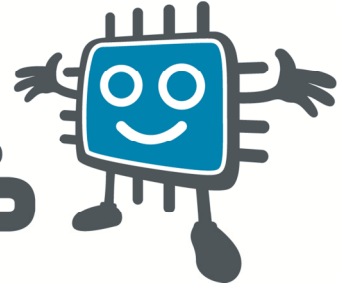


# embedded adventures

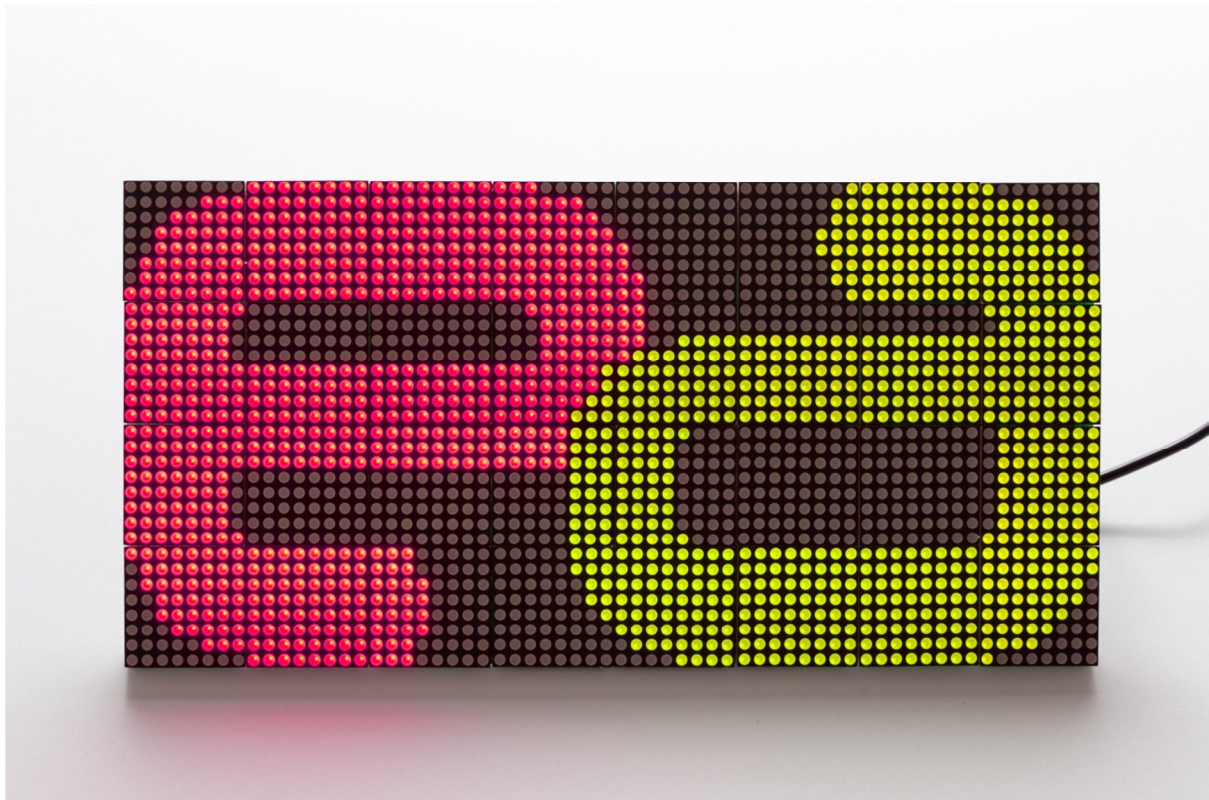


Device: LDP-6432

Document Version: 1.1

Date: July 2010

Description: 64x32 1R1G LED Display Panel



## Table of Contents

Introduction.....	3
Nomenclature .....	3
Connections.....	3
Power .....	3
Pinouts.....	4
Controlling the display .....	4
Using the display .....	5
Multiple displays .....	6
Tricks and traps .....	6
Versioning .....	7

## Introduction

---

The LDP-6432 is a dual-LED, tri-colour 64 pixel by 32 pixel LED matrix display. Each pixel consists of a red and green LED (commonly referred to as "1R1G"). There are 2048 pixels and 4096 individually addressable LED elements. It includes shift-register circuitry such that all 4096 LEDs can be controlled with only 12 microcontroller I/O lines. Pixels can show as black (that is, completely unlit), red, green or orange (red+green).

The LDP-6432 requires dynamic scanning (1/16) to activate the display. As such, customers may find it easier to use the PLT-1001 series of driver platforms available from Embedded Adventures to drive the LED panel, or use the Draw PicPack library.

## Nomenclature

---

*Set* in this document means to pull the connection logic HIGH (Vcc). *Clear* in this document means to pull the connection logic LOW or ground.

## Connections

---

The LDP-6432 has three connections on the back panel. As you look at the back panel, these are:

J1	Input data port (left hand side)
J2	Output data port (right hand side)
J3	Power / ground connection (centre terminal)

## Power

---

This panel runs at strictly 5v (+/- 0.5v). The display does not need to run on the same supply as your controller or driver board, however, the must have the same ground connection. This can be achieved by connecting your controller board to the ground connection in the J1 or J2 ports.

LEDs consume large amounts of current, and large amounts of LEDs obviously multiplies this affect. Although the board actually only displays one row at any time, at 30mA each LED, this can still result in 256 LEDs (two fully lit rows) consuming nearly more than 4 Amps.

## Pinouts

---

J1/J2 pinout as viewed from above

GND	<b>1</b>	<b>2</b>	A
GND	<b>3</b>	<b>4</b>	B
GND	<b>5</b>	<b>6</b>	C
$\overline{\text{EN}}$	<b>7</b>	<b>8</b>	D
$\overline{\text{R1}}$	<b>9</b>	<b>10</b>	$\overline{\text{G1}}$
$\overline{\text{R2}}$	<b>11</b>	<b>12</b>	$\overline{\text{G2}}$
GND	<b>13</b>	<b>14</b>	L
GND	<b>15</b>	<b>16</b>	S

Note that the back of the display does not indicate the pins exactly as described here, however we have shown the pins in this datasheet with overline to clearly indicate they are active LOW.

## Controlling the display

---

To display a particular row, the pins A,B,C,D are used as follows:

<b>D</b>	<b>C</b>	<b>B</b>	<b>A</b>	<b>Row Enabled</b>
0	0	0	0	0 (top row) and 16
0	0	0	1	1 and 17
0	0	1	0	2 and 18
0	0	1	1	3 and 19
0	1	0	0	4 and 20
0	1	0	1	5 and 21
0	1	1	0	6 and 22
0	1	1	1	7 and 23
1	0	0	0	8 and 24
1	0	0	1	9 and 25
1	0	1	0	10 and 26
1	0	1	1	11 and 27
1	1	0	0	12 and 28
1	1	0	1	13 and 29
1	1	1	0	14 and 30

## LDP-6432 datasheet

---

1	1	1	1	15 and 31 (bottom row)
---	---	---	---	------------------------

To set pixels within a row, the following pins are used:

$\overline{R1}$	Data for Red LED in pixel in upper 16 rows
$\overline{G1}$	Data for Green LED in pixel in upper 16 rows
$\overline{R2}$	Data for Red LED in pixel in lower 16 rows
$\overline{G2}$	Data for Green LED in pixel in
$\overline{S}$	Shift data
L	Latch data

Note that these lines are active low, so set to 0 for lit LED, 1 for dark led,

$\overline{R1}$  and  $\overline{G1}$  reference a pixel in the upper 16 rows.  $\overline{R2}$  and  $\overline{G2}$  reference a pixel in the lower 16 rows.

To enable the display:

$\overline{EN}$	Enable display (active low)
-----------------	-----------------------------

### Using the display

---

The display is designed to show two rows of LEDs of the 32 rows available at any one time. Once those rows have been displayed, the next two must be shown, and so on. The panel has protective circuitry that prevents the display from being enabled entirely if it is being cycled too slowly. It is possible to run it slow enough to see the cycling of the rows if you wish, but do be aware that you can't simply output one row and then expect to see that displayed. You must be cycling through all 16 row pairs (although, of course, you don't have to be actually lighting any pixels on any of those rows).

The display can be considered as a pixels 0-63 (x direction, 0 being the left-most column) and pixels 0-31 (y direction, 0 being the top-most row). As such, in order to display a single row:

Set/Clear pins  $\overline{R1}$  and  $\overline{G1}$  for the upper 16 rows as appropriate (pull low to display pixel)

Set/Clear pins  $\overline{R2}$  and  $\overline{G2}$  for the lower 16 rows as appropriate (pull low to display pixel)

Pulse S low to shift in these pixels  
(repeat until all 64 pixels have been shifted in)

Note that pixel x=0 is shifted in first, pixel x=63 is shifted in last.

Once an entire row is shifted in:

Set  $\overline{EN}$  to turn off display  
Set/Clear A,B,C,D to select the rows  
Latch the data by pulsing L (set, then clear)  
Clear  $\overline{EN}$  to turn on the display

To control brightness, PWM may be used on  $\overline{EN}$ , which will also reduce the current consumption.

### Multiple displays

---

The data ports may be daisy-chained from one display to the next – note that all pins are shared except  $\overline{R1}$ ,  $\overline{G1}$ ,  $\overline{R2}$  and  $\overline{G2}$ , which are the connected to the output of the (effective) 64 bit shift registers. This means that if two displays are connected together, they can be treated like one display with 128 pixel rows. The displays don't need to be physically placed vertically, of course. You can create a 64x64 pixel display, or a 128x32 pixel display.

### Tricks and traps

---

The LED panel display will consume a lot of current if you display large quantities of pixels. So much so that depending on your power supply, you may see a considerable voltage drop across the supply. If this goes below the specification for your microcontroller, you may cause it to behave erratically. Even if you are within specification, the brown-out reset functionality of your microcontroller may be tripped (see the config fuses for your microcontroller for details). In this situation we suggest you run the panel on a different supply.

Generally we don't recommend lighting seriously large quantities of LEDs all at the same time – it really does use a lot of current (4 amps+). As such, please make sure all wiring to your power supply can handle this sort of load. Or better yet, use some form of PWM on  $\overline{EN}$  (or simply not leaving the row enabled for the entire time before the next row). This will reduce brightness of the LEDs, increase life-span of the panel, reduce power usage and reduce voltage drop.

This panel requires a 5 volt supply. You will damage the display irreparably if you put a higher voltage through it.

### Versioning

---

Version 1.1 – July 2010

Corrected enable line  $\overline{EN}$  description in "Using the display" section.  $\overline{EN}$  needs to go high while changing ABCD etc, then back to low to enable display again.