CAN shield pin open	CAN shield pin grounded	
LK9	Alpha LCD back-light pin open	Alpha LCD back-light connected to Alpha LCD Vcc	
LK10	*I2C Bus not powered*	I2C Bus Vcc is 3.3V	I2C Bus Vcc is 5V
LK11	*Alpha LCD not powered*	Alpha LCD Vcc is 3.3V	Alpha LCD Vcc is 5V
LK12	*Back-light on/off signal open*	Backlight controlled by Chan $26	Back-light controlled by *BacklightPWM* signal (VM2D2 / 5920 only)
LK13, LK14	*CAN, USB not connected: Chans $1B, $1C are free*	CAN Tx, Rx connected: uses *Chans $1B and $1C*	USB data connected: uses *Chans $1B and $1C*

Link	Open	Link pins 1 & 2	Link pins 2 & 3
LK15	*No audio source, Chans $14, $18 are both free*	Audio from DAC output, uses chan $14	Audio from PWM output, uses chan $18

Table of channel usage

This table details each of the VM2 channels and what it is used for on this Application Board. Each of the channels also appears on the 'breakout' connectors labelled BRK1 and BRK2. The table describes what you must to do to free a channel so you may connect to it on a breakout connector. If this column is blank then the channel may be used immediately.

VM2 Channel	Use on 5922 Application Board	To use for other purposes
$10	Backlight control when LK12 pins 2-3 linked	Unlink pins 2-3 on LK12
$11		
$12	Serial2 Tx	Can be used as output
$13	Serial2 Rx	Not easy to free up
$14	DAC: Audio output [& SPI bus on GLCD connector]	Unlink using LK15 & Don't use SPI on GLCD connector
$15	[SPI bus on GLCD connector]	Don't use SPI on GLCD connector
$16	[SPI bus on GLCD connector]	Don't use SPI on GLCD connector
$17	[SPI bus on GLCD connector]	Don't use SPI on GLCD connector
$18	PWM Audio output	Unlink using LK15
$19	Serial1 Tx	Not easy to free up
$1A	Serial1 Rx	Not easy to free up
$1B	CAN Rx & USB	Unlink using LK14
$1C	CAN Tx & USB	Unlink using LK13
$26	Back-light ON/OFF/PWM	Don't use GLCD
$28	I2C Bus 1 SCL	Not easy to free up
$29	I2C Bus 1 SDA	Not easy to free up
$2A		
$2B		
$2C	SPI Bus 2: Ethernet, Memory Card	Can add more devices to the bus if SPI ADDR decoded
$2D	SPI Bus 2: Ethernet, Memory Card	[same as above]
$2E	SPI Bus 2: Ethernet, Memory Card	[same as above]
$2F	SPI Bus 2: Ethernet, Memory Card	[same as above]
$30		
$31		
$32		
$33		
$34	Ethernet interrupt	Unlink using LK1
$35	Memory card power enable	Not easy to free up
$36		
$37		
$38		
$39	USB Pull-up Control	Not easy to free up
$3A	Serial4 Tx	Can be used as output
$3B	Serial4 Rx	Unlink using LK3

$3C		
$42		
$43	Serial2 CTS	Not easy to free up
$52	GLCD Bus	Don't use GLCD
$56	GLCD Bus	[same as above]
$60	GLCD Bus, Alpha LCD Bus	[same as above] or ALCD
$61	GLCD Bus, Alpha LCD Bus	[same as above] or ALCD
$62	GLCD Bus, Alpha LCD Bus	[same as above] or ALCD
$63	GLCD Bus, Alpha LCD Bus	[same as above] or ALCD
$64	GLCD Bus, Alpha LCD Bus	[same as above] or ALCD
$65	GLCD Bus, Alpha LCD Bus	[same as above] or ALCD
$66	GLCD Bus	[same as above]
$67	GLCD Bus	[same as above]
$68	GLCD Bus	[same as above]
$69	GLCD Bus	[same as above]
$6A	GLCD Bus	[same as above]
$6B	GLCD Bus	[same as above]
$6C	GLCD Bus	[same as above]
$6D	GLCD Bus	[same as above]
$6E	GLCD Bus	[same as above]
$6F	GLCD Bus	[same as above]
$70	SPI Bus 2 Address 0 (Ethernet, Mem card)	Can be used as output so long as Ethernet and mem card not used
$71		
$72	Serial1 RTS	Not easy to free up
$73	Serial1 CTS	Not easy to free up
$74		
$75		
$76	Serial4 RTS	Can be used as output
$77		
$78		
$7B	Serial2 RTS	Can be used as output
$7D	GLCD Bus	Don't use GLCD
$7E	GLCD Bus	Don't use GLCD
$7F		

Controller capabilities

The table below shows which controllers are cable of driving which interfaces on this board.

Interface	**VM2**	**VM2D** **VM2-D2**	**VM2L**
2 off RS232 serial ports	●	●	●
1 off RS485 Serial Port (half-duplex)	●	●	●
Ethernet interface – 10 Base T	●	●	●
Memory card socket (MMC or SD)	●	●	X
I2C Bus (3.3V or 5V)	●	●	●
CAN Bus	●	●	●
USB connection to the VM2's Flash Filing System	●	●	X
Graphics Display interface	1	2	X
Alphanumeric LCD interface (3.3V or 5V)	●	X	●
Audio output: play WAV sounds	●	●	X
Audio output: PWM 'beeps'	●	●	●
EEPROM non volatile parameter storage memory	●	●	●
Matrix Keypad interface (4 x 4)	●	●	●
Power inlet (7.5 – 34 VDC)	●	●	●
Every VM2 pin appears on a 0.1” breakout connector.	●	●	●

Notes:
1: The VM2 can drive displays with on-board display controllers and 8 or 16 bit interfaces.
2: The VM2D (or VM2-D2) can drive controller-less displays that are compatible with the built-in display driver, e.g. TFTs at up to WVGA resolution.