LK204-24-USB
Technical Manual

Revision: 1.0

Contents

1 Introduction

The LK204-24-USB is equipped with the following features;

- 20 column by 4 line text display
- Built in font with provision for up to 8 user defined characters
- USB communication or serial TTL communication
- Use of up to 100 modules on the same USB host
- Fully buffered so that no delays in transmission should ever be necessary
- Ability to add a customized splash / start up screen
- Use of "Remember" functions to save settings
- Software controlled contrast
- Software controlled backlight brightness
- Backlight with configurable time-out setting up to 180 minutes
- Use of up to a 24 key keypad with a 10 key buffer
- Six general purpose outputs for a variety of applications
- 4 Advanced general purpose outputs
- 3 Low power general purpose ouputs
- Horizontal or vertical bar graphs
- Power and data from one USB cable
- Extended temperature option
- A perfect fit to Matrix Orbital's PC Bay inserts without any modifications
- Dallas 1-wire Bridge

1.1 What to Expect From the LK204-24-USB

The LK204-24-USB is designed as the display unit for an associated controller. The controller may be anything from a single board, special purpose micro-controller to a PC, depending on the application. This controller is responsible for what is displayed on the screen of the display.

The display provides a simple command structure which allows texts and bar graphs to be displayed on the screen. Text fonts are built in and use standard ASCII mapping. Provision is made for up to 8 user defined characters.

The screen is backlit for low light situations. Backlighting may be turned on or off under program control. Contrast is adjustable to compensate for differing lighting conditions and viewing angles.

General purpose outputs allow the controller to switch up to six electronic or electro-mechanical devices by issuing commands to the display unit. These can be used for controlling LEDs, relays, etc.

1.2 What Not to Expect From the LK204-24-USB

The display does not include bitmap graphics capability, except that permitted by defining special characters.

1.3 Keypad Interface

The keypad interface takes row / column input and converts it it ASCII characters, which are delivered out the USB port to the associated controller.

NOTE The keypad is not used to directly control any aspect of the operation of the display. The display acts simply as a matrix to serial converter.

1.4 Setup for Testing

Before setting up the application, it is best to test out the display. This is easily done with a PC. The following is a list of requirements for testing;

- An available powered USB source
- USB Drivers which can be located on Matrix Orbital's website and e-CD
- Test software such as Alpha Demo or Display Tuner

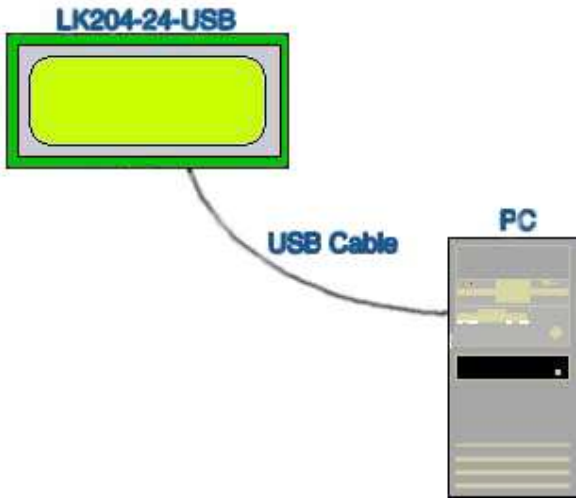


Figure 1: Connections for Testing

Once the above test requirements have been met, the user may proceed with the following steps;

1. Refer to the diagram above for the following steps.
2. Download or copy the USB drivers into a directory.
3. Uncompress the files. They will be a self extracting ZIP file.
4. Connect the USB cable to the display and the computer.
5. Windows will give a prompt for drivers for a USB ⇔ Serial Device.
6. Select "Specify location", and navigate to the directory the file was uncompressed to.
7. Test the display.

1.5 Trying out the LK204-24-USB

The unit should be connected to the USB and the backlight and custom startup screen should come on.

- Run a PC program such as Hyperterm to experiment with typing text. Make certain it's configured to use the correct port. Set the baud rate to 19,200 and turn flow control off.

Once this is complete, try typing some characters on the keyboard. The characters should now appear on the display screen.

Alpha Demo, Display Tuner or MOGD are excellent for basic display tests.

1.6 Trying out a Keypad

Since a number of different keypad types can be connected to the display, the results may be a little unpredictable. At this point the user should make certain the keypad and interface work, and possibly generate an ASCII map for programming needs.

The keypad interface on the display converts a row / column connection to an ASCII character. By default, a keypress is transmitted as serial data immediately. Keypad buffering can be selected using the appropriate commands.

The keypad should be a matrix style or momentary switches. It is possible to simulate key strokes by shorting out a row and column pin.

1.6.1 Here's what to do:

1. The PC should be running a terminal program, such as Hyperterm.
2. With the display connected to the PC, plug in the keypad. If the connector has fewer pins than the one on the display, centre it as well as possible.

NOTES

- The keypad connector must be wired with columns on one side and rows on the other side of the center of the connector. If the keypad isn't wired this way the user will need to make an adapter or rewire the connector to meet this requirement.
- The connector is reversible. Reversing the connector will not damage the keypad or the display, but it will however, change the ASCII character map

---

3. Press a key on the keypad. An upper case ASCII character (A-X) should appear on the PC screen. Different keys should generate different characters.

To experiment, reverse the connector to see if it generates a more logical set of characters. Ultimately the program in the controller will have to 'map' these characters to the ones marked on the keypad, which will likely be different.

2 Connections

2.1 Connector Pinout

Refer to the Figure below for this chapter.

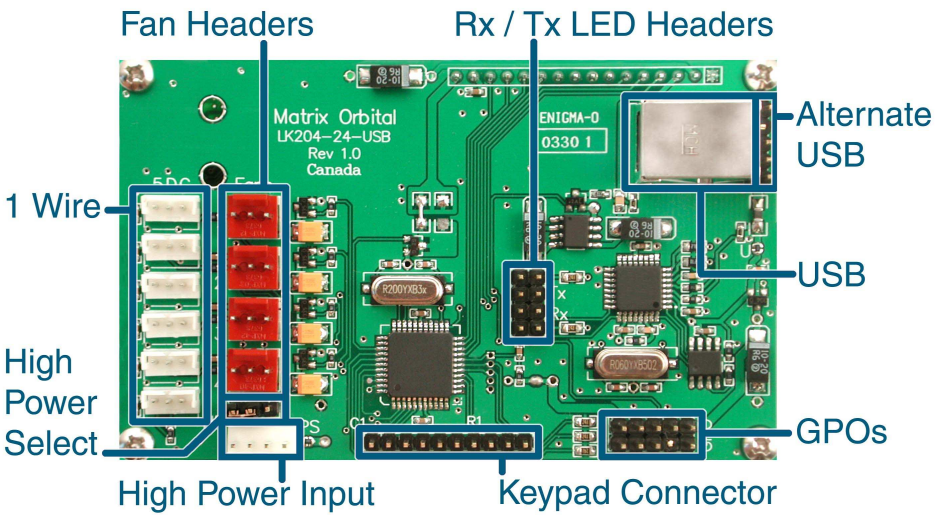


Figure 2: Electrical Connections

Table 1: Connectors & Functions

Connector	Function
14 Pin dual header	General Purpose Outputs and +5V Output
Rx / Tx	Receive and Transmit LED outputs
10 Pin header	Keypad
USB	Communication / Power
Alt. USB	Alternate USB / Power Header
4 Pin White Header	Optional High Power GPO Power Input

Table 2: Mating Connectors

Connector	Part #	Mate Part #
3 Pin White Header	AMP 173979	AMP 173977
3 Pin Red Header	MOLEX 43009	MOLEX 7879
10 Pin Dual Header	MOLEX 42375	Many, ex. MOLEX 70058
Rx/Tx	MOLEX 42375	Many, ex. MOLEX 70058
10 Pin Header	MOLEX 42375	Many, ex. MOLEX 70058
USB	MOLEX 67068	Standard USB cable
Alt. USB	MOLEX 42375	Many, ex. MOLEX 70058
4 Pin White Header	AMP 171825-4	AMP 170205-1

2.1.1 Power

USB High power ports can supply 500mA of power. Low power devices, such as un-powered hubs, can only supply 150mA of power. Please be sure to observe the total power usage on the display when connected to the USB port. The display will require between 150mA to 250mA depending on the model and the number of GPOs being used.

Table 3: GPO Power Levels

GPO	Maximum Power
+5V Low Power	20mA
+5V High Power	1000mA. Please note; an unmodified USB host can only supply a maximum of 500mA on a high power output and only 150mA on low power.
+12V High Power	1000mA

Power is applied via the USB cable or the alternate USB header. Power requirement is +5 VDC $\pm$0.25V and 180mA minimum.

WARNINGS

- When using the alternate USB header, verify all the cable pin outs before applying power. Incorrect power application may damage the display on one host.
- Make sure the host is capable of supplying all the necessary power. Add the display power requirement and the GPO power if used for a total power requirement.

The display has a secondary power connector used for High Power GPOs. Only if the user is planning on using GPO 4, GPO 5 and GPO 6 as +12V, should the user plug in a unmodified 3.5” floppy power cable.

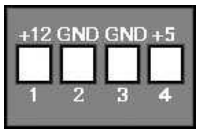


Figure 3: Header

Table 4: Connector Pinout

Pin	Description
Pin 4	+5.0 VDC (normally from PC power supply)
Pin 3	Ground
Pin 2	Ground
Pin 1	+12.0 VDC (normally from PC power supply)

Make certain to have sufficient current capacity to handle the desired load. Each High Power GPO can source over 750mA.

2.1.2 Power by 3.5” floppy cable

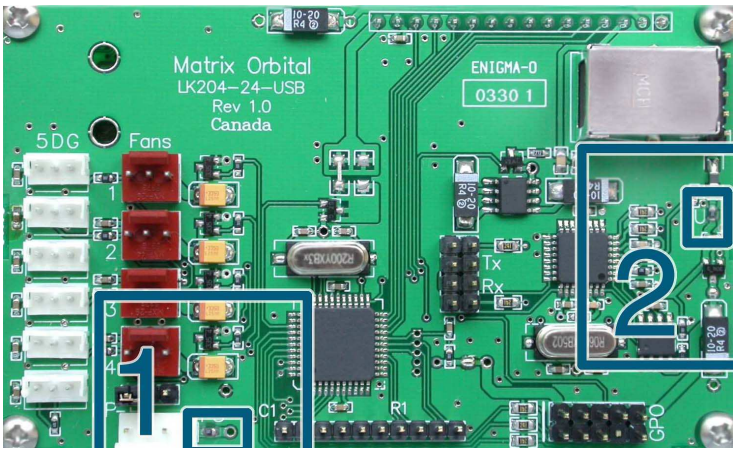


Figure 4: Jumper Locations

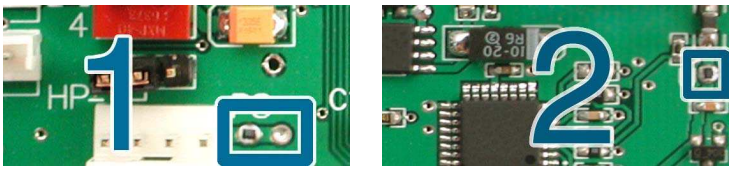


Figure 5: Jumpers 1 and 2

The display can be powered by the USB port or by external power. Soldering Jumper 1, and removing Jumper 2 will allow for power to be supplied externally. The advantage to this is the display will power up right away, allowing fans to start, for example, without waiting for the OS to power the display. In addition, if the advanced GPOs are set to the +5V position power will be drawn from external power rather than the USB port, thus allowing higher current draw.

NOTE Jumpers 1 and 2 are ON from factory.

2.1.3 USB Communications

A standard B type USB header is provided on the display for USB communication. The USB cable provides power and data to the display. There are two ways of communicating to the display. Under Windows, the user will have direct access to the display drivers or create a virtual COM Port (VCP). With the VCP, a new COM Port gets created in Windows. Under the control panel the user can set all the options for the USB device, including speed. Windows 98, Windows ME, Windows 2000, and Windows XP drivers have been tested and work. VCP drivers are also available for Apple OS-X, OS-8, and OS-9. Linux drivers are available as well, but have not been tested by us, nor will Matrix Orbital be able to provide any support for them.

For more information please visit `www.FTDIchip.com`.

8bit, no parity, one stop bit.

Speed: 19.2 Kb/s.

Table 5: USB ID

VID 0403

PID	Description
FA00	USB 2 Serial Communication
FA01	MX2 / MX3
FA02	MX4 / MX5
FA03	LK / VK202-24-USB
FA04	LK / VK204-24-USB

2.1.4 Alternate USB Communications

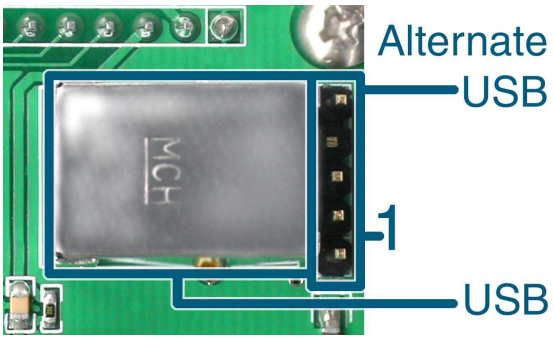


Figure 6: USB & Alternate USB

Table 6: Connector Pinout

Pin Number	Description
5	Ground
4	Not Used
3	D+
2	D-
1	+5V

2.1.5 Serial TTL Communication

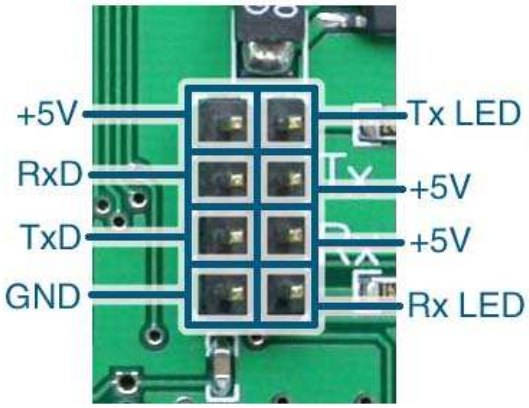


Figure 7: Rx / Tx LED Header

The display can also be talked to at TTL (logic 0V to +5V) levels. This will allow TTL communications through the 4-pin header. Communicating by Serial TTL will bypass the USB components and power will be required to be provided by this header.

8bit, no parity, one stop bit.

Speed: 19.2 Kb/s.

NOTE Do not power or communicate by USB and Serial TTL at the same time.

2.1.6 LED Headers

The LED headers provide a visual indication of communication over the Rx and Tx line if a LED is connected. The LEDs blink frequency will increase as the amount of data increases. The headers are current limited to 20mA at +5V. This feature only works with USB communication and will not work with Serial TTL.

2.2 General Purpose Outputs

The display has a number of general purpose outputs for a variety of applications.

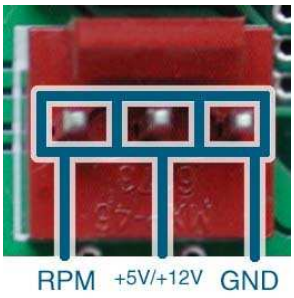


Figure 8: Fan Header

2.2.1 Advanced General Purpose Outputs & GPO's 1 to 4

These outputs are capable of controlling high current draw devices, saving power up state, being switched to PWM mode, and repeat back RPM via Hall effect sensor.

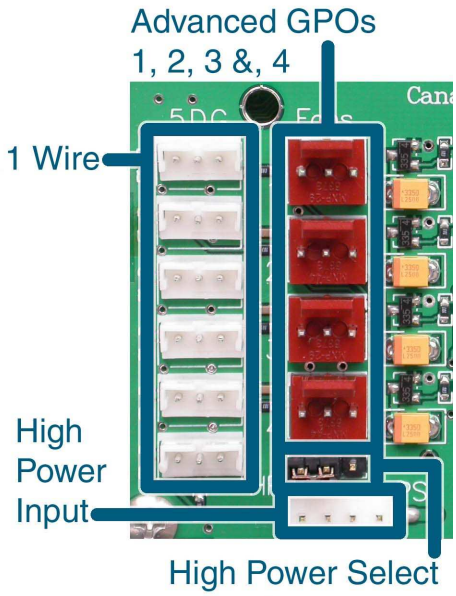
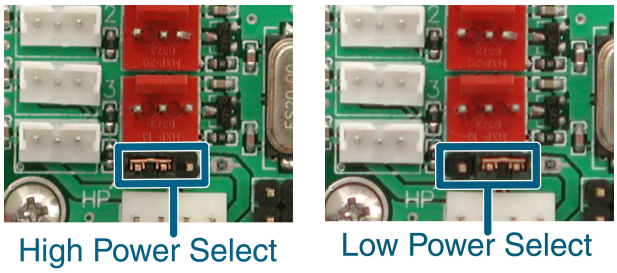


Figure 9: GPOs 1 to 4

WARNINGS

- RPM reading should not be used at 5V
- There are no current limiting resistors
- Maximum current draw is 1000mA
- As of Firmware Revision 1.1, the default PWM frequency is 19.1Hz
- Default voltage setting is 12V

2.2.2 Advanced GPO Power Select



12 Volt 5 Volt

Figure 10: High / Low Power Select

When the jumper is placed in the high power position, the unit supplies +12V to the advanced GPOs. If the jumper is placed in the low power position +5V is supplied to the GPOs. For 12V, external power supplied by a 3.5” floppy connector will be required.

2.2.3 GPO 5, GPO 6, and GPO 7

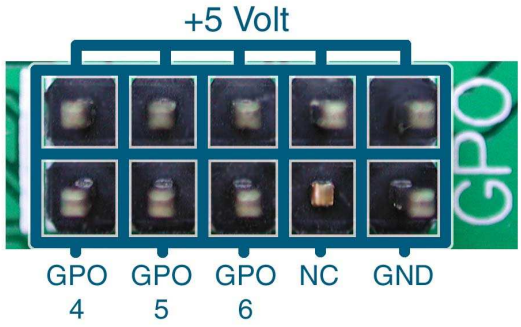


Figure 11: General Purpose Outputs

GPO's 5, 6, and 7 are low power ouputs providing +5V at 20mA enforced by a current limiting resistors. They are ideal for driving LEDs and relays directly.

2.2.4 Dallas 1-Wire Bridge

The display offers one Dallas 1-wire bridge. All 6 headers are inter-connected to one communication line. A maximum of 32 1-wire devices can be connected to the display at a time.

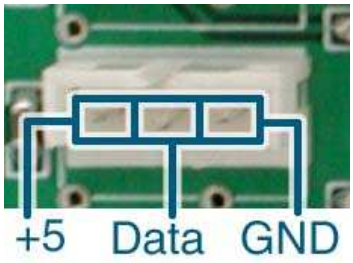


Figure 12: 1-Wire Pinout

2.3 General

Text is displayed on the LK204-24-USB using the built in 5x8 dot matrix font, in addition to up to 8 user defined characters.

2.4 The Built In Character Font

The display includes a built in 5x8 dot matrix font with the full range of ASCII characters plus a variety of extended characters, as shown in the Figure below.

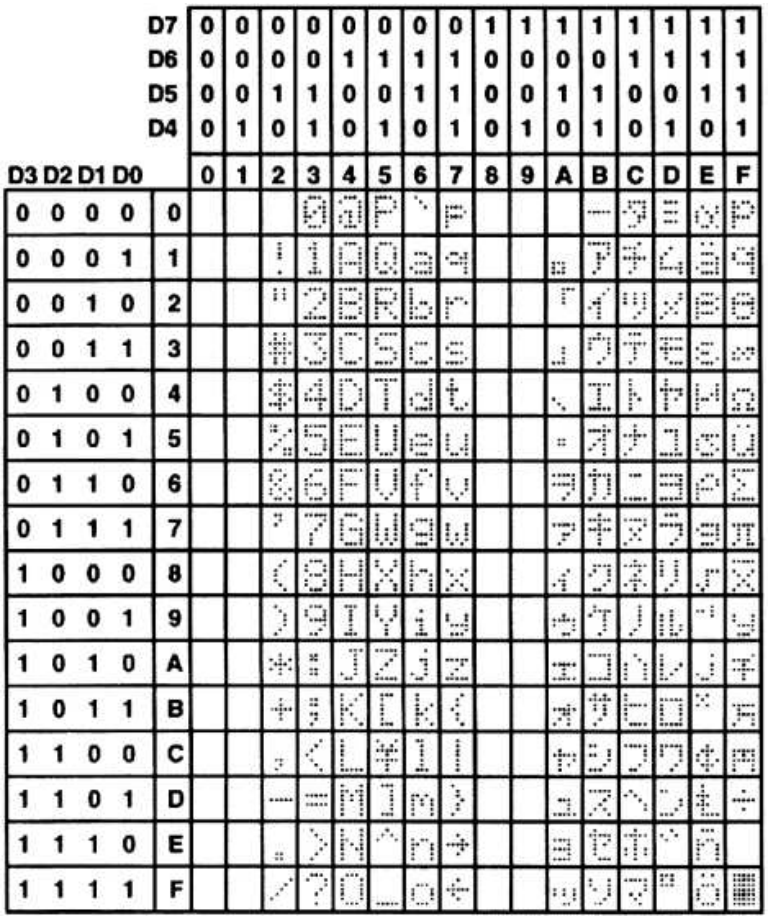


Figure 13: Character Set

In addition to the built in characters, users may define up to 8 special characters, which once defined, occupy positions 0x00 to 0x07 in the above chart. The display does not have provision to download other fonts.

2.5 Writing Text to the Display

When the display receives a character, it displays that character at the position currently defined. The next character sent to the module then advances to the following position on the display. Characters are drawn using the built in font, and only characters defined in the font are actually displayed. Characters which are not defined by the built in font print as a space.

The position where text is to be inserted is a character location stored in the display's volatile memory and maintained internally by the display's firmware. This position is manipulated by the commands shown in the following section.

2.6 Text Commands

In this section commands are identified by their names and decimal values.

Some commands marked with an "R", may be Remembered" to set new defaults that will be in effect each time the unit is powered on.

2.6.1 Auto Line Wrap On (254 67)(R)

Enables automatic line wrapping. Note that this word is not 'word wrapping' and wraps may occur in the middle of a word.

2.6.2 Auto Line Wrap Off (254 68)(R)

Disables automatic line wrapping. Characters beyond the end of a line may be lost.

2.6.3 Auto Scroll On (254 81)(R)

When auto scrolling is on, it causes the display to shift the entire display's contents up to make room for a new line of text when the text reaches the scroll position, which is the bottom right character position.

2.6.4 Auto Scroll Off (254 82)(R)

When auto scrolling is disabled, text will wrap to the top left corner of the display area. Existing text in the display area is not erased before the new text is placed. A series of 'spaces', followed by a "Cursor home" command, may be used to erase the top line of text.

2.6.5 Set Cursor Position (254 71 [column][row])

This command sets the cursor position (text insertion point) to the [column] and [row] specified. Columns have values from 1 to 20 (0x01 to 0x14) and rows have values of 1 and 2 (0x01 and 0x02).

2.6.6 Send Cursor Home (254 72)

This command moves the cursor position (text insertion point) to the top left of the display area.

2.6.7 Turn On Underline Cursor (254 74)(R)

Turns on the underline cursor. The cursor shows the current text insertion point. Both underline and blinking cursors may be turned on or off independently. The cursor if off by default.

2.6.8 Turn Off Underline Cursor (254 75)(R)

Turns off the underline cursor. Does not affect the blinking block cursor.

2.6.9 Turn On Block (blinking) Cursor (254 83)(R)

Turns on the blinking block cursor. The cursor shows the current text insertion point. Both blinking and underline cursors may be turned on or off independently. The cursor is off by default.

2.6.10 Turn Off Block (blinking) Cursor (254 84)(R)

Turns off the blinking block cursor. Does not affect the underline cursor.

2.6.11 Cursor Left (254 76)

Moves the cursor one position to the left but does not erase any character that may be in that position. Note that this command moves the text insertion point even if the cursor is turned off.

NOTE A 'destructive backspace', which erases the character to the left of the original position, may be done by issuing the following sequence: cursor left, space, cursor left.

2.6.12 Cursor Right (254 77)

Moves the cursor one position to the right but does not erase any character that may be in that position. Note that this command moves the text insertion point even if the cursor is turned off.

3 Keypad Interface

This chapter describes the keypad interface and associated commands in detail.

3.1 General

The display keypad interface processes the keypad row / column matrix into a serial data byte stream. Aside from this processing, the keypad has no effect on the display. To send keystrokes to the display, they must be routed through the controller.

3.2 Connections

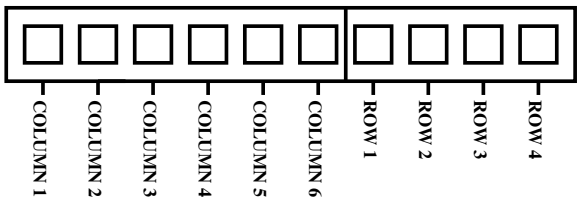


Figure 14: Keypad Connector

The connector is not 'keyed', so the keypad will probably plug in either of two ways. The display will not be damaged by reversing the connector. However, the keypad will generate a different ASCII character mapping for each position. If the connector has fewer than 10 pins it should be centered on pins 6 and 7 of the connector.

Pins 1 through 6 are columns, and Pins 7 through 10 are rows. The keypad is scanned wherever a key is passed; there is no continuous key scan. This means that keypresses are dealt with immediately without any appreciable latency. This also prevents electrical noise which is often caused by continuous key scans.

NOTE Please note that keypads may be laid out in a different pattern. If this is the case, the user will need to interpret the key codes differently.

Table 7: Keypad Layout

	Columns						
		1	2	3	4	5	6
	1	A	B	C	D	E	F
Rows	2	G	H	I	J	K	L
	3	M	N	O	P	Q	R
	4	S	T	U	V	W	X

NOTE The keypad connector must be wired with columns on one side and rows on the other side of the centre of the connector. If the keypad isn't wired this way the user will need to make an adapter or rewire the connector to meet this requirement.

3.3 Keypad Commands

Some commands, marked with an "R", may be "Remembered" to set new defaults that will be in effect each time the unit is powered on.

3.3.1 Auto Repeat Mode On (254 126[mode])(R)

[mode]=0x00 gives Resend Key Code mode

[mode]=0x01 gives Key Down / Key Up Code mode

Two modes of auto repeat are available and are set via the same command.

1. **Resend Key Mode:** This mode is similar to the action of a keyboard on a PC. In this mode, when a key is held down, the key code is transmitted, immediately followed by a 1/2 second delay.
2. **Key down / Key up codes:** This code may be used when the typematic parameters of the "Resend key code" mode are unacceptable or if the unit is being operated in polled mode. The host system detects the press of a key and stimulates an auto repeat inside the host system until the key release is detected.

In this mode, when a key is held down, the key code is transmitted immediately and no other codes will be sent until the key is released. On the release of the key, the key release code transmitted will be a value equal to the key down code plus 20 hex. For example, if the key code associated with key "P"(0x50) is pressed, the release code is "p"(0x70).

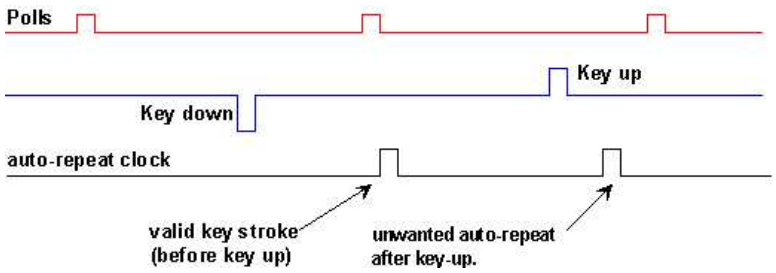


Figure 15: Poll Timing

3.3.2 Auto Repeat Mode Off (254 96)(R)

This command turns off the auto repeat mode

3.3.3 Auto Transmit Keypresses On (254 65)(R)

In this mode, all keypresses are sent immediately to the host system without the use of the poll keypad command. This is the default mode on power up.

3.3.4 Auto Transmit Keypresses Off (254 79)(R)

In this mode, up to 10 keypresses are buffered until the unit is polled by the host system via the poll keypad command. Issuing this command places the unit in polled mode.

3.3.5 Clear Key Buffer (254 69)

This command clears any unread keypresses. In a menuing application, if the user presses a key which changes the menu context, any following key presses may be inaccurate and can be cleared out of the buffer between menu changes to prevent jumping around the menu tree. It may also be used to, in effect, reset the keypad in case the host application resets for whatever reason.

3.3.6 Poll Keypad (254 38)

The host system must be set up to receive the key codes. When the display receives this command it will immediately return any unbuffered keypresses which may have not been read already. If there is more than one keypress buffered, then the high order bit (MSB) of this returned keycode will be set (1). If this is the only buffered keypress, then the MSB will be reset (0). If there are no buffered keypresses, then the returned code will be 0x00. Please note, in order to make use of this command, the “Auto transmit keypress” mode should be off.

3.3.7 Set Debounce Time (254 85 [time])(R)

This command sets the time between key press and key read. All key types with the exception of latched piezo switches will 'bounce' for a varying time depending on their physical characteristics. The default debounce time for the module is about 52 mS, which is adequate for most membrane keypads.

4 Bar Graphs and Special Characters

The display includes the ability to draw bar graphs (either horizontal or vertical) and allows users to define up to eight special characters.

Eight characters (ASCII values 0x00 to 0x07) are set aside for use with bar graphs, user defined characters, and big numbers. Since the same 8 characters are used for each function, **the functions may not be used simultaneously**. The characters may be defined or redefined at any time by issuing the commands shown in this section. Once defined, they may be used either by means of the bar graph commands, or by simply issuing one of the ASCII values 0x00 to 0x07, which is not prefixed by the command byte, 254.

4.1 Command List

4.1.1 Initialize Wide Vertical Bar Graph (254 118)

This command defines the 8 special / user characters to be blocks suitable for use in drawing wide (5 pixel) vertical bar graphs. Any previously existing definitions will be lost. Once this command has been issued, any number of vertical bar graphs may be drawn unless the characters are redefined by another command.

4.1.2 Initialize Narrow Vertical Bar Graph (254 115)

This command defines the 8 special / user characters to be blocks suitable for use in drawing narrow (2 pixel) vertical bar graphs. Any previously existing definitions will be lost. Once this command has been issued, any number of vertical bar graphs may be drawn unless the characters are redefined by another command.

4.1.3 Draw Vertical Bar Graph (254 61 [column][height])

Draws a vertical bar graph in [column] having a height of [height] pixels. The height may range from 0 to 20 (0x00 to 0x14) pixels. The necessary characters must first be initialized by either of the commands shown in section 4.1.1 or 4.1.2, which will determine the width of the bar graph. The graph may be erased by drawing a bar graph of height=0 in the same column.

4.1.4 Initialize Horizontal Bar Graph (254 104)

This command defines the 8 special / user characters to be blocks suitable for use in drawing horizontal bar graphs. Any previously existing definitions will be lost. Once this command has been issued, any number of horizontal bar graphs may be drawn unless the characters are redefined by another command.

4.1.5 Draw Horizontal Bar Graph (254 124 [column][row][dir][length])

Draws a horizontal bar graph in [row] starting at [column] with a length of [length] pixels. [row] may have a value of 0x01 or 0x02, column may range from 0x01 to 0x14 and length may be from 0x00 to 0x64 (0 to 100) if the graph can extend to the full width of the screen. Each column is 5 pixels wide (spaces between the columns don't count).

[dir] specifies the direction: 0x00 goes from left to right, 0x01 goes from right to left.

4.1.6 Define Custom Character (254 78 [c][8 bytes])

The display allows up to 8 user defined (custom) characters. These characters occupy the first 8 (0x00 to 0x07) places in the character set.

Custom characters occupy a 5x8 pixel matrix. Built in characters are 5x8; the bottom row of pixels is normally reserved for the underline cursor. The underline cursor should be turned off if the bottom row of pixels forms part of a custom character.

The characters are defined by issuing the command 254 78 [c] followed by 8 bytes to define the character. [c] is the character number (0x00 to 0x07). The 8 bytes are mapped as shown in the Table below.

Table 8: 8 Byte Map

MSB							**LSB**	
*	*	*	1	2	3	4	5	Data Byte 1
*	*	*	6	7	8	9	10	Data Byte 2
*	*	*	11	12	13	14	15	Data Byte 3
*	*	*	16	17	18	19	20	Data Byte 4
*	*	*	21	22	23	24	25	Data Byte 5
*	*	*	26	27	28	29	30	Data Byte 6
*	*	*	31	32	33	34	35	Data Byte 7
*	*	*	36	37	38	39	40	Data Byte 8

A "1" bit indicates an "on" (black) pixel, while a "0" indicates an "off" (clear) pixel.

Once defined, a character is displayed simply by issuing a value (0x00 to 0x07) corresponding to the character number. The character will be laid out as follows;

Table 9: Character Values

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25
26	27	28	29	30
31	32	33	34	35
36	37	38	39	40

NOTE Custom characters will be erased if any of the "Initialize bar graph" commands are issued.

Example of a degree symbol;

```
txUart.sendByte(0xFE); //command prefix
txUart.sendByte('N'); //custom character command
txUart.sendByte(0x00); //custom character value 0-7
txUart.sendByte(12); //8 bytes to create
txUart.sendByte(18); //the custom character
txUart.sendByte(18);
txUart.sendByte(12);
txUart.sendByte(0);
txUart.sendByte(0);
txUart.sendByte(0);
txUart.sendByte(0);
txUart.sendByte(0x00); //display custom character 0
```

4.1.7 Remember Custom Character (254 194 [c][8 bytes])

This command will store a custom character to be used with the "Custom startup screen". It does not affect or alter the current custom characters that are stored in the unit. The syntax is identical to the previous command.

5 Fan and GPO Commands

5.1 Display Return Protocol

To facilitate the reporting of information other than keypresses, the "Display return protocol", (DRP) was developed. This protocol allows the display to return arbitrary information back to the controller. This

protocol is used for reading fan speeds and retrieving 1-wire bus information. The basic structure of the protocol is described in the Table below.

Table 10: Display Return Protocol

Offset (Bytes)	**Length** (Bytes)	**Value**	**Description**
0	2	0x23 0x2A	Preamble
2	1		Continued / Size
3	1		Packet Type
4	1–127		Packet Data

The first two bytes are the standard preamble to separate the protocol from returned keypresses. The next byte is described in the Table below. The CONT flag specifies that the data will be continued in the next DRP packet. The lower seven bits contains the size of the data section excluding the four byte header. The type specifies what type of information is contained in the packet. Finally, the data returned is specific to the packet type.

Table 11: Continued / Size Byte

Bit	**Description**
7	Continued
6	Packet Size
5	
4	
3	
2	
1	
0	

Table 12: Display Return Protocol Types

Decimal	**Hex**	**ASCII**	**Description**
49	0x31	'1'	1-Wire data
82	0x52	'R'	Fan RPM data

5.2 Fan and GPO Commands

5.2.1 General Purpose Output Off (254 86 [gpo #])

This command turns OFF any of the general purpose outputs. [gpo #] is 1 to 6.

5.2.2 General Purpose Output On (254 87 [gpo #])

This command turns ON any of the general purpose outputs. [gpo #] is 1 to 6.

5.2.3 PWM Value (254 192 [fan #] [PWM value])

This command sets one of the high power GPOs (GPOs 1-4) into PWM mode. This permits speed control of a fan. A PWM value of 0 is off, 128 is 50% power, and 255 is full power. The fan# can be 1 to 3.

5.2.4 Return Fan RPM (254 193 [fan #])

This command will return a packet to the host with the fan RPM contained in it. The structure of the fan is described in the Table below.

Table 13: Return RPM Structure

Offset	Size	Description
0	2	0x232A
2	1	0x03
3	1	0x52
4	1	Fan #
5	2	Fan Period (MSB first)

To convert the fan period into an RPM value, follow the following formula;

$$RPM = \frac{18750000}{X * n}$$

Where X is the fan period, and n is the number of ticks that the fan produces per period. The number of ticks is usually 1, 2 or 4. For unknown fans some experimentation is required.

NOTE It is not recommended that the RPM is checked more than once every two seconds. If the RPM is checked more frequently, the actual RPM readings can become very erratic.

5.2.5 Remember GPO / PWM State (254 195 [fan #] [PWM value])

This command will set the startup state for all the GPOs. When the device is powered up the next time, the GPOs will be set to the values from this command. For GPOs 1 to 4 it behaves exactly as the PWM Value command. For GPOs 5 to 7, a non-zero value for the PWM value make the GPO on for future startups. A PWM value of 0 will result in the GPO being off.

This command does not affect the current state of the GPOs or fans, just power up.

5.2.6 Set PWM Base Frequency (254 196 [index])

NOTE This command was added in Firmware Version 1.1. It is not present in previous versions.

This command sets the base frequency for the PWM modulation. The index selects a present frequency as shown in the following Table.

Table 14: PWM Base Frequencies

Index	Frequency	Steps
0	0.3 Hz	256
1	0.6 Hz	256
2	1.2 Hz	256
3	2.4 Hz	256
4	4.8 Hz	256
5	9.6 Hz	256
6	19.1 Hz	256
7	38.2 Hz	256
8	76.3 Hz	256
9	152.6 Hz	129
10	305.2 Hz	65
11	610.4 Hz	33
12	1220.7 Hz	17
13	2441.4 Hz	9
14	4882.9 Hz	5
15	9765.8 Hz	3

As the frequency increases,the number of valid PWM states will decrease. For example, with an index of 14, there are only 5 PWM states.

Table 15: Example PWM Values for Index = 14

Input PWM	Actual PWM
0	0
1 -63	25%
64 -127	50%
128- 191	75%
192- 255	100%

Frequencies in the range of 9.6Hz to 38.2Hz are desirable for fan control as they minimize the noise due to PWM modulation. For visual applications such as controlling cold cathode lights, PWM frequencies of 76.3Hz to 305.2Hz are desirable to minimize flicker.

5.2.7 Remember PWM Base Frequency (254 197 [index])

NOTE This command was added in Firmware Version 1.1. It is not present in previous versions.

This command will set the PWM frequency for startup. It does not alter the current PWM base frequency. Refer to the "Set PWM base frequency" command for valid values of index.

6 1-Wire Commands

The 1-wire bus is capable of communicating with many devices over a single wire plus a ground reference. This chapter deals with the capabilities of the display and a brief introduction to the 1-Wire standard. For more detail consult `www.maxim.com`

6.1 Device Identification

Each 1-wire device contains a unique 64-bit address in which to identify them with. The address is guaranteed to be unique from any other device, allowing a virtually unlimited number of devices on to be attached to the bus. The address itself contains a family code and a cyclic redundancy check (CRC). The family code is unique to a particular device model. For example, the family code for the DS18S20 temperature probes is 10H. The CRC byte is included as a verification that the correct address was transmitted or received.

6.2 Protocol

The transaction sequence for accessing a 1-wire device is as follows;

1. Reset and detect presence.
2. ROM command, followed by any required data exchange.
3. Device specific function command, followed by any required data exchange.

Before communication can begin, the bus must be reset to force all devices to begin listening. After the reset, all the devices will transmit a presence pulse which indicates that there is at least one device on the bus. Once the presence of at least one device has been confirmed, the master must select which group of devices will be involved in the rest of the transaction. A device will use a ROM command to determine if the following transaction is intended for it. If not, the device will ignore all communication on the bus until the next bus reset. Otherwise, the device will read and process the rest of the transaction. The final part of the transaction is the device specific function command. To determine what functions your device will respond to, consult the devices' data sheet.

It is very important to follow this sequence for every transaction. If any of these steps are omitted or performed in the wrong order, the devices will not respond.

None of the device addresses are known ahead of time, and as such, each of their address must be searched out and determined. The 1-wire bus provides a means of searching the bus for devices and determining their address.

6.3 ROM Commands

The ROM commands allow a device to be singled out for communication or all devices to be included. This manual only presents the three most used ROM commands. For a more detailed listing and description of all the ROM commands, consult the data sheet for the 1-wire device being used.

- **Match ROM [55h]:** To single out a device the Match ROM command is used. After this command has been issued the 64-bit target address is transmitted in LSB to MSB order. Any device with an address that doesn't match will ignore all further communication until the next bus reset.
- **Skip ROM [CCh]:** After this command, all devices will continue to listen and process the transaction. This is equivalent to broadcasting to all devices. This command is useful when probes need to be informed to get their measurement ready. With this command all the probes can be instructed simultaneously.
- **Read ROM [33h]:** All devices will begin transmitting their address after this command. This command will only succeed when there is one device on the bus. When multiple devices are present, all devices will begin transmitting their addresses, overlapping each other. This command can be used to determine if there is more than one device on the bus. After the address has been read back, if the CRC is valid, there is only one device on the bus. Otherwise there are multiple devices on the bus and each address must be searched out.

6.4 Display 1-Wire functions

The transaction command allows data to be put onto the bus and read off the bus for transactions. And the search command identifies all the devices on the bus for further communication.

6.4.1 Transaction command (254 C8 1 [flags] [Send Bits] [Recieve bits] [Send data])

The transaction command will perform a single transaction on the 1-wire bus in this order;

1. Bus Reset.
2. Transmit data onto the bus.
3. Receive data from the bus.

Table 16: 1-Wire Transaction

Offset (Bytes)	**Length** (Bytes)	**Name**	**Description**
0	1	Flags	The flag byte controls the optional components of the transaction.
1	1	Send Bits	The number of bits that will be transmitted onto the bus. The actual bits to be transmitted are held in the Send Data section.
2	1	Recieve Bits	The number of bits to read off the bus after the data to be put on the bus has been sent.
3	Variable	Send Data	The data to be transmitted onto the bus. The data is transmitted MSB to LSB in the order that they are received.

Table 17: 1-Wire Flags

Bit	**Description**
7	Unused (0 for future compatibility)
6	
5	
4	
3	Add a CRC8 to the end of the transmitted data
2	Unused (0 for future compatibility)
1	Assume last received byte is a CRC8 and validate it
0	Reset bus before transaction

The received data is sent back using the “Display return protocol”. The return type will be 0x31 or ’1’, and the error codes are described in the Table below.

Table 18: 1-Wire Error Codes

Code	Description
0x00	Success
0x01	Unknown 1-Wire command
0x02	No devices on the bus
0x03	Fatal search error

6.4.2 Search command (254 C8 2)

This is used to find the addresses of all 1-Wire devices on the bus. After this command the display will return one or more “Display return protocol” packets containing either an error code or addresses of 1-wire packets. The structure of these packets is shown in the Table below.

Table 19: Search Return Packet

Offset (Bytes)	**Length** (Bytes)	**Description**
0	2	0x232A - Preamble.
2	1	0x8A - Packet 10 bytes long, another address will follow. 0x0A - Packet is 10 bytes long, this is the last address.
3	1	0x31 - 1-Wire Packet Type
4	1	Error Code (0x00 for success)
5	8	1-Wire Address
13	1	CRC8 - 0x00 means the last address was valid.

7 Miscellaneous Commands

The commands listed in this chapter don’t readily fit in any of the other categories, or are used in more than one category.

7.1 Command List

Some commands, marked with an “R”, may be "Remembered" to set new defaults that will be in effect each time the unit is powered on.

7.1.1 Remember (254 147 [0|1])

This command allows a number of settings, such as cursor state, backlight, etc., to automatically be stored in non-volatile memory, so they become new defaults. The command should only be used when required for two reasons;

1. Writing to non-volatile memory is time consuming and slows down the operation of the display.
2. Non-volatile memory has a 'write limit' and may only be changed approximately 100,00 times.

Commands which may be used with the remember function are marked with an "R" in the default column in the command tables.

The example shows the procedure to set "Auto scroll on" as the default condition. Commands are shown in both decimal and hex in the Table below.

Table 20: Command Example

Decimal	Hex	Function
254 147 1	FE 93 01	Turn on "Remember" function
254 81	FE 51	Turn on auto scroll. Since "Remember" is ON, this setting will be saved in non-volatile memory
254 147 0	FE 93 00	Turn off "Remember" function

Any number of commands may be entered between the "Remember ON", and "Remember OFF" commands, and all settings will be memorized.

7.1.2 Clear display (254 88)

This command clears the display and resets the text insertion point to the top left of the screen.

7.1.3 Set Contrast (254 80 [contrast])

This command sets the display's contrast to [contrast], where [contrast] is a value between 0x00 and 0xFF (between 0 and 255). Lower values cause "on" elements in the display area to appear lighter, while higher values cause "on" elements to appear darker. Lighting conditions will affect the actual value used for optimal viewing. Individual display modules will also differ slightly from each other in appearance. In addition, values for optimal viewing while the display backlight is on may differ from values used when the backlight is off.

7.1.4 Set Contrast and save (254 145[contrast])

This command works in exactly the same way as the "Set contrast" command. The only difference is that it saves the contrast value in the non-volatile memory of the module, whereas the previous command

only changes the value until the next power down.

7.1.5 Backlight on (254 66[minutes]) (R)

This command turns on the backlight for a time of [minutes] minutes. If [minutes] is zero (0), the backlight will remain on indefinitely.

NOTE The factory default for backlight is on.

7.1.6 Backlight off (254 70) (R)

This command turns the display off.

7.1.7 Set Backlight Brightness (254 153[brightness])

This command sets the display's backlight brightness.

7.1.8 Set and Save Backlight Brightness (254 152[brightness])

This command saves [brightness] as default.

7.1.9 Load Startup Screen (254 64 [40 characters])

This command sets and memorizes the startup screen which will appear each time the display is turned on. By default the screen shows;

Table 21: Default Screen

Matrix Orbital
LK204-24-USB

The 80 characters define the four 20 character rows of the screen.

Table 22: Default Screen

Character 1	Character 20
Character 21	Character 40
Character 41	Character 60
Character 61	Character 80

If sending more then 10 characters to be stored, add in ~10ms per character delay. Predefined custom characters can be used in the “Startup screen” as well, by using 0x00 through 0x07 characters.

7.1.10 Read Module Type (254 55)

This command will return the model type value of the module as a 1 byte hex value. Values for various modules at the time of this publication are as follows;

Table 23: Module Values

LCD0821 - 0x01	LCD2021 - 0x03	LCD2041 - 0x05
LCD4021-0x06	LCD4041 - 0x07	LK204-24-USB - 0x08
LK204-25-0x09	LK404-55 - 0x0A	VFD2021 - 0x0B
VFD2041 - 0x0C	VFD4021 - 0x0D	VK202-25 - 0x0E
VK204-25 - 0x0F	GLC12232 - 0x10	GLC24064 - 0x13
GLK24064-25 - 0x15	GLK12232-25 - 0x22	LK404-AT - 0x31
LK402-12 - 0x33	LK162-12 - 0x34	LK204-25PC - 0x35
LK202-24-USB - 0x36	VK202-24-USB - 0x37	**LK204-24-USB - 0x38**
VK204-24-USB - 0x39		

7.1.11 Set Serial Number (254 52 [byte1][byte2])

Modules may be delivered with the serial number blank. In this case the user may set the desired 2 byte serial number using this one time only command.

Upon the execution of this command, the module will echo these two bytes back over the RS-232 interface. The serial number may be set only once. Any future attempt to execute this command will result in no change and the module will return to the originally set serial number.

7.1.12 Read Serial Number (254 53)

This command will return a 2 byte hex value.

7.1.13 Read Version Number (254 54)

This command will return a 1 byte hex value.

8 Appendix: Command Summary

8.1 General

The operation of the LK204-24-USB is controlled by a simple and consistent command set. Commands control;

- Text display
- Graphic display
- Keypad interface
- Miscellaneous operating parameters

This chapter includes summary tables of all commands.

8.2 Issuing Commands

Commands are issued to the display by the controller. In a test setup, commands can be issued to the display by means of a BASIC program, using the chr$() function. In the tables below, commands are shown in hex, ASCII and decimal form. All commands begin with the prefix character 0xFE (254 decimal). These commands are issued on the serial communications link USB, at the currently defined baud rate.

For example (using a BASIC setup), the user could issue the command to clear the screen on the display by including the line;

```
PRINT#1,chr$(254);chr$(88)
```

Or, with C the user could (using Zcomm serial library)

```
ZComm1->WriteCommByte (0xfe) ;
ZComm1->WriteCommByte ('X') ;
```

8.3 On Numbers

Like all computerized devices, the display operates with commands and values in the form of binary numbers. These binary numbers are arranged in 8 digit (i.e., 8 bit) groups called bytes. The decimal value of a byte may have any value from 0 to 255.

Bytes are usually specified in either decimal or hexadecimal (base 16) form for convenience, since binary numbers are confusing to deal with directly. Hexadecimal (hex) numbers are particularly convenient because exactly two hexadecimal digits make up one byte, each hex digit representing 4 binary digits (4 bits) as shown in the Tale below.

Table 24: Hex Value Table

Binary	Hex	Decimal	Binary	Hex	Decimal
0000	0	0	1000	8	8
0001	1	1	1001	9	9
0010	2	2	1010	A	10
0011	3	3	1011	B	11
0100	4	4	1100	C	12
0101	5	5	1101	D	13
0110	6	6	1110	E	14
0111	7	7	1111	F	15

Based on the table, the byte 01001011 can be represented in hex as 4B, which is usually written as any of 4Bh, 4BH, 4B hex or 0x4B.

The numbers can be expressed in decimal form if preferred.

8.3.1 ASCII Characters

Since computers deal internally with numbers only, but externally with both letters and numbers, several schemes were developed to 'map' written characters to numeric values. One such scheme has become universal; the American Standard Code for Information Interchange, or ASCII. ASCII tables are readily available from a number of sources. A few examples will do here;

Table 25: Example of an ASCII Table

The letter	A	has a value of	65 decimal or	41 hex
The letter	a	has a value of	97 decimal or	61 hex
The number	0	has a value of	48 decimal or	30 hex
The number	9	has a value of	57 decimal or	39 hex

This gives rise to the possibility of confusion when parameters are being set on the display. For example, the GPO ON and OFF commands use a number to indicate which GPO is being controlled. We're told that acceptable values are 0 to 6. All such parameters must use numeric values (i.e., actual byte values). If we send the ASCII number by mistake it will actually give the value of 48 decimal (30 hex) to the parameter, which is wrong.

In the tables given in the following sections ASCII characters are shown as 'A', with single quotes.

8.4 Text Commands

Syntax in the tables below are given in hex, decimal and decimal with ASCII, in that order, one per line.

NOTE The letter “R” in the default column indicates that this state can be saved to non-volatile memory with the “Remember” command.

Table 26: Text Commands

Command	Syntax	Default	Notes
Auto line wrap on	FE 43 254 67 254 “C”	on R	Enables line wrapping (not word wrap).
Auto line wrap off	FE 44 254 68 254 “D”	on R	Disables line wrapping.
Auto scroll on	FE 51 254 81 254 “Q”	off R	Enables scroll at bottom of screen. Text will push display up one line to make room for new line.
Auto scroll off	FE 52 254 82 254 “R”	off R	Disables auto scroll. Text will wrap to top left and overwrite existing text.
Set cursor position	FE 47 [col][row] 254 71 [col][row] 254 “G” [col][row]	n/a	Moves cursor to the specified column and row. The cursor marks the text insertion point in this and all commands.
Send cursor home	FE 48 254 72 254 “H”		This command moves the cursor to the top left of the display area.
Underline cursor on	FE 4A 254 74 254 “J”	off R	Turns on the underline cursor.
Underline cursor off	FE 4B 254 75 254 “K”	R	Turns off the underline cursor.
Block cursor on	FE 53 254 83 254 “S”	on R	Turns on the blinking block cursor.

Command	Syntax	Default	Notes
Block cursor off	FE 54 254 84 254 “T”	R	Turns off the blinking block cursor.
Cursor left	FE 4C 254 76 254 “L”		Moves the cursor one position to the left. If the cursor is already at the beginning of a line it will move to the end of the other line.
Cursor right	FE 4D 254 77 254 “M”		Moves the cursor one position to the right. If the cursor is already at the end of a line it will move to the beginning of the other line.

8.5 Keypad Interface Commands

NOTE The letter “R” in the default column indicates that this state can be saved to non-volatile memory with the “Remember” command.

Table 28: Keypad Interface Commands

Command	Syntax	Default	Notes
Auto repeat mode on	FE 7E [0x00 | 0x01] 254 126 [0|1] 254 “~”[0|1]	off R	Applies to keypad only. 0x00 = 200 ms typematic, 0x01 = key down/key up codes sent.
Auto repeat mode off	FE 60 254 96 254 ““”	off R	Applies to keypad only.

Command	Syntax	Default	Notes
Auto transmit key presses on	FE 41 254 65 254 “A”	on R	Sets auto transmit mode for keypad. Key presses are transmitted to host without polling.
Auto transmit key presses off	FE 4F 254 79 254 “O”	off R	Up to 10 key-presses buffered until polled.
Clear key buffer	FE 45 254 69 254 “E”	n/a	Clear unread key-presses.
Poll keypad	FE 26 254 38 254 “&”	n/a	Returns buffered keypresses to application. Returns 0x00 if no key presses. High order bit set unless this is the last/only key press.
Set debounce time	FE 55 [time] 254 85 [time] 254 “U” [time]	52 ms R	Resolution: 1 = 0.6554 ms [time] is a numeric multiplier.

8.6 Bar Graphs and Special Characters

The commands in this section are used to define and display bar graphs and special characters.

Table 30: Bar Graphs and Special Characters

Command	Syntax	Notes
Initialize thick vertical bar graph	FE 76 254 118 254 “V”	Initialize the user character set to make wide vertical bar graphs.
Initialize thin vertical bar graph	FE 73 254 115 254 “s”	Initialize the user character set to make narrow vertical bar graphs.
Initialize horizontal bar graph	FE 68 254 104 254 “h”	Initialize the user character set to make horizontal bar graphs.

Command	Syntax	Notes
Define custom character	FE 4E [c][8 bytes] 254 78 [c][8 bytes] 254 “N” [c][8 bytes]	Defines one of 8 custom “user” characters. Character number is [c] between 0x00 and 0x07.
Draw vertical bar graph	FE 3D [col][length] 254 61 [col][length] 254 “=”[col][length]	Draws a vertical bar graph at column [col] of length [length]. Length is measured in pixels (0x00 to 0x14). User must first use the “v” or “s” command to initialize characters.
Draw horizontal bar graph	FE 7C [c][r][d][length] 254 124 [c][r][d][length] 254 “|” [c][r][d][length]	Draws a horizontal bar graph starting at column [c] on row [r] with direction [d](0 is right, 1 is left) of length [length]. Length is measured in pixels (0x00 to 0x64 if starting in column 1). User must first use the “h” command to initialize characters.

8.7 Fan and GPO Commands

Table 32: Fan and GPO Commands

Command	Syntax	Default	Notes
General purpose output off	FE 56 254 86 254 “V”	Off	This command turns OFF any of the General Purpose Outputs. [gpo#] is 1 to 6.
General purpose output on	FE 57 254 87 254 “W”	Off	This command turns ON any of the General Purpose Outputs. [gpo#] is 1 to 6.
PWM Value	FE C0 254 192	0	This command sets one of the high power GPOs (GPOs 1-4) into PWM mode.
Return fan RPM	FE C1 254 193	n/a	This command will return a packet to the host with the fan RPM contained in it.

Command	Syntax	Default	Notes
Remember GPO / PWM state	FE C2 254 195	n/a	This command will set the start up state for all the GPOs.
Set PWM base frequency (New in Firmware Revision 1.1)	FE C4 [index] 254 196 [index]	6	This command sets the base frequency for the PWM modulation.
Remember PWM base frequency (New in Firmware Revision 1.1)	FE C5 [index] 254 197 [index]	6	This command will set the PWM frequency for startup.

8.8 Miscellaneous Commands

NOTE The letter "R" in the default column indicates that this state can be saved to non-volatile memory with the "Remember" command.

Table 34: Miscellaneous Commands

Command	Syntax	Default	Notes
Remember	FE 93 [0|1] 254 147	off	Turns the "remember" function on [1] or off [0].
Clear display	FE 58 254 88 254 "X"	n/a	Clears screen of text and graphics, places text cursor at top left.
Set contrast	FE 50 [contrast] 254 80 [contrast] 254 "P" [contrast]	0x80 128	Sets display contrast. Compensates for viewing angle. Contrast is a value between 0 and 255 (hex 0 to FF). Larger = darker.

Command	Syntax	Default	Notes
Set contrast and save	FE 91 [contrast] 254 145 [contrast]	0x80 128	Same as "set contrast" but saves [contrast] as default.
Backlight on	FE 42 [minutes] 254 66 [minutes] 254 “B” [minutes]	on R	Backlight will stay on for [minutes]. If [minutes] = 0 backlight will stay on permanently.
Backlight off	FE 46 254 70 254 “F”	on R	Turns off backlight.
Set backlight brightness	FE 99 [brightness] 254 153 [brightness] 254	0xFF 255	Sets display backlight brightness.
Set backlight brightness and save	FE 98 [brightness] 254 152 [brightness] 254	0xFF 255	Same as “Set backlight brightness” but saves [brightness] as default.
Load startup screen	FE 40 [40 char] 254 64 [40 char] 254 “@” [40 char]	Matrix Orbital LK204-24-USB	Loads new startup screen (40 characters). Screen is remembered for subsequent power ups.
General purpose output off	FE 56 [gpo #] 254 86 [gpo #] 254 “V” [gpo #]	off	Turns a general purpose output OFF. [gpo #] may be from 1 to 6.
General purpose output on	FE 57 [gpo #] 254 87 [gpo #] 254 “W” [gpo #]	off	Turns a general purpose output ON. [gpo #] may be from 1 to 6.
Read module type	FE 37 254 55 254 “7”	see table	Reads the module type.

Command	Syntax	Default	Notes
Set serial number	FE 34 [byte1][byte2] 254 52 [byte1][byte2] 254 “4” [byte1][byte2]		This is a one-time-use command which works only on units without factory set serial numbers.
Read serial number	FE 35 254 53 254 “5”		Reads the two byte serial numbers of the module.
Read version number	FE 36 254 54 254 "6"		Reads the firmware version number of the module.
Enter flow control mode	FE 3A [full][empty] 254 58 [full][empty] 254 ’:’ [full][empty]	Off	Sets “full” and “empty” marks for the 80 byte display buffer. When buffer reaches [full] display will return 0xFE to host. When buffer reaches [empty] display will return OxFF.
Exit flow control mode	FE 3B 254 59 254 ’;’		Turns off flow control.

9 Appendix: Specifications and Options

9.1 Specifications

Table 36: Environmental Specifications

	Standard Temperature
Operating Temperature	0C to +50C
Storage Temperature	-20C to +70C
Operating Relative Humidity	90% max non-condensing

Table 37: Electrical Specifications

Supply Voltage	4.75 = Vdc
Supply Current	9 mA typical
Supply Backlight Current	115 mA typical

Table 38: Optical Characteristics

Number of characters	40 (20 characters by 2 lines)
Matrix format	5x8 with underline
Display area	82.2 x 18.20 mm XxY
Character size	3.20 x 5.55 mm (XxY), not including underline
Character pitch	3.7 mm
Line pitch	5.95 mm
Dot size	0.60 x 0.65 mm (XxY)
Dot pitch	0.65 x 0.70 mm (XxY)
LED Backlight life	100, 000 hours typical
Colour of Illumination	Yellow Green, Ice Blue, Inverse Blue, Inverse Red, Inverse Yellow.

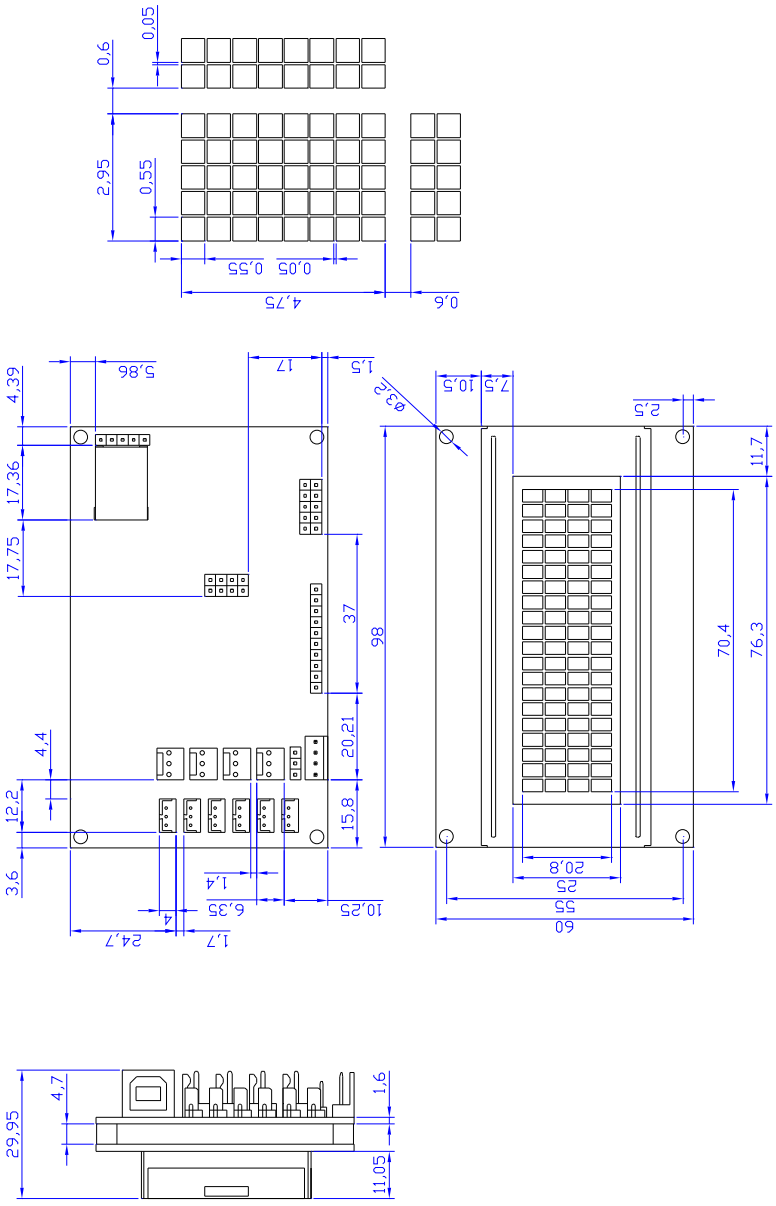


Figure 16: Physical Layout

9.2 Options

Table 39: Options Available on the LK204-24-USB

Inverse Yellow; black backlight with yellow text	-IY
Inverse Blue; blue backlight with white text	-WB
White backlight with grey glass (Ice blue)	-GW
Inverse Red; red backlight with white text	-R

10 Appendix: Glossary

Table 40: Appendix: Glossary

ASCII	American Standard Code for Information Interchange. A 7 bit binary code representing the English alphabet, decimal numbers and common punctuation marks. Also includes control character such as carriage return or end of text. An 8 bit superset of the standard ASCII codes is often used today to include foreign characters and other symbols. These supersets are often called extended ASCII character sets.
Backlight	A backlit display is illuminated from behind to provide nighttime and improved daytime readability.
Baudrate	The (data and signaling) bit transmission rate of an RS-232 device.
Binary Number	A number written using binary notation, which only uses zeros and ones.
Bit	The smallest unit of information a computer can work with. Each bit is either 0 or 1. Binary digit.
Bitmap	A representation, consisting of rows and columns of dots, of a graphics image in computer memory. The value of each dot (whether it is filled in or not) is stored in one or more bits of data.
Byte	A grouping of eight binary bits.
CCFL	Cold Cathode Fluorescent Lamp. A high brightness backlighting source consists of a fluorescent tube powered by a high voltage A.C. source.

Configuration	The way a system is set up, or the assortment of components that make up the system. Configuration can refer to either hardware or software, or the combination of both.
Contrast	The ratio of luminance between the light state of the display to the dark state of the display.
Controller	The micro-controller or PC used to control the Matrix Orbital display unit.
DB-9	The designation of a connector used in the RS232 interface: 9 pin connector
Firmware	Software (programs or data) that has been written onto read-only memory (ROM). Firmware is a combination of software and hardware. ROMs, PROMs and EPROMs and flash EEPROMs that have data or programs recorded on them are firmware.
Font	A design for a set of characters. A font is the combination of typeface and other qualities, such as size, pitch, and spacing.
Font Metric	A definition of where font is to be placed, such as margins and spacing between characters and lines.
Hexadecimal	Refers to the base-16 number system, which consists of 16 unique symbols: the numbers 0 to 9 and the letters A to F. For example, the decimal number 15 is represented as F in the hexadecimal numbering system. The hexadecimal system is useful because it can represent every byte (8 bits) as two consecutive hexadecimal digits. It is easier for humans to read hexadecimal numbers than binary numbers.
Interface	A means by which two systems interact.
LCD	Liquid Crystal Display
Module Type Value	This refers to the model number of the module.
Pixel	The smallest individually controllable element of a display.
Pre-Generated Fonts	Pre-determined fonts which can be downloaded into graphic liquid crystal displays.
Primitive	A low-level object or operation from which higher-level, more complex objects and operations can be constructed. In graphics, primitives are basic elements, such as lines, curves, and polygons, which you can combine to create more complex graphical images.
RS-232	Short for recommended standard-232C, a standard interface approved by the Electronic Industries Association (EIA) for connecting serial devices.

Scroll	To view consecutive lines of data on the display screen. The term scroll means that once the screen is full, each new line appears at the bottom edge of the screen and all other lines move up one position.
Serial Number	A number that is one of a series and is used for identification of the module.
Serial Port	A port, or interface, that can be used for serial communication, in which only 1 bit is transmitted at a time.
Version Number	This refers to the firmware revision number of the module.
Volatile Memory	Temporary memory. Once the power supply is turned off volatile memory is then erased.