Pixblasters MS1 Video LED Controller User's Guide

July 2024
(FPGA ver 110, SW ver 106)

1 INTRODUCTION

The Pixblasters® MS1 Video LED Controller connects to any computer and any operating system as an ordinary monitor to display the user-selected portion of the monitor image on up to 16,384 RGB LEDs at 60 frames per second (FPS). This controller enables digital signage and smart lighting professionals, and even DIY enthusiasts with minimal technical skills, to turn a bunch of addressable RGB LED strips or LED panels (matrices) into immense video LED displays. The Pixblasters MS1 controller enables new levels of professional-grade digital signage that cannot be attained by standard LED modules. The LED strips can be curved and glued to different surfaces to form giant yet economically viable video installations that can span entire buildings. The Pixblasters MS1 controller is often used in smart lighting installations with multiple and heterogeneous sensors, i.e. ultrasound occupancy sensors, which require real-time controls from a central monitoring computer.

> ℹ️ Multiple Pixblasters MS1 controllers can be easily chained to drive immense video displays built of hundreds of thousands of perfectly synchronized LEDs.
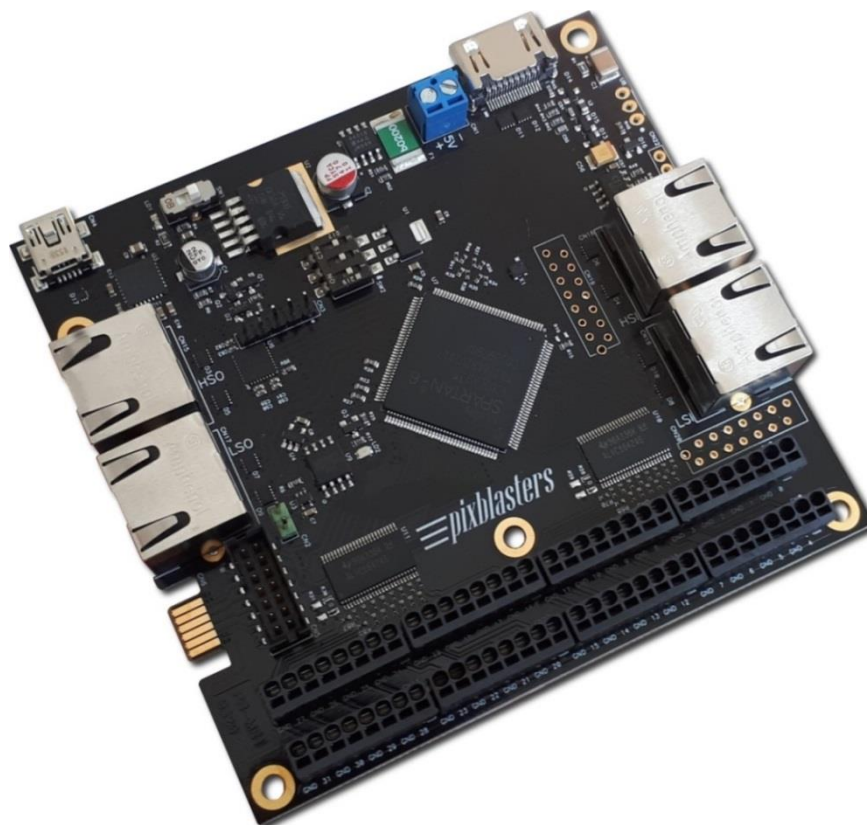
Figure 1.1 The Pixblasters MS1 Video LED Controller (ver 0520)

The Pixblasters MS1 displays any visual content with absolutely no programming required and with no burden on the driving computer that is free to run digital signage players, media players, and other software at the full speed. The LED displays controlled by the Pixblasters MS1 can be remotely managed anywhere in the world by virtually any digital signage software. With the MS1 LED controller, even those with minimal technical skills can build giant LED displays with no soldering and by using only simple tools like pliers, wire cutters and screwdrivers.

- 6 -

1.1 Display Video from Any Computer

The Pixblasters MS1 video LED controller connects to any computer as an ordinary monitor and puts no burden on the driving computer, which is free to run digital signage players, media players, and other video software at full speed. The LED displays controlled by the Pixblasters MS1 can be remotely managed by the digital signage software of your choice from anywhere in the world.
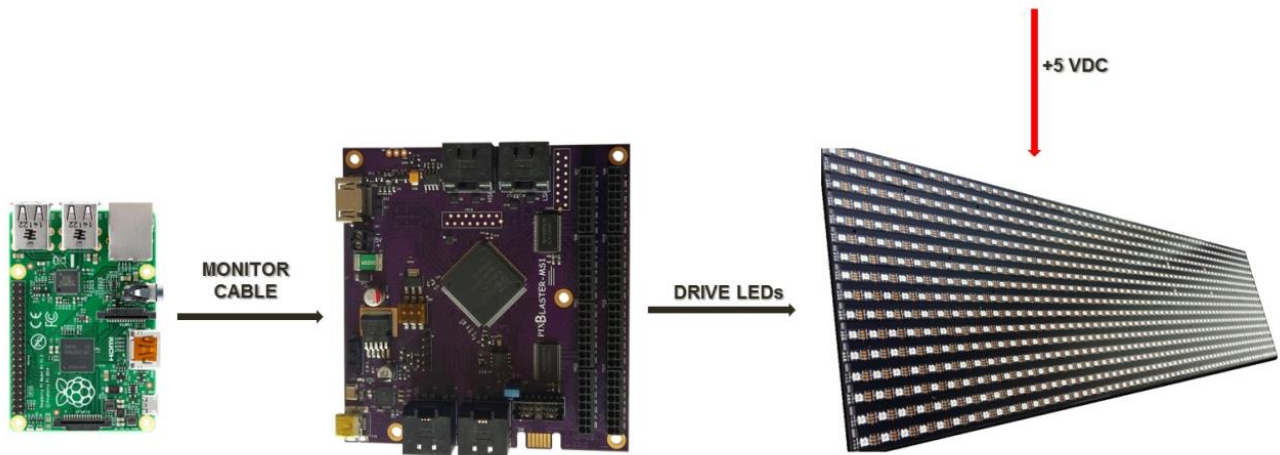
Figure 1.2 Pixblasters Controller Inserts between the Video Source and the LED Matrix

The Pixblasters Video LED Controller has been designed to support professional-looking and easy to use video LED displays built of addressable RGB LEDs (strips or panels). To display any graphics and video with no limitations, Pixblasters users just need to plug in the monitor cable. Stills and animations, AVI, FLV, WMV, MP4, etc. - what you see on the computer monitor connected to the selected computer is what you get on the Pixblasters MS1 controller-driven LED display with absolutely no coding required!

Image part for the LED Display enclosed by the red line
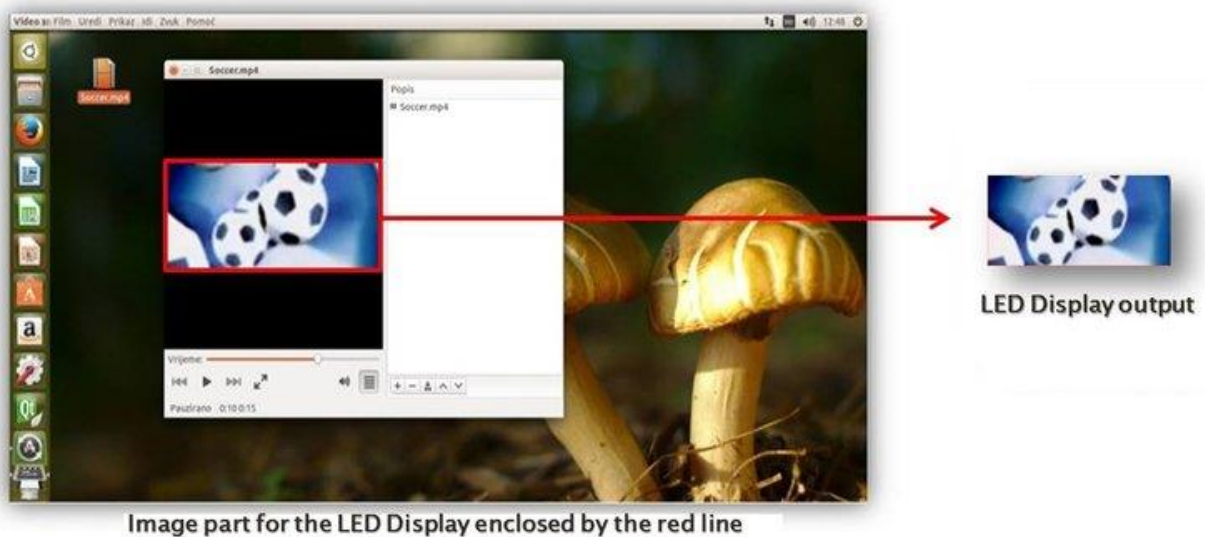
Figure 1.3 Selecting and Cropping of the Image Part for the LED Display

A single Pixblasters Video LED Controller supports up to 16,384 LEDs that can be arranged in LED display matrices with various resolutions. Although respectable, such LED driving capacity allows for the LED display of a smaller portion of standard monitor image. The Pixblasters LED controller receives the video formatted for the monitor, crops the image part selected (see **Video Input Cropping**) for the LED display (**Figure 1.3**), and drives the RGB LED strips or panels by properly formatted video data. The complexity of LED driving is hidden from the driving computer.

To keep the promise of supporting professional and easy-to-use video LED displays, the Pixblasters LED integrates several important features. For example, it includes an actual EDID chip to smoothly connect to any operating system as a standard monitor. Consumer monitors use an EEPROM chip called the EDID. It contains monitor's name and technical info about its capabilities. This information enables any standard video source, i.e. the PC computer, to properly detect the monitor and automatically format the video image suitable for the specific monitor.

The Pixblasters really shines in big video installations, which may be oversized and hardly portable. Such installations accept only minimal maintenance and must work automatically when powered. Consequently, the LED controller requires retentive memory to store important configuration data, such as the coordinates of the image window for cropping. It also needs a "house-keeping" device to take care on SRAM-based FPGA chip with no internal storage for its configuration data, PC connectivity, multi-controller display configurations, etc. All these functions are integrated in on-board Microchip PIC18F26J50 microcontroller that connects to the PC and runs a simple User's Interface. Besides the display configuration, it also enables firmware updates with new features.

1.2 System Connections

The Pixblasters Video LED Controller is designed to be expandable and to support really big video LED displays. While a single Pixblasters MS1 video LED controller drives up to 16,384 LEDs, multiple daisy-chained Pixblasters MS1 controllers can potentially drive hundreds of thousands of LEDs. To provide the simplest possible video display's architectures, each Pixblasters controller can operate either in Master or Slave mode.
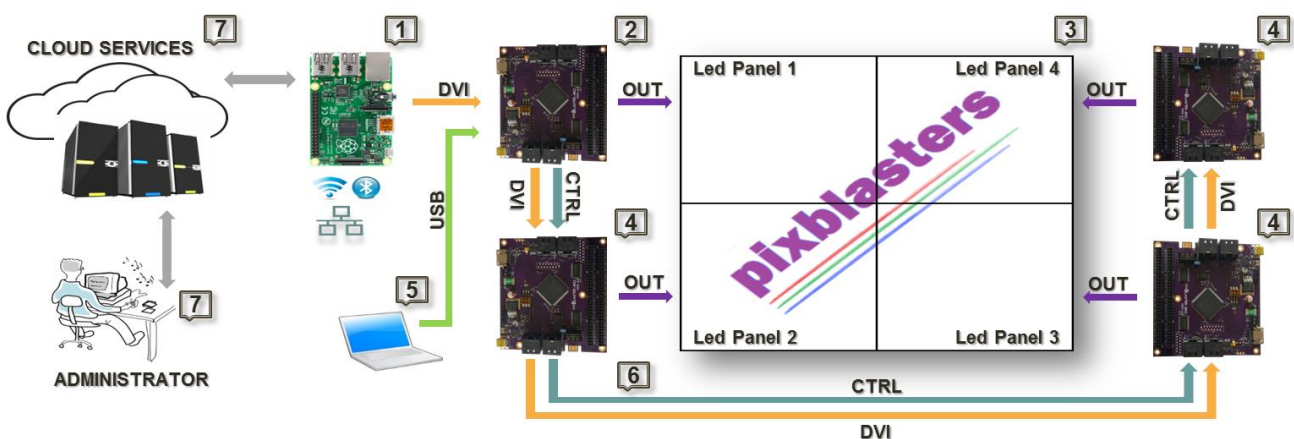
Figure 1.4 Architecture of Video LED Display Driven by Daisy-chained Pixblasters Controllers

Figure 1.4 explains the Pixblasters daisy-chain system connections and an example architecture of 18.8 m^2 (202 ft^2) video display built of 60 LEDs/meter addressable RGB LED strips. Paragraph **Multi-controller Demo** Display Architecture showcases the Pixblasters demo display built with multiple LED controllers.

- Control computer (1) generates video for the complete LED display, and with the standard monitor cable, connects to the DVI compatible video input of the Master Pixblasters MS1 video LED controller (2).

- Optionally, the computer (1) can be locally or remotely controlled through different network interfaces. It can run any video display application, including digital signage players that automatically connect to online digital content services (7).

- The Master Pixblasters controller (2) controls Slave controllers (4) through the DVI video link. Each slave controller displays the exact portion of the video image on the attached LED panels (3) built of up to 16,384 RGB LEDs.

- The video (6) and the control links* between the chained Pixblasters MS1 controllers use ordinary Ethernet cables to enable simple and flexible display wiring. The links guarantee automatic and perfect LEDs synchronization.

- Each controller (2, 4) is configured for Master/Slave mode by on-board DIP switches, or optionally through the configuration menu. Other configuration parameters are setup only once for each board through a simple configuration menu on the PC (5) connected with the USB cable.

- Display's configuration parameters, such as LED timing parameters and the video input image part for display on the attached LEDs, are permanently stored in the on-board microcontroller that also controls the LED system's boot-up.

- Separate +5V DC power supplies, which are not shown in **Figure 1.4**, must provide sufficient power for the complete display.

* Future expansion; currently unsupported

1.3 Compatible LED Materials

The Pixblasters MS1 supports addressable RGB LEDs that comes in the following material forms suitable for video displays:

- 3-wire LED strips, such as WS2811, WS2812B…

- 4-wire LED strips, such as APA102, SK9822, HD107S…, and

- WS2812B compatible LED panels with different horizontal and vertical resolution, such as 8x8, 16x16, 32x8, 22x22, 44x11 and other custom formats

1.4 Flexible Enough for DIYers, Powerful Enough for Professionals

Figure 1.5 Pixblasters Demo Made Using LED Strips Glued to Aluminum Plates

The Pixblasters Video LED Controller is flexible enough to enable DIYers and developers ideas and at the same time, to enable established professionals to take their signage business to the next level. With the Pixblasters controller, it is finally possible to make professionally looking LED displays with addressable LED strips, without any of usual constraints on the size, graphics and video content, display management, storage capacity, or other limitation common to the competing LED controllers. The Pixblasters video LED controller enables DIY makers to tackle any LED signage project in a cost-effective and flexible way - with no programming, no soldering, and with no more than a simple screwdriver.

The possibilities are endless; here are just a few we thought of to get you started:

- Build a scoreboard for your school gym
- For musicians - incorporate a bold light feature into your stage set
- Holiday home decorations - elevate and customize!
- Artistic light installations or exhibitions
- Entrepreneur showing off a new idea or project
- Smart lighting systems that adapts to changing ambient and occupancy conditions in real-time

Signage professionals can use the Pixblasters MS1 Video LED Controller in many different applications. Small signage shops that make non-illuminated signs can expand their portfolio and offerings by incorporating the Pixblasters controller into their existing production workflow. Adding video LEDs can take any sign to the next level and make it more memorable.

Pixblasters can also be implemented (with small changes) within housings of existing illuminated signs. The installed video signs can be remotely controlled, so companies have more control over what kind of digital content they're displaying to their customers, and can make any changes needed quickly. Using the special LED strip's features to deploy very large and very cost-effective display installations will upgrade and improve your professional signage.

1.5 Easy Video Content Management

Pixblasters makes it easy to manage your video content. Instead of browsing the Internet in search of code snippets for specific LED patterns or odd software libraries that require guru-level software skills to format into an LED display image, Pixblasters users just need to plug in the monitor cable to display any graphics and video with no limitations. It also means that the Pixblasters MS1 enables user-defined custom software – whatever works for the regular monitor, works for LEDs as well.
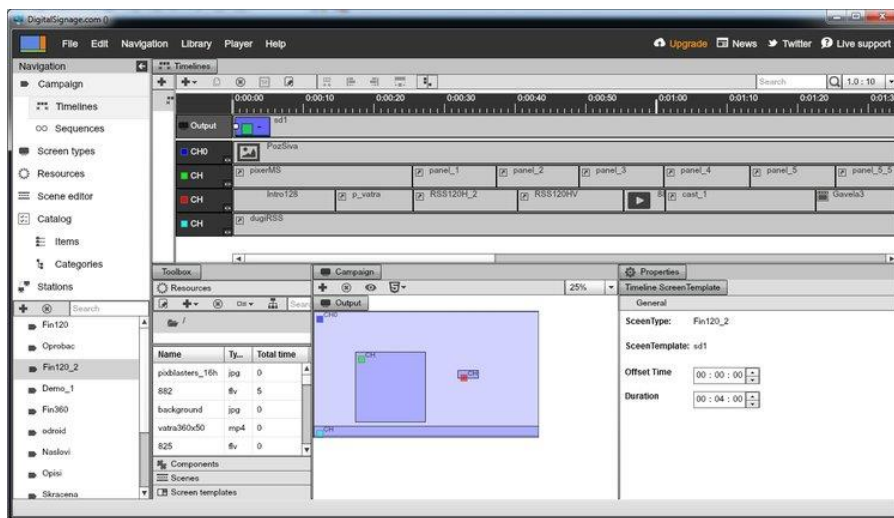
Figure 1.6 Example Digital Signage Software from MediaSignage, Inc. (aka DigitalSignage.com)

Stills and animations, AVI, FLV, WMV, MP4, etc. - what you see on the computer monitor connected to the selected computer is what you get on the Pixblasters MS1 controller-driven LED display with absolutely no coding required!

The LED displays crops the input video image and shows a portion of the original video.
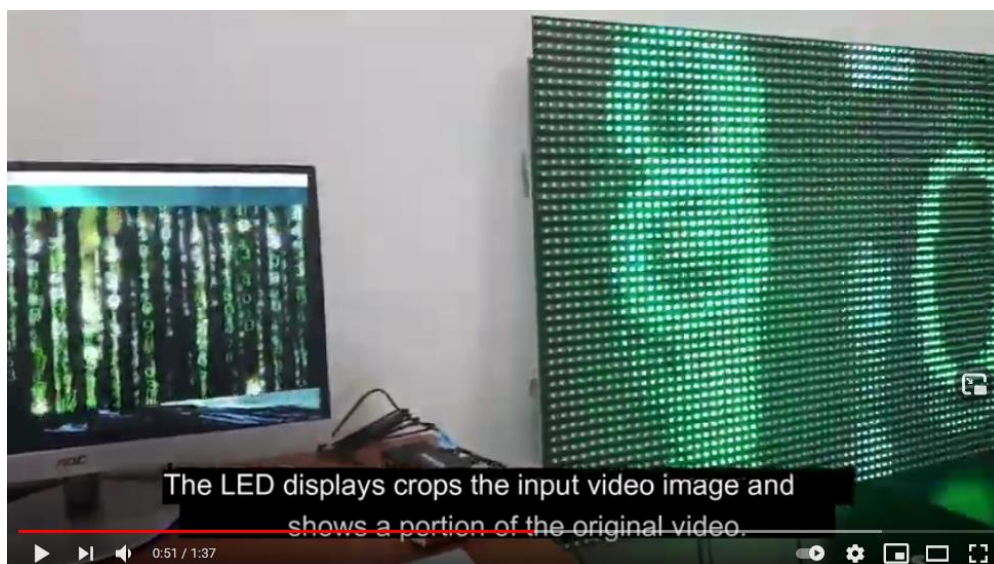
Figure 1.7 Demo Clip with the Raspberry Pi

> ℹ This video clip shows the Pixblasters driven LED display and the Raspberry Pi Omxplayer example: https://youtu.be/DXew5YJgVcU

Pixblasters users can generate the LED display content in many different ways. For example, one can use the Windows Media Player animated visualizations to play music synchronized with video effects, or the Raspberry Pi Omxplayer video player to play MP4 video clips on the LED display. Those with special display applications (such as big real-time timetables and scoreboards) can develop custom software like graphics for a computer monitor.

The Pixblasters MS1 controller works with virtually any digital signage software and allows users to operate and manage display content at the same level as high-end professional LED displays. This makes the generation of display content very easy. Either open source or fully commercial, digital signage software lets users quickly define scheduled playlists, combine video and graphics, incorporate news feeds and social networking posts, etc.

Additionally, users can define the operating hours of a remote display, check what is currently playing, change the stored signage campaign on a single remote displays, or update multiple connected displays. Quality software enables the preview function and local campaign checkup prior to downloading to the video LED installation.

1.6 Key Features

- **Easy to use with any computer, any OS**:

  Pixblasters connects to any computer as an ordinary external monitor, with no programming needed. You can use it with PC, Raspberry Pi, media boxes, phones, etc. Simply plug in the monitor cable and the controller will smoothly drive the LEDs at 60fps or higher frame-rates

- **Straightforward display content and management:**

  Anything from the selected part of the monitor image shows on the LEDs as well. No need for special software for anti-aliasing or fonts. Use any digital signage software and add layered screen divisions, text, animations, video, RSS... LEDs work at their maximum speed and are perfectly synchronized, no matter the number of LEDs and chained MS1 controllers, or the display size

- **Supports LED strips and LED panels**

  Various LED material can be used for Pixblasters MS1 controlled video LED displays: 3-wire (WS2811, WS2812B…) and 4-wire (APA102, SK9822, HD107S…) LED strips, and WS2812B LED panels: 8x8, 16x16, 32x8, 22x22, 44x11 and other custom formats

- **Display size and resolution:**

  A single MS1 can control 4.7/17.7[*] m² (~ 50.6/190.5 ft²) while daisy-chained MS1s can control an LED display of hundreds of square meters using thirty to sixty LED strips. Daisy-chained controllers that drive 16,384 LEDs can support HD LED displays

  Calculation based on LED strips with 60 LEDs/m and 30 LEDs/m with an assumed rectangular pixel pitch

- **Remote controllable and customizable:**

  Control it remotely from anywhere in the world through the network interfaces of the driving computer. Do not worry about the LED display's architecture. Users with special display

applications (such as big, real-time timetables and scoreboards) can develop custom software for a computer monitor

- **Open source FPGA demo:-**

Use the hardware platform for experimenting with, and learning about, driving RGB LEDs.

> ℹ️ The open source FPGA demo designs support a subset of the fully-featured Pixblasters MS1 LED controller. Source code is available from the Pixblasters GitHub. Demo explanations and links to the code can be found here:
>
> 3-wire WS2812B Example: https://www.hackster.io/pixigreen/convert-a-bunch-of-led-strips-into-a-big-video-display-c90380
>
> 4-wire APA102 Example: https://www.hackster.io/pixigreen/apa102-rgb-led-fpga-video-controller-471c29

1.7 Technical Characteristics

- **Xilinx® Spartan®-6 XC6LX9-3 FPGA chip**

    - Max number of different FPGA configurations is seven (7). Each configuration supports a specific LED type and mode of operation.
    - Delivered configurations include i.e. 32-outputs WS2812B master controller, 16-outputs APA102 slave controller, LED panels controller, etc. – see
    - **Table** 12.1
    - Pixblasters offers configuration updates through company's website

> ℹ️ For FPGA updates and updates of LED pixel mapping matrices, please visit:
> https://pixblasters.com/deliverables/

- **On-board Microchip PIC18F26J50 microcontroller:**

    - The microcontroller connects to the PC, runs the configuration software, and permanently stores the configuration data. It also enables simple firmware updates with any new features.

- **Output resolutions (H x V)**

    - Controller supports various resolutions through its segmented and non-segmented operation modes for LED strips and the LED panels operation mode
    - Example resolutions include 512 x 32 (native), 256 x 64, 128 x 128, 180 x 96, etc.

- **LED details:**

    - 32 LED digital outputs: max. 512 LEDs per output in 3-wire mode and max. 1024 LEDs per output in 4-wire mode
    - Maximum WS2812B/SK6812 LED display frame rate is 60 fps
    - Maximum APA102 LED display frame rate is 60 fps

- – Supported 3-wire LED types: WS2811, WS2812, WS2812B, WS2815, SK6812, GS8208B…
- – Supported 4-wire LED types: APA102, HD107S, SK9822 …
- – Supported WS2812B LED panels: 8x8, 16x16, 32x8, 22x22, 44x11 and other custom formats
- – Supports RGB666 (256K colors) color format
- – Maximum driving capacity of the single MS1 controller is 16,384 RGB LEDs
- – 5 VDC 24 mA digital outputs compatible with the most popular RGB LED types

- **Connectivity:**

  - – DVI (HDMI connector) video input allows for easy connection from the computer - it also works with adapter cables should your computer have a DVI® output
  - – Push-in strip power and signal connectors enable easy wiring without soldering
  - – Validated with different computers and different OS's: Raspberry Pi Linux, Microsoft® Windows® PC, Apple macOS, media boxes, phones and tablets…
  - – On-board EDID flash enables automatic connection to any computer and any OS

- **Display controls:**

  - – Cropping image window
  - – Display formatting
  - – LED timing parameters
  - – Supports different input video resolutions (max. 720p - 1280 x 720 at 60 fps)
  - – Integrated video input cropping and mirroring
  - – Multiple lines per output mode
  - – LED gamma correction
  - – Brightness controls

- **Daisy-chain multiple controllers:**

  - – Each MS1 board can work either as the Master or the Slave video controller
  - – Up to 48 controllers chained together
  - – Daisy-chained controllers can drive more than 200,000 perfectly synchronized LEDs differently arranged in HD LED displays
  - – The high-speed video link between MS1 boards assures max. fps and synchronicity
  - – low-speed control link interface (not supported; future expansion)
  - – Video and control links use common UTP cables (Ethernet)
  - – On-board DIP switches allow for an easy selection of the Master, Slave and six other FPGA configurations, including user-defined FPGA configurations
  - – A single digital output has the ability drive multiple output video horizontal lines
  - – Configuration is fully customizable through a menu on the PC connected via the USB serial cable

- **Power and General Info:**

  - – Protected 5 VDC ±5% power supply input
  - – The LED power supplies need to be wired separately to fulfill the power and current requirements of giant LED displays
  - – Board dimensions: 95 x 100 mm
  - – Power supply voltage: 5V ±5%
  - – Current consumption: Max 0.5A,
  - – Operating temperature: -40°C to +85°C

1.8 Pixblasters Support

For more information on the Pixblasters MS1 controller, please visit our official website www.pixblasters.com or contact us at info@pixblasters.com. Follow us on social media:

Facebook: https://www.facebook.com/Pixblasters/
Instagram: https://www.instagram.com/pixblasters/
GitHub: https://github.com/PixiGreen/Pixblasters-MicroDemo
YouTube: https://www.youtube.com/@pixblasters

Our YouTube channel brings comprehensive and illustrative how-to video clips that can help you to quickly start building great LED display installations: http://pixblasters.com/videos/

Figure 1.8 Example Pixblasters Video Clip

2 LED STRIPS DISPLAY - QUICK START GUIDE

Before the first use, or in a case of a significant change of the LED display controlled by already initialized Pixblasters LED controller, the MS1 Video LED controller must be initialized through the embedded user's interface.

The Pixblasters MS1 Video LED Controller controls very diverse LED display configurations and its control electronics must be programmed with the exact display's resolution, display's architecture (non-segmented, segmented, clock speed), and with the exact part of the input video for the LED display. Additionally, by means of on-board DIP switches, the MS1 Video LED Controller must be configured for either master or slave operation modes

The complete initialization process for use with WS2812B LED strips is described in this short video clip: https://youtu.be/6BcOaJHcRD8.

> ⚠️ Please make sure to use the DIPs combination 001 instead of the 011 combination shown in the video clip. Some Pixblasters boards, based on the firmware version, may miss the 011 FPGA configuration and behave differently from the filmed board.
>
> However, the described process is the same for any DIP-selected FPGA configuration.
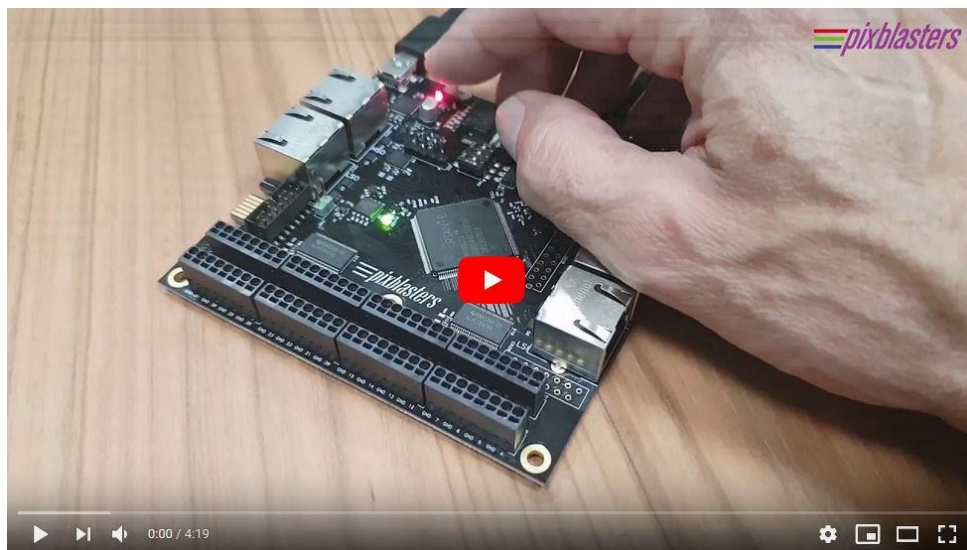
Figure 2.1 Quick Start Guide - Click on Photo to Play Video

Initialization is very simple and can be done within a couple of minutes. It is also permanent, since the initialization parameters stay permanently stored in on-board memory. In case that the LED video display's architecture, such as the number of lines (pixels rows) does not change during its lifetime, the Pixblasters MS1 Video LED Controller's initialization must be executed only once!

> ℹ️ Quick start guide is fully described in the PAPP002 application note:
> http://pixblasters.com/wp-content/uploads/2021/05/papp002-Pixblasters-Quick-Start-Guide_v1.1.pdf

3   LED PANELS DISPLAY - QUICK START GUIDE

Before the first use with LED panels, or in a case of a significant change of the LED display controlled by already initialized Pixblasters LED controller, the MS1 Video LED controller must be initialized through the embedded user's interface.

The Pixblasters MS1 Video LED Controller controls very diverse LED display configurations and its control electronics must be programmed with the exact display's resolution, display's architecture (type and resolution of LED panels…), and with the exact part of the input video for the LED display. Additionally, by means of on-board DIP switches, the MS1 Video LED Controller must be configured for either master or slave operation modes

The complete initialization process for use with WS2812B LED panels is described in this short video clip: https://www.youtube.com/watch?v=uhBz0K_IGW8.
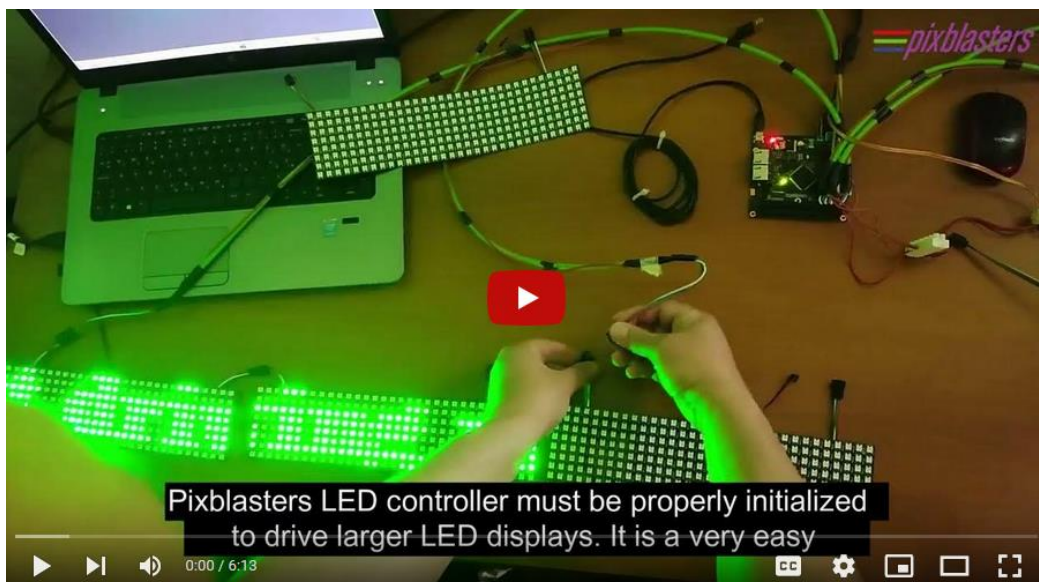
Figure 3.1 Quick Start Guide - Click on Photo to Play Video

Pixblasters MS1 controller's initialization for LED strips and LED panels is very similar. It can be done within a couple of minutes. The initialization parameters are permanently stored in on-board memory.

> ℹ️ Quick start guide is fully described in the PAPP005 - Pixblasters MS1 Controller - Quick Start Guide with WS2812B RGB LED Panels: https://pixblasters.com/wp-content/uploads/2024/07/papp005-Pixblasters-Panels-Quick-Start-Guide_v1.0.pdf

.

> ℹ️ To learn how to build big video LED displays with RGB LED Panels, please read PAPP006 - Pixblasters MS1 Controller - How to Build Video Displays with RGB LED Panels: https://pixblasters.com/wp-content/uploads/2024/07/papp006-Pixblasters-Build-VideoDisplay-with-LED-Panels_v1.0.pdf

4   BOARD OVERVIEW

Figure 4.1 shows elements of the Pixblasters MS1 Video LED Controller board.
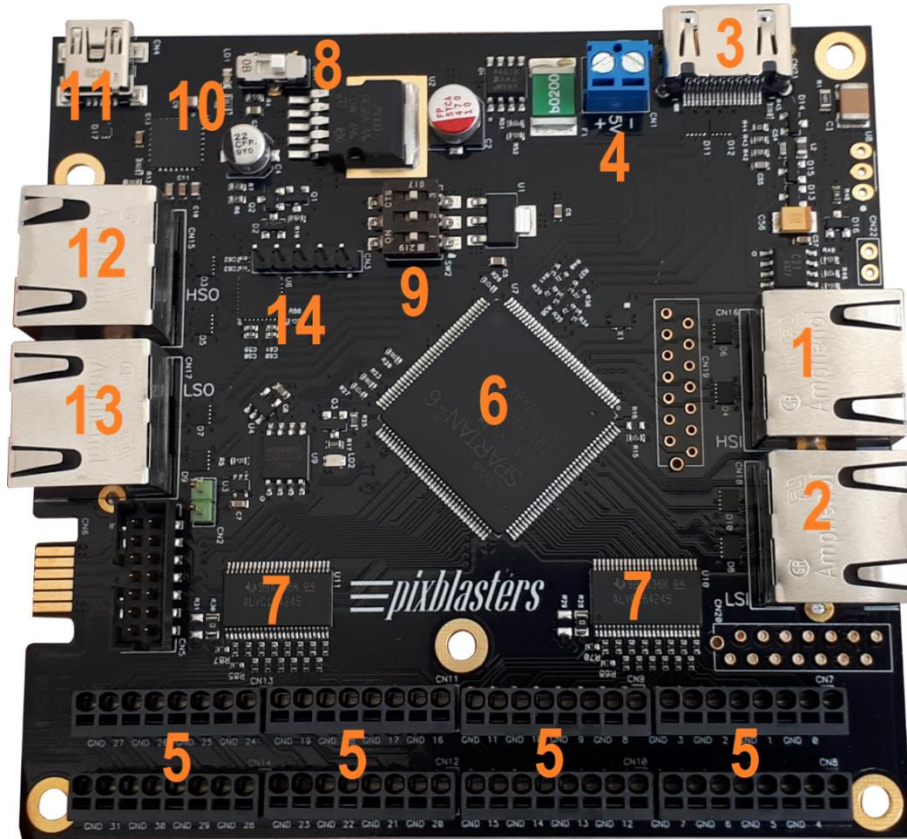
Figure 4.1 Pixblasters Controller Board Elements

1. High-Speed Input (HSI) port
2. Low-Speed Input (LSI) port
3. HDMI input
4. Power supply input
5. LED output ports
6. FPGA
7. LED drivers
8. ON/OFF switch
9. FPGA configuration select
10. PIC µC (microcontroller)
11. USB port
12. High-Speed Output (HSO) port
13. Low-Speed Output (LSO) port
14. LVDS Buffer/Repeater with Pre-Emphasis (from PCB revision 0520). Older 1019 PCB revision does not include this LVDS chip.

4.1 High-Speed Input (HSI) Port

In a system with multiple Pixblasters controllers chained together, the HSI port receives the video signal from the neighboring "upstream" device. For more info, please go to the chapter **Multi-Controller Topology**.

4.2 Low-Speed Input Port

In a system with multiple Pixblasters controllers chained together, the LSI port receives the control signals from the neighboring "upstream" device. LSI port functionality is currently not supported. For more info, please go to the chapter **Multi-Controller Topology**.

> *i* Future expansion. Currently not used.

4.3 HDMI Video Input Port

This is the main, DVI-compatible video input port that uses the HDMI compatible connector. In a system comprised of multiple Pixblasters controllers chained together, the main video signal source (**Figure 1.4**) must be connected to this video input port of the uppermost device (Master board). For more information, please read the chapter **Multi-Controller Topology**.

4.4 Power Supply Input

The main 5V power supply input for the Pixblasters MS1 controller.

> ⚠ To prevent the short circuit, please respect the PCB voltage markings.

4.5 LED Output Ports

These are wiring connections for LED strips and LED panels. While the ground (GND) connections are available on these ports, the power must be connected outside of the Pixblasters board. It is necessary due to possibly very high power requirements that surpass conducting capabilities of electronics PCB boards.

4.6 FPGA

Xilinx XC6SLX9 Spartan-6 FPGA chip configured with custom Pixblasters designs.

4.7 LED Drivers

SN74ALVC164245DGGT FPGA LED driver ICs.

- 19 -

4.8 ON/OFF Switch

This is the main Pixblasters MS1 board's power switch.

4.9 FPGA Configuration (Operation Mode) Selector

The Pixblasters MS1 Video LED Controller boards include several pre-programmed FPGA configuration files stored in the on-board non-volatile flash memory. Use this switch (**Figure 12.2**) to select FPGA configuration to load into the FPGA chip. Different FPGA configurations are used to support different kinds of LED strips. Up to 7 (1 - 7) different configurations may be used. Any of these 7 configurations may be updated by user.

> Configuration '0' serves for updates of stored FPGA configurations and cannot be modified by user. This configuration is also used to program pixel mapping matrix for LED panels in the on-board memory reserved for FPGA configuration No 7 (111).

4.10 PIC Microcontroller

The Microchip PIC18F26J50 microcontroller connects to the PC, runs the configuration software, and permanently stores the configuration data.

> The original configuration of the on-board Microchip PIC18F26J50 microcontroller **MUST NOT** be changed under any circumstances and any change will permanently cease the operation of the Pixblasters firmware. The PCB board and all user-defined FPGA designs will continue to work with no limitations!

4.11 USB Port

The USB device port enables Pixblasters controller connection to the PC USB host port.

4.12 High-Speed Output (HSO) Port

In a system comprised of multiple Pixblasters controllers chained together, the HSO port sends the video signal to the connected neighboring "downstream" device. For more info, please go to the chapter **Multi-Controller Topology**.

4.13 Low-Speed Output (LSO) Port

In a system comprised of multiple Pixblasters controllers chained together, the LSO port sends control signals to the connected neighboring "downstream" device. The LSO port functionality is currently not supported. For more info, please read the chapter **Multi-Controller Topology**.

> Future expansion. Currently not used.

4.14 LVDS Buffer/Repeater with Pre-Emphasis

Starting from Pixblasters MS1 PCB revision 0520, the controller includes this LVDS chip on the board. It effectively extends the distance between the chained MS1 boards in extreme video installations to recommended 5 meters, which is more than doubled distance recommended for earlier board's revision. This allows for an additional design freedom and enables even bigger video installations than before.

5 BUILDING LED STRIPS DISPLAY USING PIXBLASTERS CONTROLLER

5.1 About LED Strips

The current Pixblasters MS1 Video LED Controller version supports both, 4-wire APA102-like and 3-wire WS2812B-like RGB LED interfaces. Supported 3-wire LED types include, but are not limited to: WS2811, WS2812, WS2812B, WS2815, SK6812, GS8208B and similar RGB LEDs. Supported 4-wire LED types include, but are not limited to: APA102, HD107S, SK9822 and similar RGB LEDs.

Pixblasters MS1 LED controller can be easily updated to support any other LED interface that may appear in the future. We will distribute new configuration files with the upgraded features sets and the Pixblasters users will update the controller through a simple procedure that does not require special tools or knowledge.

> ℹ️ The PAPP003 application note explains the Pixblasters FPGA LED controller update: https://pixblasters.com/wp-content/uploads/2021/09/papp003-Pixblasters-FPGA-Update-Guide_v1.0.pdf .

> ℹ️ The latest Pixblasters User's Manual and FPGA updates can be found at: https://pixblasters.com/deliverables/.

Just connect the Pixblasters controller to the PC via a serial USB cable, use on-board DIP switches to select the firmware flasher FPGA configuration and configure the controller through a simple user interface.

The WS2812B (**Figure 5.1**) is one of our preferred LED strip models because it is very cost effective, and in combination with the Pixblasters MS1 video LED controller, can work at full 60 frames per second (fps) video refresh rate. This is much faster than it is possible with the majority of competing controllers and enables smooth video reproduction regardless of the LED display's size.
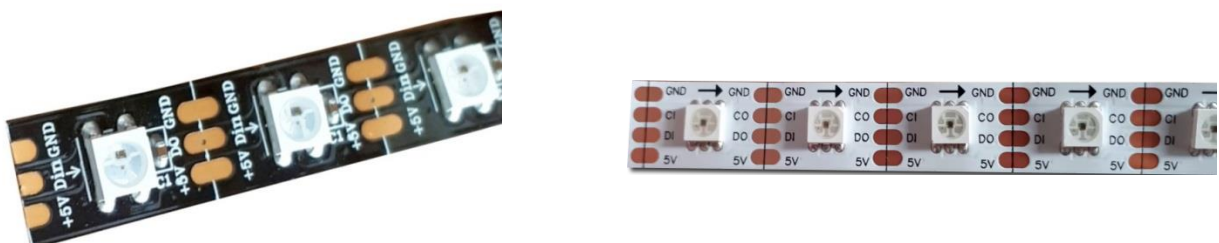
Figure 5.1 3-wire WS2812B LED on the Left, 4-wire APA102 LED on the Right Side

---

4-wire RGB LEDs (**Figure 5.1**) are often used in fully software-controlled LED installations because of the separated clock signal that can be controlled by the software. Of course, Pixblasters enable maximum speed with this sort of LEDs. An interesting feature of 4-wire LEDs is the brightness control.

The WS2812B RGB LED strips have the 3-pin interface: +5V power input, GND ground pin, and the DIN control data input.

The APA102 RGB LEDs have the 4-pin interface: +5V power input, GND ground pin, DIN control data input and CI clock input signal.

> 🚫 Note the small arrow sign on the LED strip. It shows the data flow direction and it **must be respected at all times**, or the LED strip will not light. The LED strip must be always connected to the Pixblasters video controller with the arrow pointing away from the board's digital output.

The **Figure 5.2** shows a simple LED video display formatting tool – the scissors. Use it to cut the video display line to the requested length.

> ℹ️ The WS2812B and APA102 LED strips can be cut between any two LEDs. The cutting positions for some other strips can be less flexible, i.e. some can be cut behind groups of three LEDs.

Figure 5.2 Define the LED Strip's Size and Display's Format

LED strips can be used without soldering. While being skilled with the soldering iron is definitely helpful, one can use the locking connectors, like the one shown on **Figure 5.3**, to connect the wire leads to the strip without soldering. The wire leads with the removed isolation simply insert into the Pixblasters controller's high-quality snap-action connectors. This way, the reliable contact between the strip and the controller can be established without soldering.
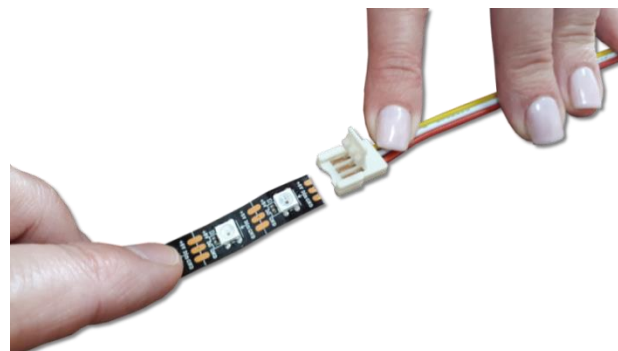
Figure 5.3 An Example Locking Connector for LED Strips

Besides the connector type shown on **Figure 5.3**, there are different types locking connectors that connect two pieces of the LED strip.

5.2 Connecting LED strips

The Pixblasters MS1 controller has 8 LED strip connectors CN7 – CN14. **Figure 5.4** shows LED strip connector layout used for 3-wire RGB LED strips. Strip 0–31 are control data outputs that connect to LED strips' DIN data inputs.

Pixblasters MS1 Board Detail   **LED Strip Connector Labeling**
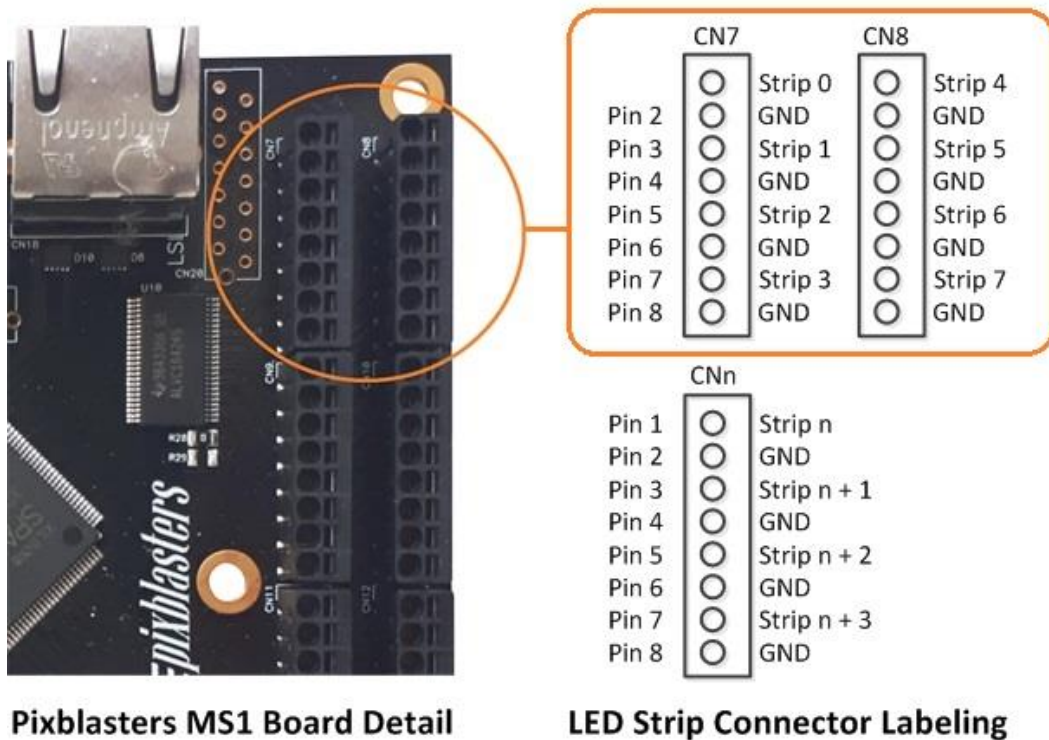
Figure 5.4 3-wire WS2812-like LED Strip Connector Layout

Each 3-wire LED strip (**Figure 5.4**) is connected to Pixblasters controller using one "Strip" and one "GND" pin. LED strip power supply is not connected to the Pixblasters controller directly. Separate Power and GND lines must be used for applying power to the LED strip (**Figure 8.1**).

The Pixblasters MS1 LED controller configured for 3-wire RGB LED strips control provides 32 digital control data outputs. Each data output is capable to drive 512 RGB LEDs.

The Pixblasters MS1 LED controller configured for 4-wire RGB LED strips control provides 16 digital control data outputs and 16 clock outputs. Each data output is capable to drive 1024 RGB LEDs.

Effectively, the maximum driving capacity of a single MS1 LED controller is equal for both LED types.

The next image shows the LED Strip Connector Layout used when the MS1 LED controller drives 4-wire RGB LED strips.

Pixblasters MS1 Board Detail   **4-wire LED Strip Connector Labeling**
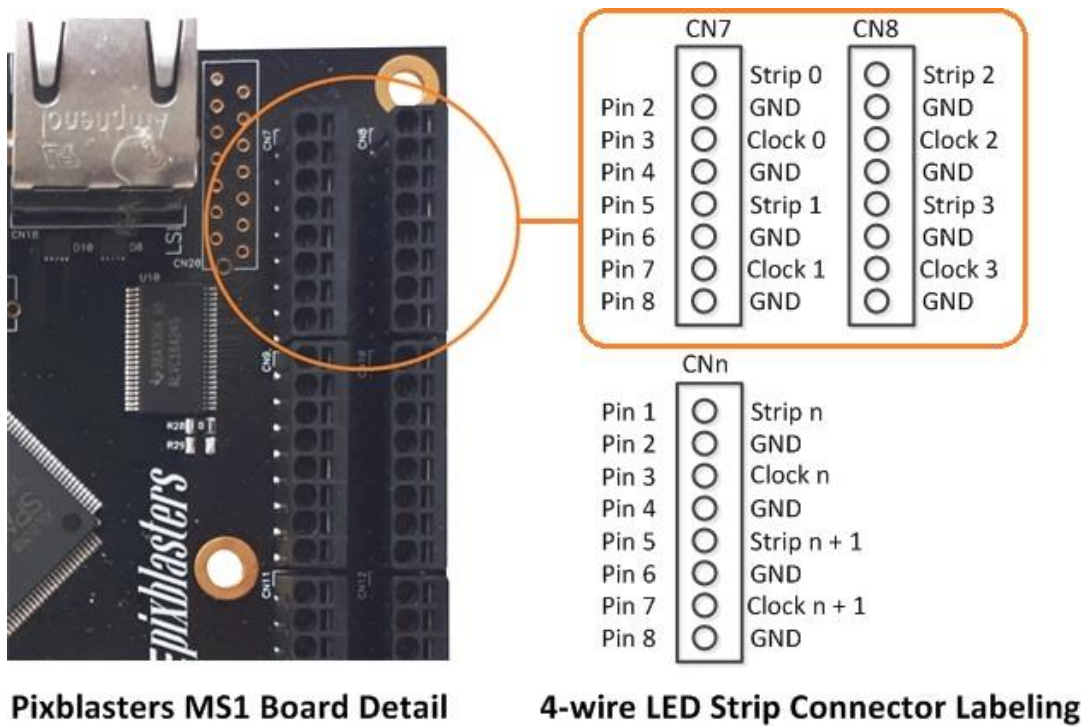
Figure 5.5 4-wire APA102-like LED Strip Connector Layout

Each RGB LED draws approximately 50 mA when it is set to full brightness white color and powered at V_{dd} = +5V, and this information helps the proper power supply selection. An example calculation for a 2,000 LEDs display:

Max. current: 2000 x 0,05A = 100A;

Max. DC power: $P = V_{dd} * I = 5V * 100A = 500W$

500W is valid for the worst case when all LEDs display white color. Since displays shows changing and colorful video, it enables use of power supplies of lower nominal power. This underpowered operational mode enables cost savings. Depending on the video content, the above 2,000 LEDs example display would comfortably work powered by 350W power supply.

Table 5.1: Pinout of LED Strip Connectors – 3-WIRE Configuration

Connector	Pin	Description	Pin	Description
CN7	1	Strip 0	2	GND
CN7	3	Strip 1	4	GND
CN7	5	Strip 2	6	GND
CN7	7	Strip 3	8	GND
CN8	1	Strip 4	2	GND
CN8	3	Strip 5	4	GND
CN8	5	Strip 6	6	GND
CN8	7	Strip 7	8	GND
CN9	1	Strip 8	2	GND
CN9	3	Strip 9	4	GND
CN9	5	Strip 10	6	GND
CN9	7	Strip 11	8	GND
CN10	1	Strip 12	2	GND
CN10	3	Strip 13	4	GND
CN10	5	Strip 14	6	GND
CN10	7	Strip 15	8	GND
CN11	1	Strip 16	2	GND
CN11	3	Strip 17	4	GND
CN11	5	Strip 18	6	GND
CN11	7	Strip 19	8	GND
CN12	1	Strip 20	2	GND
CN12	3	Strip 21	4	GND
CN12	5	Strip 22	6	GND
CN12	7	Strip 23	8	GND
CN13	1	Strip 24	2	GND
CN13	3	Strip 25	4	GND
CN13	5	Strip 26	6	GND
CN13	7	Strip 27	8	GND
CN14	1	Strip 28	2	GND
CN14	3	Strip 29	4	GND
CN14	5	Strip 30	6	GND
CN14	7	Strip 31	8	GND

Twisting wires that connect LED strips to the Pixblasters MS1 is highly recommended. It removes the electromagnetic interference between the wires, assures more reliability and flicker-free LED display operation. Connecting all GND pins between the Pixblasters LED controller and LED strips is also highly recommended.

Table 5.2: Pinout of LED Strip Connectors – 4-WIRE Configuration

Connector	Pin	Description	Pin	Description
CN7	1	Strip 0	2	GND
CN7	3	Clock 0	4	GND
CN7	5	Strip 1	6	GND
CN7	7	Clock 1	8	GND
CN8	1	Strip 2	2	GND
CN8	3	Clock 2	4	GND
CN8	5	Strip 3	6	GND
CN8	7	Clock 3	8	GND
CN9	1	Strip 4	2	GND
CN9	3	Clock 4	4	GND
CN9	5	Strip 5	6	GND
CN9	7	Clock 5	8	GND
CN10	1	Strip 6	2	GND
CN10	3	Clock 6	4	GND
CN10	5	Strip 7	6	GND
CN10	7	Clock 7	8	GND
CN11	1	Strip 8	2	GND
CN11	3	Clock 8	4	GND
CN11	5	Strip 9	6	GND
CN11	7	Clock 9	8	GND
CN12	1	Strip 10	2	GND
CN12	3	Clock 10	4	GND
CN12	5	Strip 11	6	GND
CN12	7	Clock 11	8	GND
CN13	1	Strip 12	2	GND
CN13	3	Clock 12	4	GND
CN13	5	Strip 13	6	GND
CN13	7	Clock 13	8	GND
CN14	1	Strip 14	2	GND
CN14	3	Clock 14	4	GND
CN14	5	Strip 15	6	GND
CN14	7	Clock 15	8	GND

5.3 Building LED Display by LED Strips

LED strips can be arranged in a LED display's matrix driven by the Pixblasters controller in two ways. The **Figure 5.6** shows the 3-wire RGB LED strips arrangement without strip segmentation. Depending on the used LED type, Pixblasters MS2 controller outputs can drive different number of pixels. In a case of 3-wire LEDs, each of 32 Pixblasters MS1 controller outputs drives a display line of maximally 512 LED pixels. In a case of 4-wire LEDs, each of 16 Pixblasters MS1 controller data outputs (other 16 outputs are for clock signals) drives a display line of maximally 1024 LED pixels.
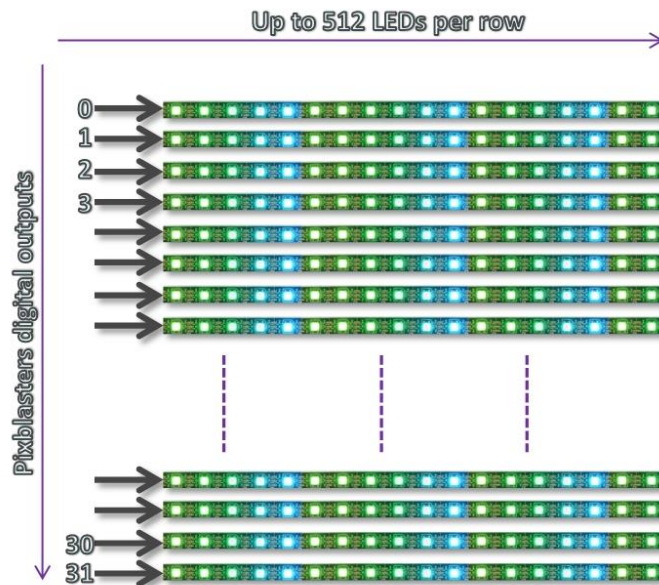

Figure 5.6 LED Strips Matrix with no Segmentation (3-wire LEDs)

Figure 5.7 shows the LED strips arrangement with the line segmentation, which enables more flexibility in display's resolution setup. Each Pixblasters MS1 controller's digital output can drive max. 512 pixels in 3-wire and 1024 pixels in 4-wire LED configuration, but those pixels can be arranged in multiple display lines. The **Figure 5.7** example shows 4 lines driven by a single Pixblasters digital output and this arrangement enables the maximum resolution 128 x 128 (Horizontal x Vertical) pixels.

The Pixblasters controller needs to be configured, through the configuration menus, for such LED matrix arrangement and the further display control is fully automatic. There is no need for any input image manipulation.

The LED matrix with the embedded segmentation requires a bit different LED strips wiring and it is absolutely necessary to take care on the data flow direction. Take for example the four horizontal lines controlled by the digital output 0 from the **Figure 5.7** example. The first LED strip segment must have the data arrow (**Figure 5.1**) pointing from left to right, the second LED segment must have the data arrow pointing from right to left, the third one from left to right, and the fourth from right to left. Such LED arrangements assures an uninterrupted serial data flow.

> The Pixblasters application note PAPP004 fully explains non-segmented and segmented (zig-zag) strips arrangements: http://pixblasters.com/wp-content/uploads/2021/03/papp004-Pixblasters-LED-Strips-Matrix-Explained_v1.0.pdf.
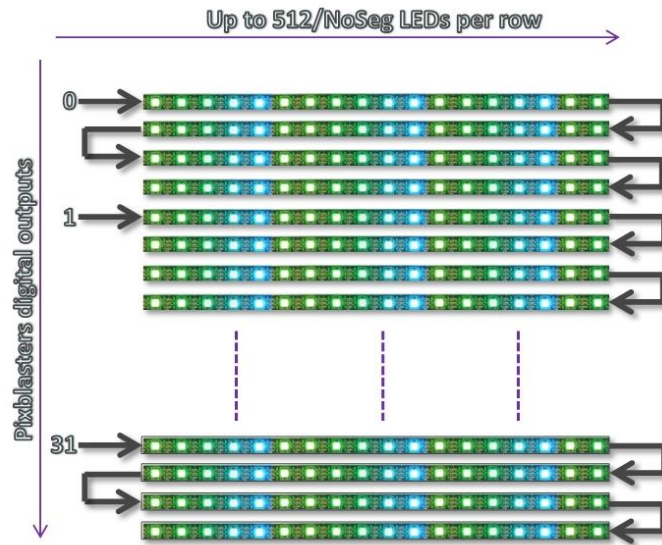
Figure 5.7 LED Strips Matrix with Segmentation (3-wire LEDs)

> ℹ️ Pixblasters MS1 LED controller drives up to 1024 pixels per each of 16 digital outputs in 4-wire APA102-like configurations. **Figure 5.7** explains the 3-wire LED matrix segmentation.

The PAPP004 application note explains details related to the display matrix description. Real demo examples are demonstrated in this short video clip: https://youtu.be/o_5djzdhzHY.
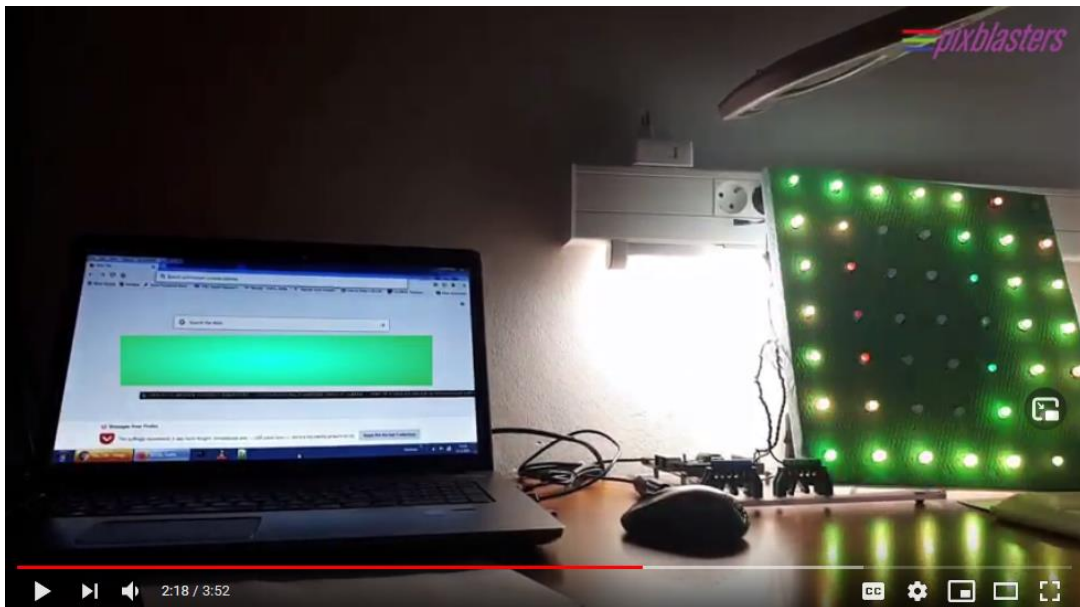
Figure 5.8 The Fastest Possible WS2811 Video LED Display

5.4 Wiring Pixblasters LED Outputs

This video clip explains how to use Pixblasters MS1 LED strip connectors with the snapping function:
https://www.youtube.com/watch?v=vgMeKpVCEYw
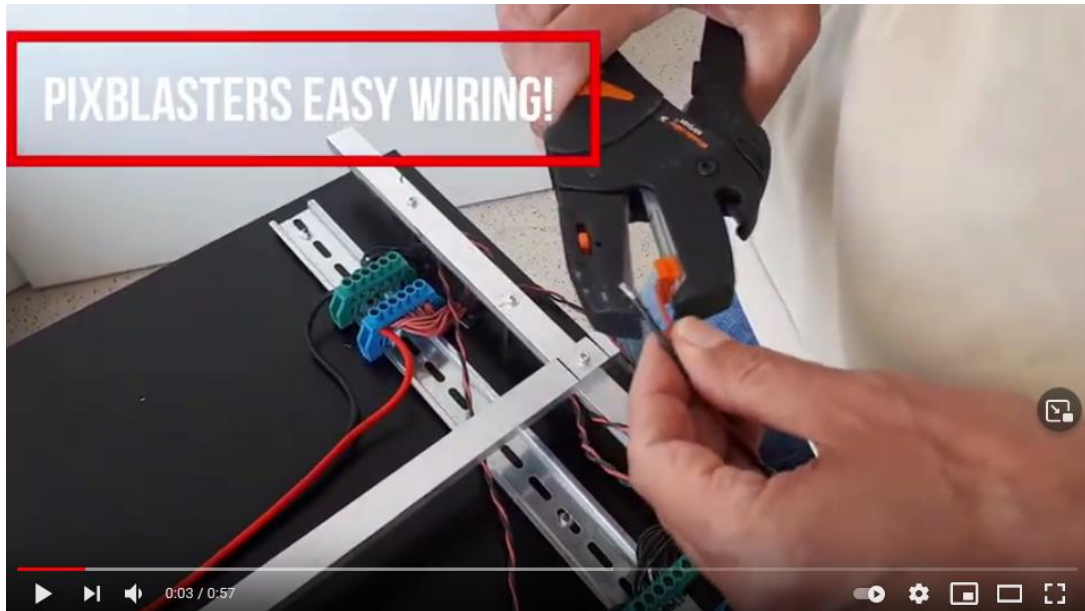
Figure 5.9 Easy Wiring of LED Outputs

5.5 Connecting Video Source to the Pixblasters MS1 Controller
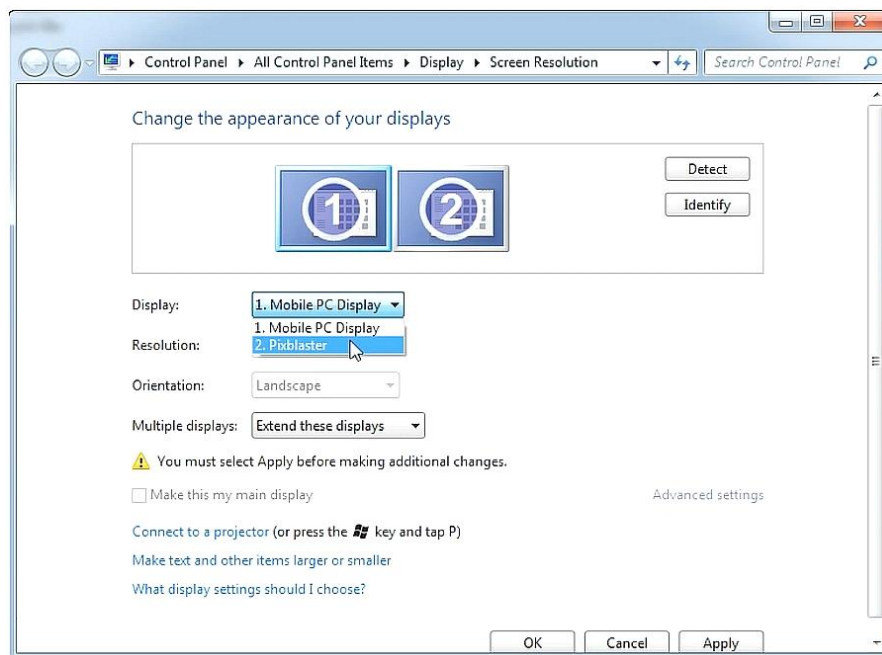
Figure 5.10 The Pixblasters Controller Recognized as a Monitor by Microsoft Windows OS

Consumer monitors have an EEPROM chip called the EDID. It contains monitor's name and technical info about its capabilities. This information enables any standard video source, i.e. the PC computer, to properly detect the monitor and automatically format the video image suitable for the specific monitor.

The Pixblasters MS1 Video LED Controller also includes Extended Display Identification Data (EDID) chip to smoothly connect to any operating system as a standard monitor.

> *i* The recommended length of the input video cable is up to 1 meter. In case of video sources with weaker driving capabilities and longer cables between the video source and the Pixblasters MS1 LED controller, a noticeable flickering of the LED display is possible.
>
> In such situations we recommend use of active signal conditioners, such as the active HDMI video cable, HDMI splitters or HDMI signal repeaters. Please note that the Pixblasters MS1 is DVI video compatible and the listed HDMI equipment should smoothly pair with it.

Figure 5.11 Active HDMI Cable

Figure 5.12 HDMI Repeater (Cable Extender)

> 🚫 Please respect the required signal direction as labeled on video equipment. For example, the Source side of the active video cable should be connected to the video source, while the TV side of the cable connects to the Pixblasters MS1 video input.

Figure 5.13 Example HDMI Splitter – Input

Figure 5.14 Example HDMI Splitter – Outputs

6 BUILDING LED PANELS DISPLAY USING PIXBLASTERS CONTROLLER

6.1 About WS2812B LED Panels

There are various types of WS2812B LED panels available on the market. They can be easily distinguished based on their horizontal and vertical resolution. Most panels have three directly soldered cables: the data input (DIN) cable, the data output (DOUT) cable, and an optional power cable.

The DIN and DOUT cables come with standard LED connectors, facilitating straightforward connectivity—DIN fits only into DOUT, preventing incorrect connections. Both DIN and DOUT cables also distribute +5 VDC power. However, due to the cables' limited thickness, power distribution can become insufficient when connecting a larger number of LED panels. This condition can be identified by the "LED browning effect," where LEDs emit yellow-brownish tones instead of a pure white color. If this occurs, the optional power cable must be used to evenly distribute power to all LEDs. Before assembling larger LED video installations, Pixblasters recommends users unequivocally determine the LED input and fully understand the panel wiring

> ℹ️ Supported LED panels are built by WS2812B compatible LEDs, and aside to two new registers and some minor changes described in the **10 internal Registers**, the controller should be initialized in a very similar way. The LED outputs must be used as explained in the **Figure 5.4 3-wire WS2812-like LED Strip Connector Layout**.

6.1.1 8 x 8 LED Panel
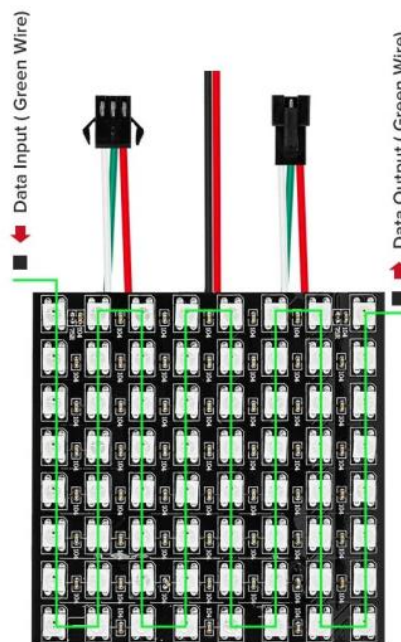
Figure 6.1 8x8 LED Panel

6.1.2 16 x 16 LED Panel
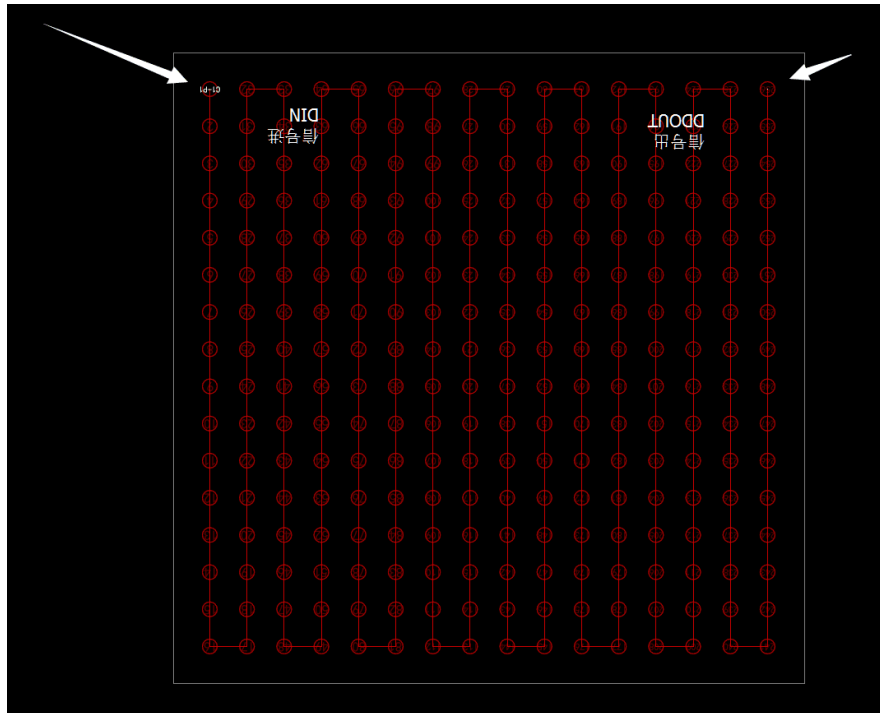
Figure 6.2 16x16 LED Panel

6.1.3 32 x 8 LED Panel
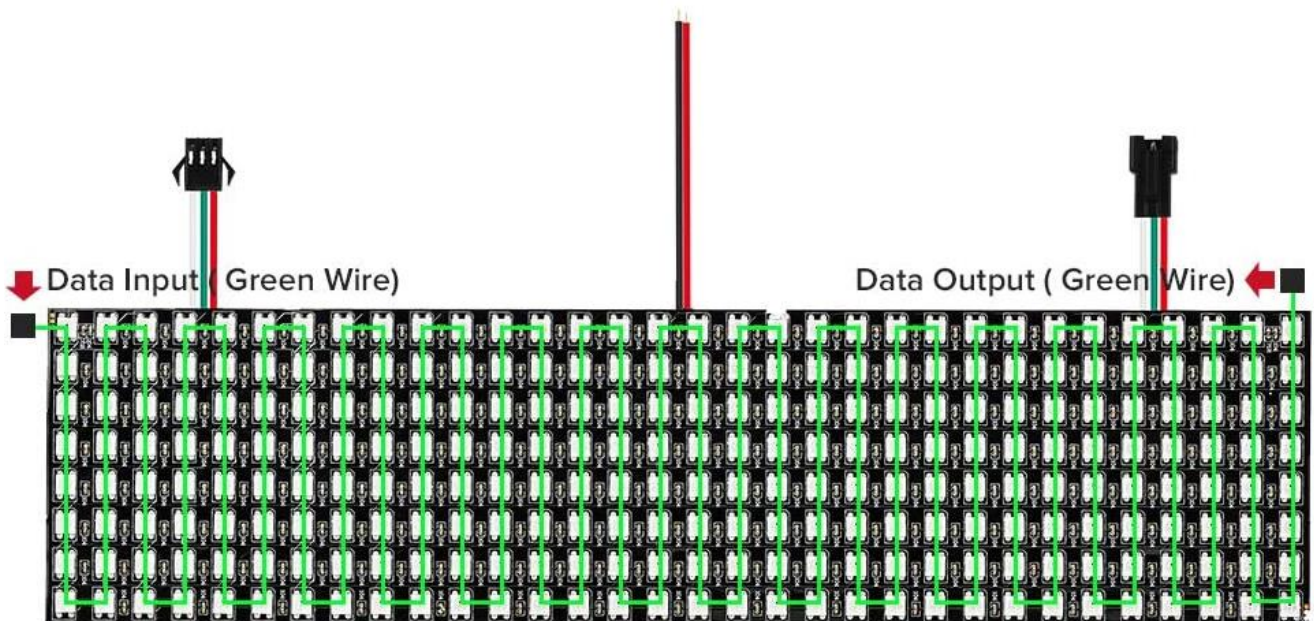
Figure 6.3 32x8 LED Panel

6.1.4 22 x 22 LED Panel
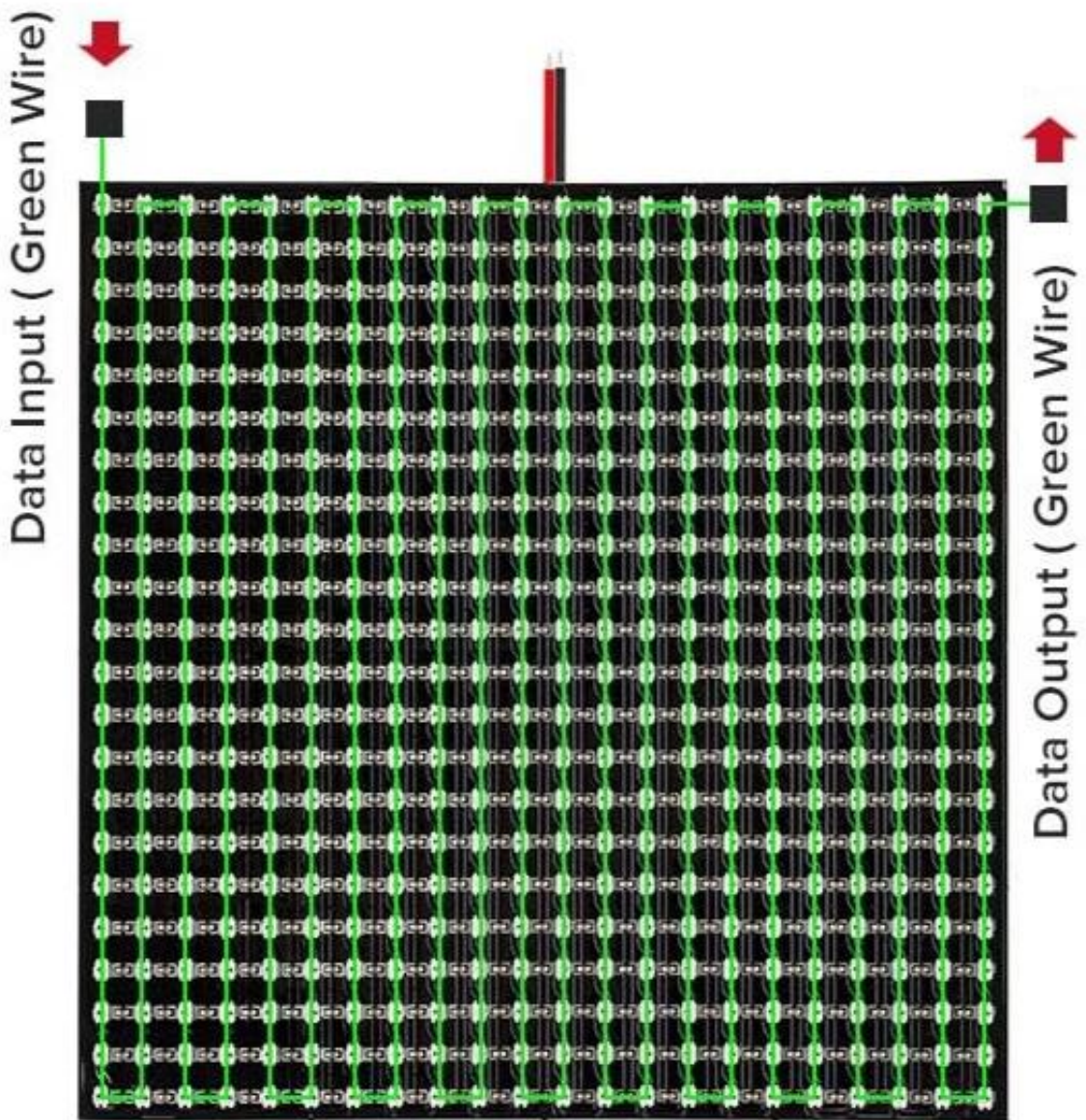
Figure 6.4 22x22 LED Panel

6.1.5 44 x 11 LED Panel
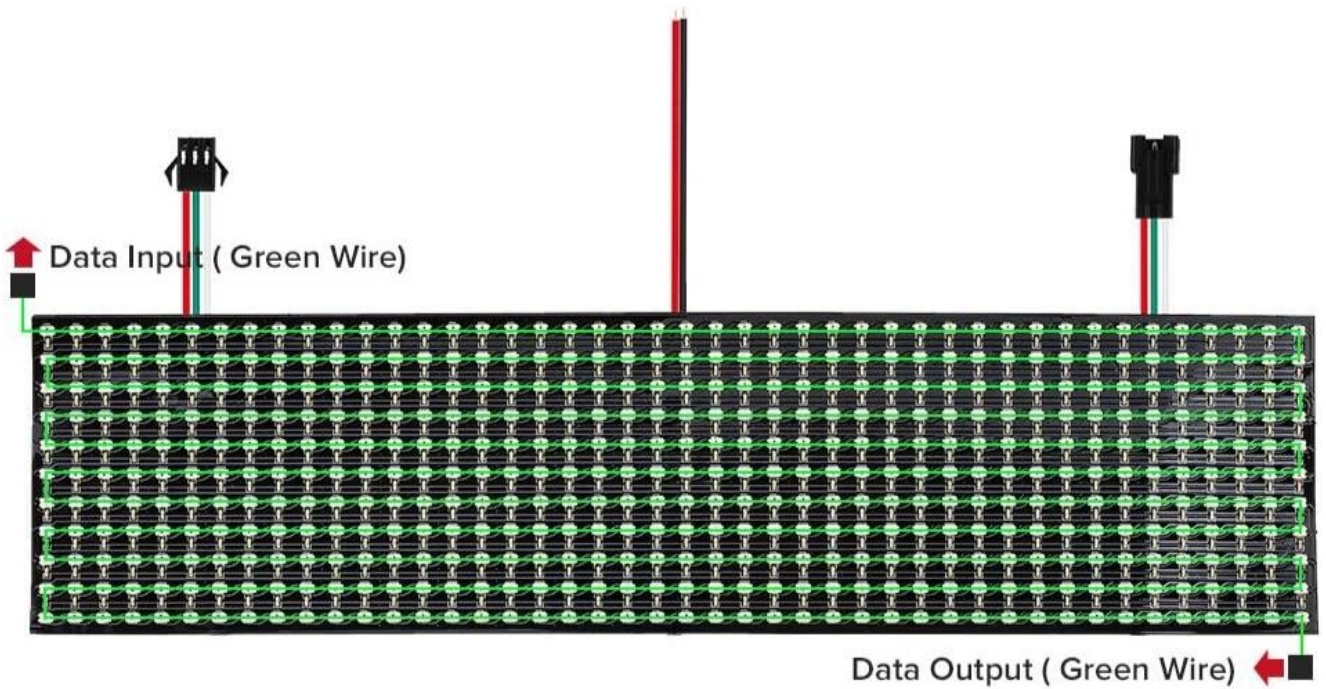
Figure 6.5 44x11 LED Panel

6.1.6 Other LED Panel Types

The Pixblasters MS1 can control different LED panels with standard and customized resolutions and wiring, including those not listed in this document. For more information and support on driving other LED panel types, please contact us at info@pixblasters.com.

6.2 LED Panels Driving Capacity

LED Panel Resolution (H x V)	Supported LED panels per a single LED output	Supported LED panels per a single MS1 LED controller
8 x 8	8	256
16 x 16	2	64
32 x 8	2	64
22 x 22	1	32
44 x 11	1	32
7x7	8	256

Table 6.1: Number of Supported LED Panels per a Single LED Output

The Pixblasters MS1 controller can drive up to 512 LEDs per output at the maximum refresh rate. It determines the maximum number of certain type LED panels that can be connected to a single output.

1. **8x8 LED Panels**:
    o Each panel contains 64 LEDs.
    o Up to 8 panels can be connected serially.
    o Total LEDs for 8 panels: 8 x 64 = 512 LEDs.
2. **16x16 and 32x8 LED Panels**:
    o Each panel contains 256 LEDs.
    o Up to 2 panels can be connected.
    o Total LEDs for 2 panels: 2 x 256 = 512 LEDs.
3. **22x22 LED Panels**:
    o Each panel contains 484 LEDs.
    o Only 1 panel can be connected.
    o Total LEDs for 1 panel: 1 x 484 = 484 LEDs.

Important Considerations:

- The total number of LEDs (512) must be divisible by the number of LEDs in each panel type without a remainder for optimal configuration.

- When the number of LEDs in a panel type does not evenly divide into 512, the number of supported panels will vary accordingly. For example, since 22x22 panels have 484 LEDs, only one such panel can be connected to a single LED output.

> By following these guidelines, you can ensure that LED panels are correctly connected to the Pixblasters MS1 controller for optimal performance. Please note that the Pixblasters MS1 controller's maximum driving capacity is not fully utilized when fewer than 512 LEDs are connected per output.

6.3 Connecting LED Panels
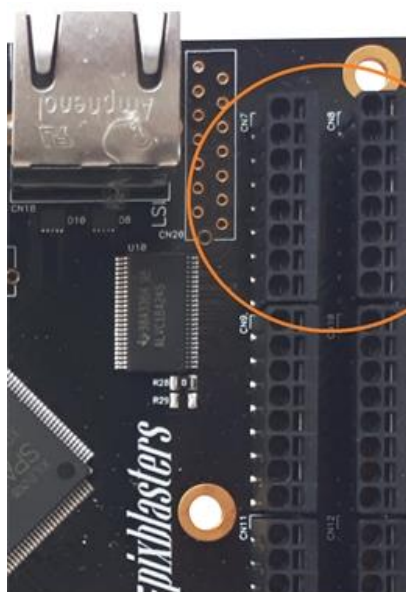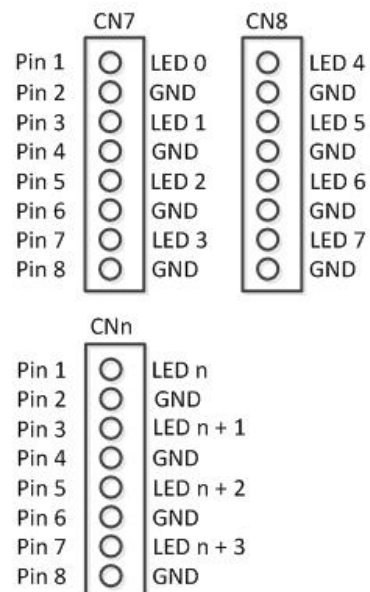
Pixblasters MS1 Board Detail   **3-wire LED Strip Connector Labeling**

Figure 6.6 3-wire WS2812-like LED Panels Connector Layout

The Pixblasters MS1 controller has 8 LED strip connectors CN7 – CN14. **Figure 6.5** shows LED connectors layout used for 3-wire RGB LED strips and LED panels. LED 0–31 are control data outputs that connect to LED panels' DIN data inputs.

Each "LED" and "GND" wire pair can control as many serially connected LED panels as it is listed in the **Table 6.2**. For, example it is possible to connect eight (8) 8x8 LED panels to LED 0 and GND control outputs.

Table 6.2: Pinout of LED Connectors – 3-WIRE Configuration

Connector	Pin	Description	Pin	Description
CN7	1	LED 0	2	GND
CN7	3	LED 1	4	GND
CN7	5	LED 2	6	GND
CN7	7	LED 3	8	GND
CN8	1	LED 4	2	GND
CN8	3	LED 5	4	GND
CN8	5	LED 6	6	GND
CN8	7	LED 7	8	GND
CN9	1	LED 8	2	GND
CN9	3	LED 9	4	GND
CN9	5	LED 10	6	GND
CN9	7	LED 11	8	GND
CN10	1	LED 12	2	GND
CN10	3	LED 13	4	GND
CN10	5	LED 14	6	GND
CN10	7	LED 15	8	GND
CN11	1	LED 16	2	GND
CN11	3	LED 17	4	GND
CN11	5	LED 18	6	GND
CN11	7	LED 19	8	GND
CN12	1	LED 20	2	GND
CN12	3	LED 21	4	GND
CN12	5	LED 22	6	GND
CN12	7	LED 23	8	GND
CN13	1	LED 24	2	GND
CN13	3	LED 25	4	GND
CN13	5	LED 26	6	GND
CN13	7	LED 27	8	GND
CN14	1	LED 28	2	GND
CN14	3	LED 29	4	GND
CN14	5	LED 30	6	GND
CN14	7	LED 31	8	GND

Power supply for LED panels is not routed through the Pixblasters controller board.

The Pixblasters MS1 LED controller configured for 3-wire RGB LED strips and LED panels control provides 32 digital control data outputs. Each data output is capable to drive 512 RGB LEDs.

Twisting wires that connect LED panels to the Pixblasters MS1 is highly recommended. It removes the electromagnetic interference between the wires, assures more reliability and flicker-free LED display operation. Connecting all GND pins between the Pixblasters LED controller and LED strips is also highly recommended.

6.4 Assembling Video Display by LED Panels

Each LED output can control multiple LED panels connected in series. Consequently, a single LED output controls an LED band with the resolution (H x V) equal to:

(Number of panels per output * LED Panel Horizontal) * LED Panel Vertical

For example, when using an 8x8 LED panel, one output (controller 512px/output) controls an LED band resolution of:

(8 panels * 8 pixels Horizontal) * 8 pixels Vertical = 64 x 8 (H x V)

The horizontal resolution of the display can be expanded by using multiple adjacent LED outputs to drive the same LED band. For instance, in a video display made of 8x8 LED panels, to achieve a horizontal resolution of 256 pixels, 4 LED outputs (controller 512px/output) are needed to drive the complete 256x8 LED band.

(4 outputs x (8 panels * 8 pixels Horizontal)) * 8 pixels Vertical = 256 x 8 (H x V)

In the same example, if you use 4 outputs for the horizontal LED band, you can have 8 such LED bands vertically because of the maximum number of LED outputs -> 32 / 4 = 8. The resolution of such a display would be:

8 * LED Band = 8 * (256 x 8) = 256 x 64 (H x V)

> ℹ️ For more details on how to build video displays with RGB LED panels, please read the application note PAPP006: https://pixblasters.com/wp-content/uploads/2024/07/papp006-Pixblasters-Build-VideoDisplay-with-LED-Panels_v1.0.pdf
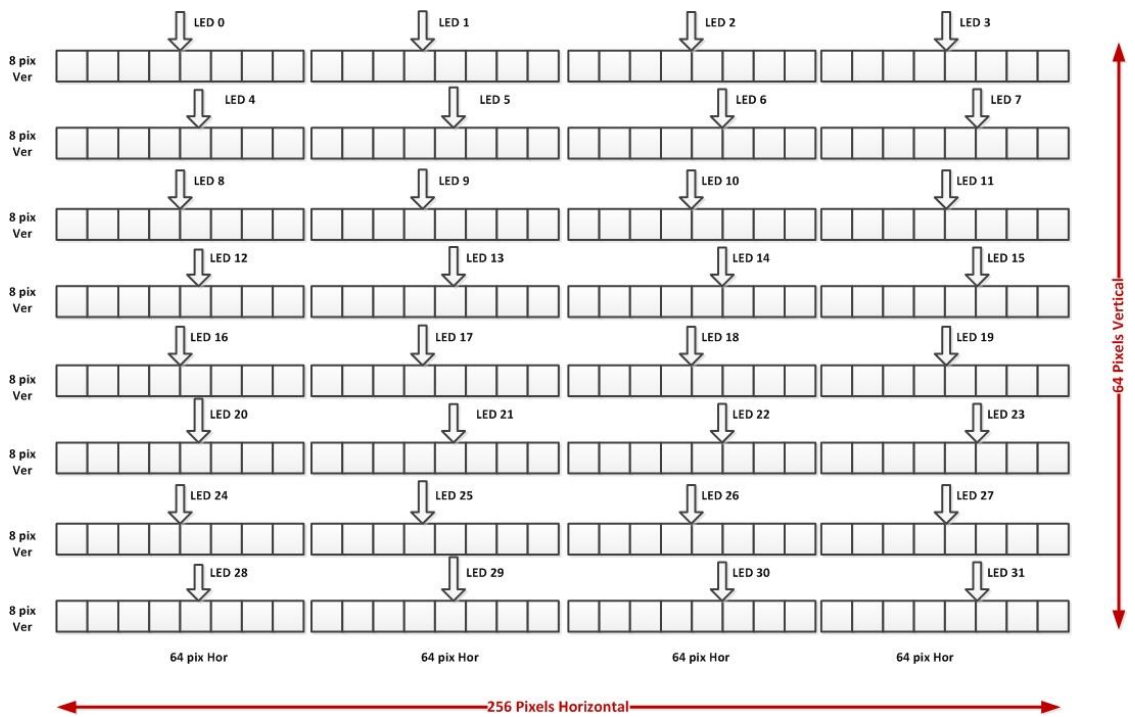

Figure 6.7 256x64 LED Display Built by 8x8 LED Panels

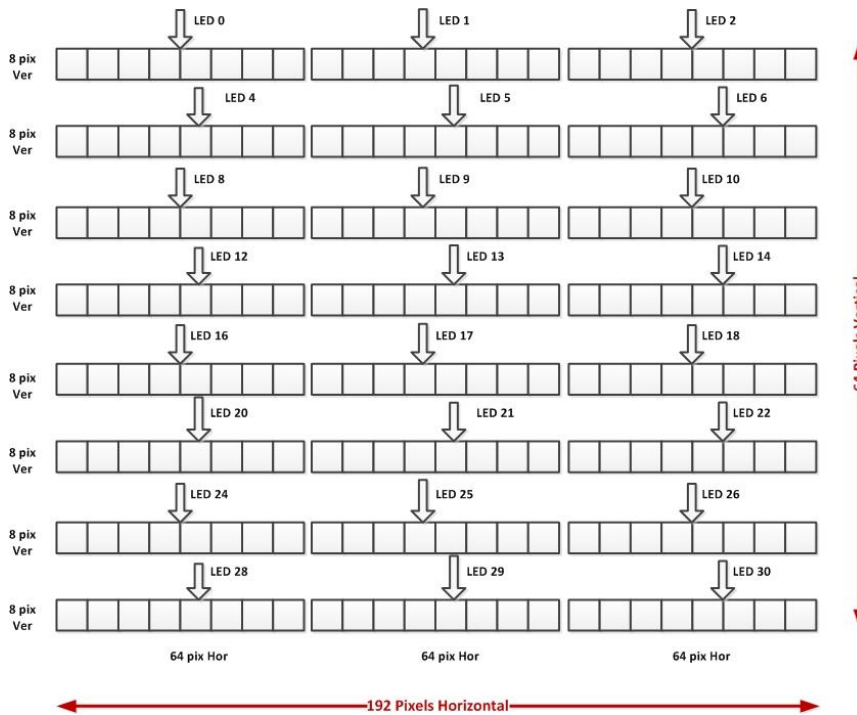## 6.5 Key Assembly Constraints for LED Panel Video Displays

Figure 6.8 192x64 LED Display Built by 8x8 LED Panels

The number of LED outputs combined to drive the horizontal LED band <u>must be divisible by two (2)</u>. The **Figure 6.8 192x64 LED Display Built by 8x8 LED Panels** explains the use of LED outputs

when the horizontal resolution is lower than the maximum provided by the number of the LED outputs dedicated per LED band.

Horizontal resolution of 192 LEDs cannot be achieved by a single LED output driving serially connected 8x8 LED panels. Whenever is necessary to use more than a single LED output, the allowed number of combined outputs is 2, 4, 6…32.

The **Figure 6.8** shows that LED outputs number 3, 7, 11… cannot be used. It effectively decreases the LED driving capacity from the max. 16,384 LEDs to 12,288 LEDs. It represents the 25% decrease of the driving capacity.
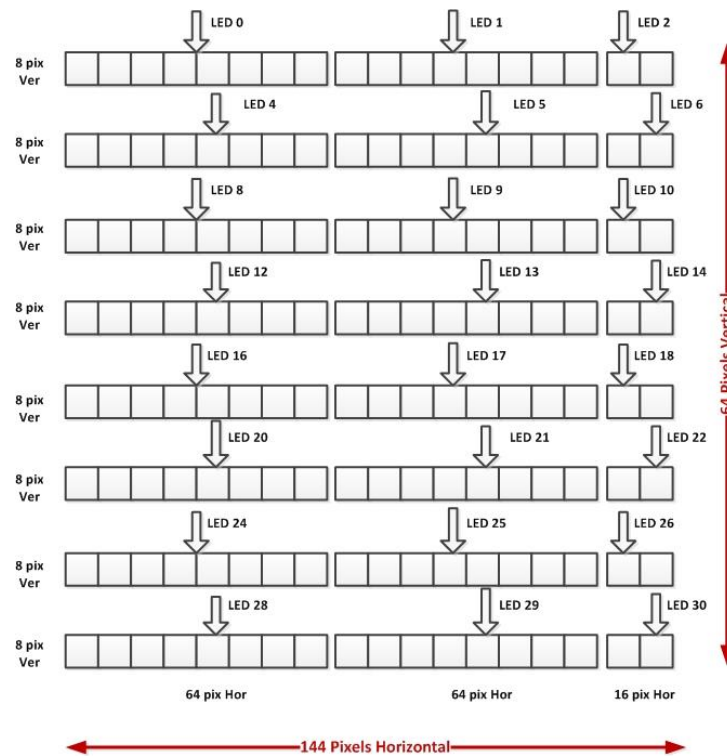
Figure 6.9 144x64 LED Display Built by 8x8 LED Panels

The **Figure 6.9** display displays a very ineffective way of display assembly with 8x8 LED panels. Because of additional 2 LED panels at the end of the LED band controlled by a single LED output, the overall LED driving capacity would be decreased very much. Groups of 4 LED outputs would drive only 14 8x8 LED panels and the driving capacity would be decreased to 7,168 LEDs.

6.6 Wiring Pixblasters LED Outputs

The process is explained in the paragraph **5.4 Wiring Pixblasters LED Outputs**.

6.7 Connecting Video Source to the Pixblasters MS1 Controller

The process is explained in the paragraph **5.5 Connecting Video Source to the Pixblasters MS1 Controller**.

7 SYSTEM TOPOLOGIES

7.1 Single Controller Topology

Pixblasters controllers may be arranged in the Single controller topology and the Multi-Controller topology. Each Pixblasters MS1 controller, depending on the loaded FPGA configuration, supports either up to 32 outputs/512 pixels for 3-wire LED strip and LED panels, or 16 outputs/1024 pixels for 4-wire LED strip combinations.

As its name suggests, the single controller topology uses only one Pixblasters MS1 controller to drive the LED video display. When it is driving LED strips, the Single controller topology is further subdivided into "Non-segmented" and "Segmented" topologies.

7.1.1 LED Strips Non-Segmented Single Controller Topology

Figure 7.1 shows the Pixblasters MS1 controller in "Non-segmented single controller" topology. "Non-segmented" means that each LED strip shows a single video line. When "Non-segmented" mode is used, vertical resolution of a single MS1 controller is limited to the number of strips connected to the Pixblasters controller.
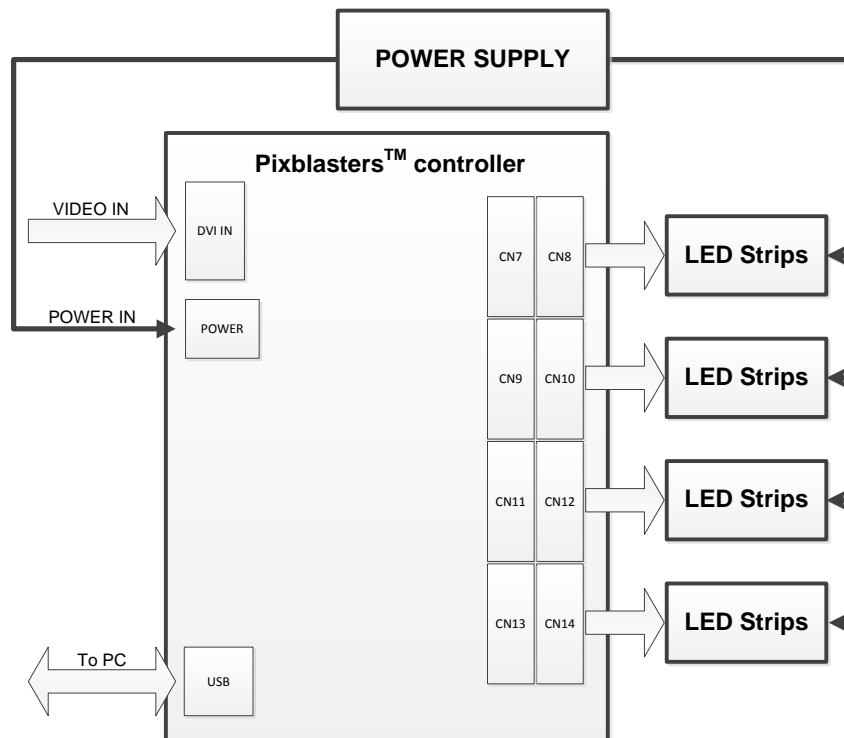

Figure 7.1 Single Controller Topology

7.1.2 LED Strips Segmented Single Controller Topology

In terms of how LED strips are connected to the Pixblasters MS1 controller, "Segmented" and "Non-segmented" topologies are identical. Therefore, block schematic shown in the **Figure 7.1** still applies. What is different is the way how LED strips are used and configured. In "Segmented" topology each LED strip may contain several video lines (see **Figure 5.7**). A number of lines depend on the value stored in the **SegmentsNO** register. This way, vertical resolution of a single MS1 controller is no longer limited to the number of connected LED strips. Since LED strip is treated by Pixblasters controller as segmented, it has to be physically configured (broken into segments) as shown in **Figure 5.7** and **Figure 7.2**.
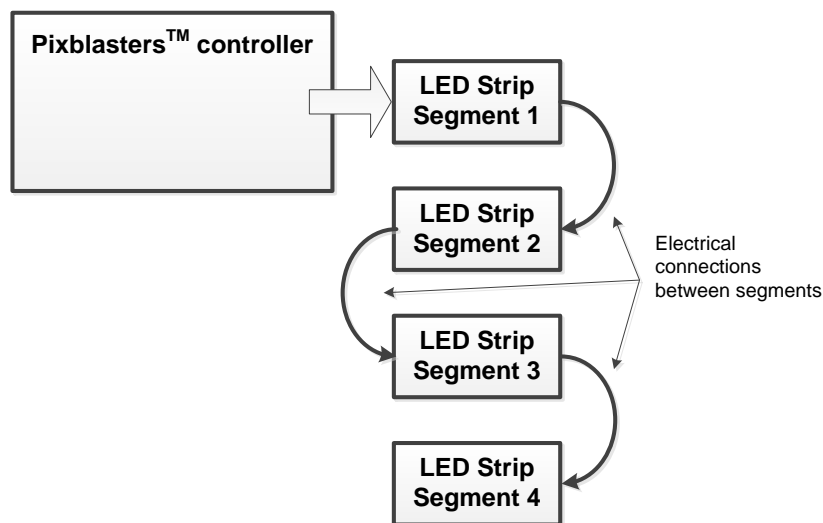
Figure 7.2 LED Strip Configuration for "Segmented" Topology

> Please find detailed explanations and controller configuration examples in the PAPP004 application note - LED Strips Matrix - Non-Segmented and Segmented (Zig-Zag):
> http://pixblasters.com/wp-content/uploads/2021/03/papp004-Pixblasters-LED-Strips-Matrix-Explained_v1.0.pdf

7.2 Multi-Controller Topology

Multi-controller topologies use more than one Pixblasters MS1 controller to drive LED displays build be larger number of LED strips and LED panels. In that case, system has one "Master" controller to which the video source is connected, and up to 48 downstream "Slave" controllers chained together using standard Ethernet cable. Please, see the chapter **Powering the Pixblasters controller based LED display** for more information.

7.2.1 Non-Mirrored Multi-Controller Topology

Figure 7.3 and **Figure 7.4** show how Pixblasters LED controllers may be connected to build a "Non-mirrored" multi controller display. The only difference between these two examples is related to a power supply configuration. More details about the power supply connections are given in the chapter **Powering the Pixblasters controller based LED display**.

> ℹ️ This topology can be used with both, LED strips and LED panels.
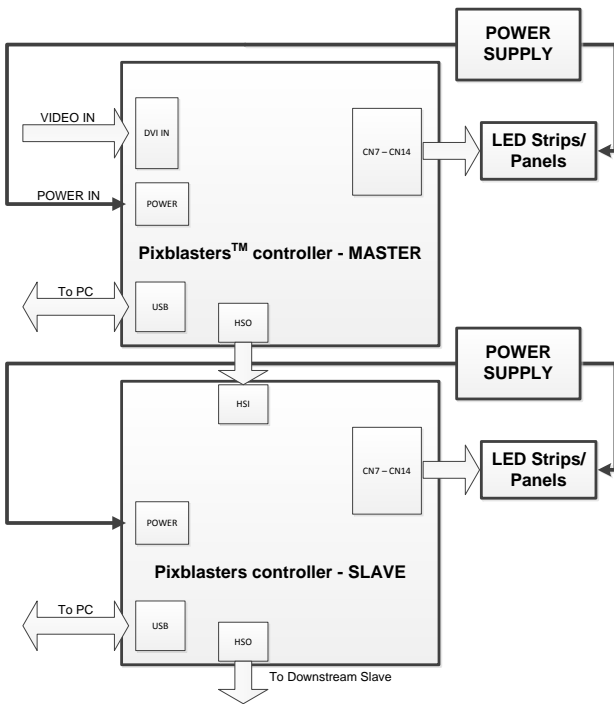
Figure 7.3 Multi-Controller Topology – Separate Power Supplies
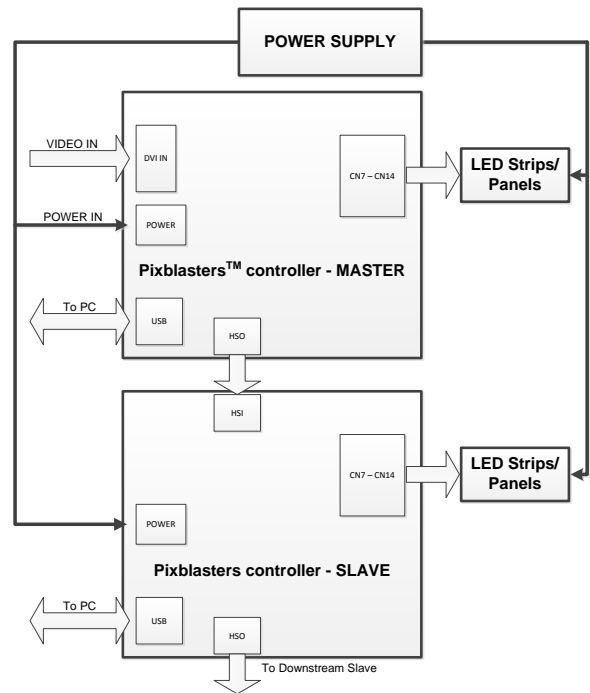
Figure 7.4 Multi-Controller Topology – Common Power Supply

7.2.2 LED Strips Mirrored Multi-Controller Topology

When used in the "Non-mirrored topology", depending on the loaded FPGA configuration, display's horizontal resolution is limited to either 512 pixels with 3-wire strips or 1024 pixels with 4-wire strips. The "Mirrored" mode is used to overcome that limit and effectively double the horizontal resolution. **Figure 7.5** shows how Pixblasters controllers should be connected when "Mirrored" mode is used. It can be seen that controllers are arranged in two adjacent columns – **Figure 7.5**.

Pixblasters controllers in the right column are "non-mirrored" and should be configured to handle right portion of the video lines. Pixblasters controllers in the left column should be "mirrored" and configured to handle the left portion of the video lines. **Figure 7.6** shows display structure when "Mirrored" multi-controller topology is used.
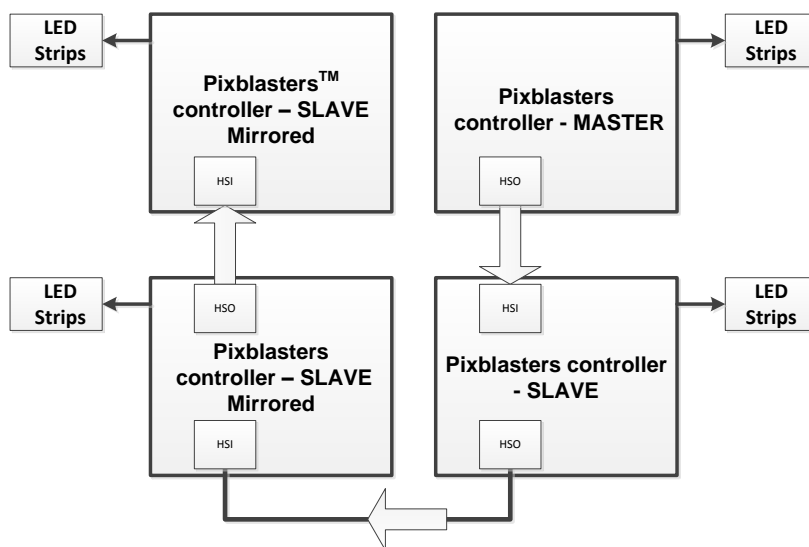
Figure 7.5 Mirrored Multi-Controller Topology

LED Strips L1	LED Strips R1
LED Strips L2	LED Strips R2

Figure 7.6 Display Structure in the Mirrored Multi-Controller Topology

7.3 Multi-controller Demo Display Architecture

Figure 7.7 shows Pixblasters demo display developed for designing and testing purposes. This small demo contains all parts necessary for building big and segmented LED displays.

There is a single master (M) Pixblasters MS1 controller that receives, through the monitor cable (1), video from the Raspberry Pi video source (VS). Slaves (S1) and (S2) MS1 LED controllers are interconnected by UTP Ethernet (2) cables that propagate the input video image. LED outputs (3) are connected to LED strips, which are placed on the opposite side of a mechanical structure and invisible on the photo.

The complete system is powered by (4) power supplies.
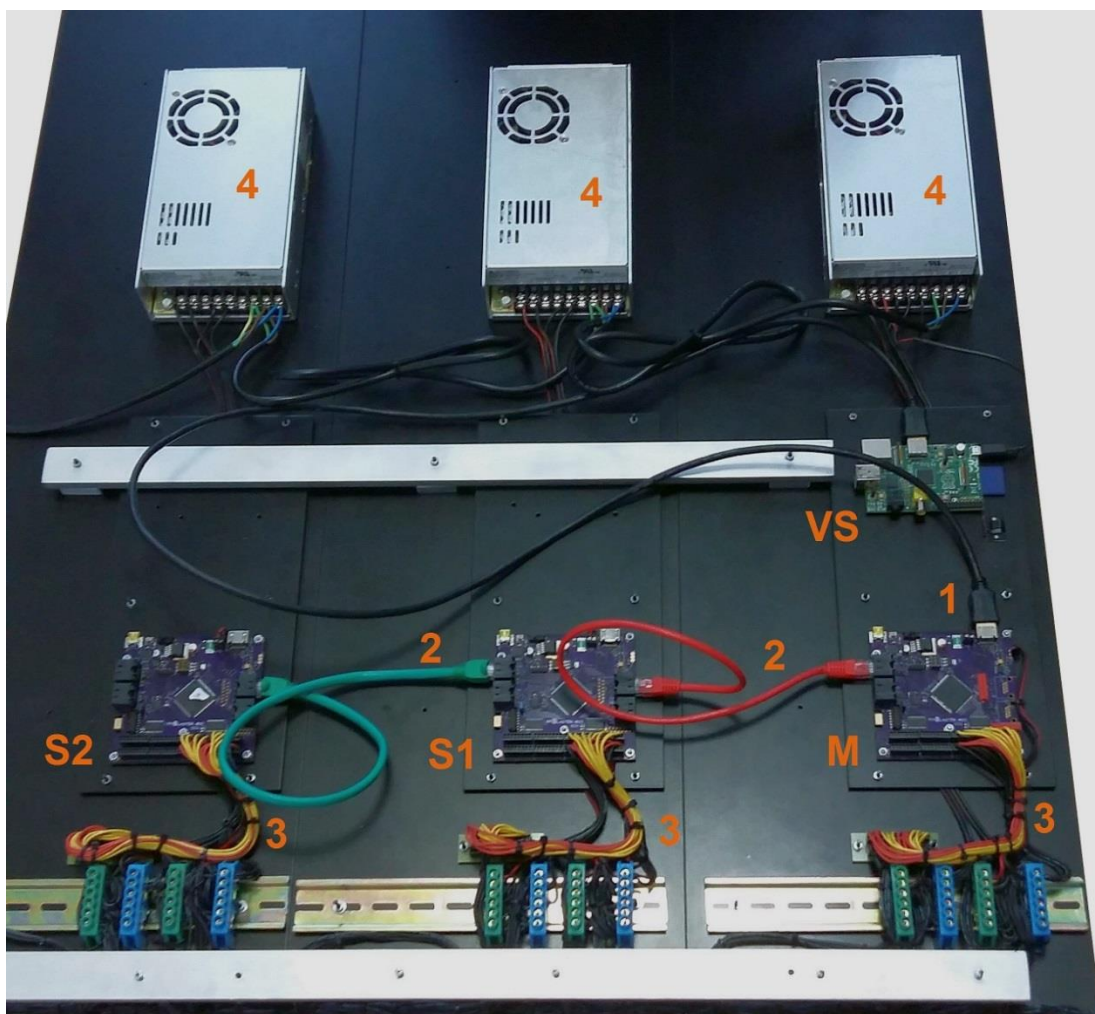
Figure 7.7 Demo Display Structure in the Multi-Controller Topology

> Check out this video clip to see 2 x 2 meters (120 x 120 pixels) Pixblasters demo/test display in action: https://www.youtube.com/watch?v=Yd34FQWuG3s

8 POWERING THE PIXBLASTERS CONTROLLER BASED LED DISPLAY
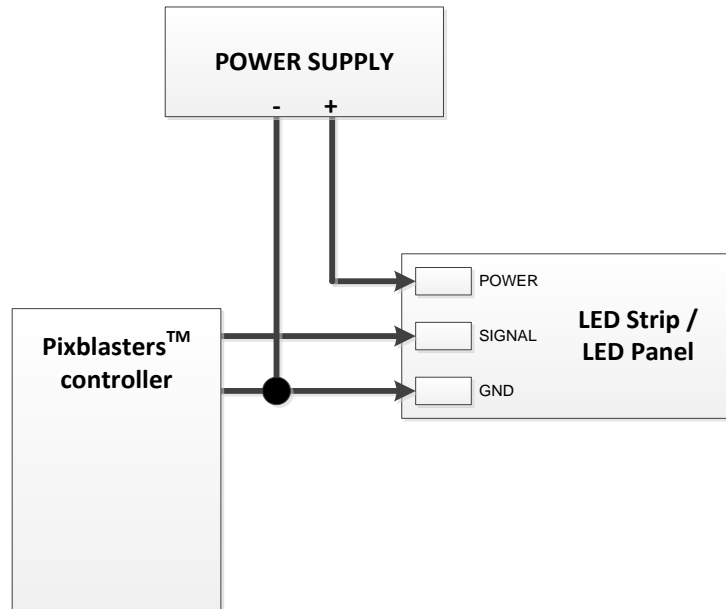
Figure 8.1 Connecting LED Strip to Power Supply and Controller

Power supply topology will mainly depend on the current consumption of the connected LEDs and on the power supply's current rating. In a single controller topology, one power supply may be enough to provide power for the whole system. If, however, current consumption of the connected LEDs requires multiple power supplies, than the scheme shown in **Figure 8.2** should be used.
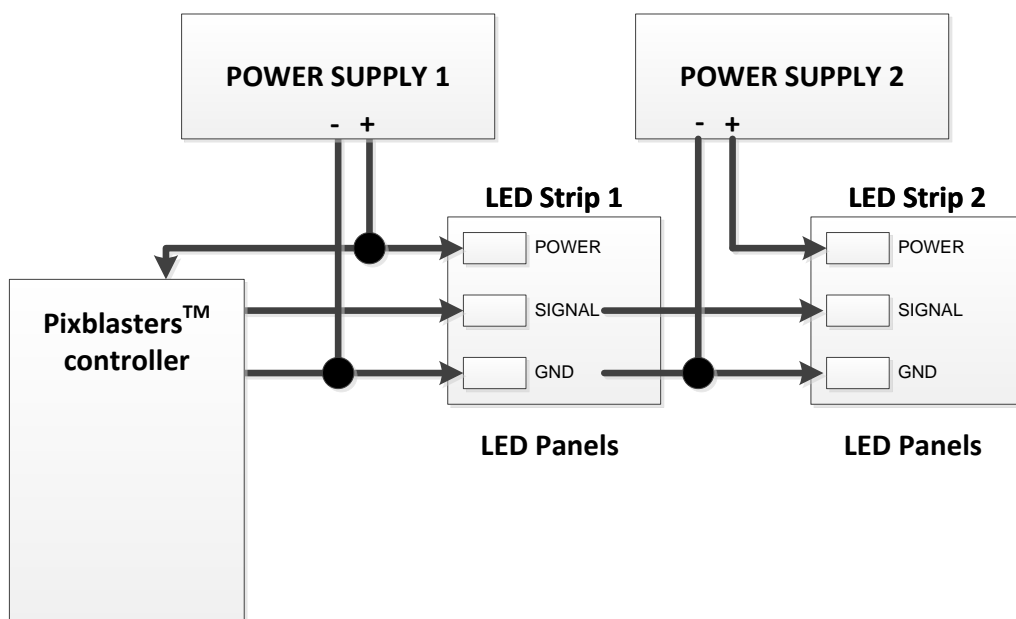
Figure 8.2 Multiple Power Supplies in a Single Controller Topology

Figure 8.2 assumes that LED strips/panels and Pixblasters controller use the same power supply voltage (5V). In that case, the Pixblasters controller should be connected to the power supply which powers the strip directly connected to the Pixblasters controller.

In the multi-controller topology, power supply connecting scheme will again mainly depend on current consumption of the connected LEDs and on the power supply's current rating. If power supply can provide enough current for all display elements (LEDs and controllers), than scheme shown in **Figure 8.3** should be used. Again, this scheme assumes that LEDs and Pixblasters controller use the same power supply voltage (5V).

> 🚫 If the Pixblasters controllers use the same power supply, video link HSO and HSI ports should be connected by Unshielded-Twisted-pair (UTP) cable! Notice the metal shield shown in **Figure 8.5** that explains differences between STP and UTP cables.
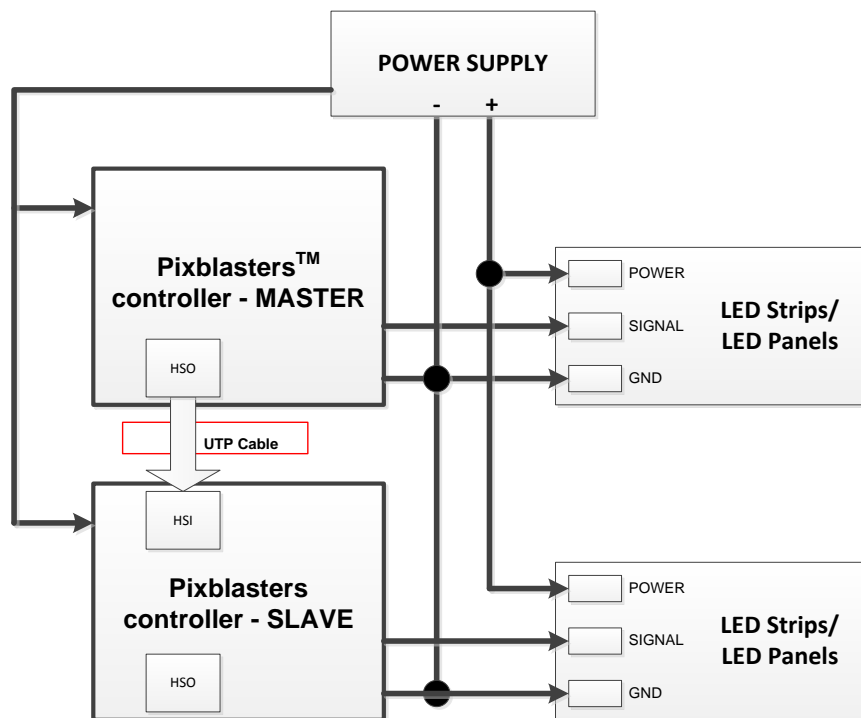
Figure 8.3 Single Power Supply in a Multi-Controller Topology

In larger systems, one power supply will most likely be insufficient to power all LEDs and MS1 video controllers. In that case, the power supply scheme may be as shown in **Figure 8.4**. Again, there is a rule covering HSO-HSI connections.

> 🚫 If the Pixblasters controllers do not use the same power supply, video link HSO and HSI ports should be connected by the Shielded-Twisted-pair (STP) cable! Notice the metal shield shown in **Figure 8.5** that explains differences between STP and UTP cables.
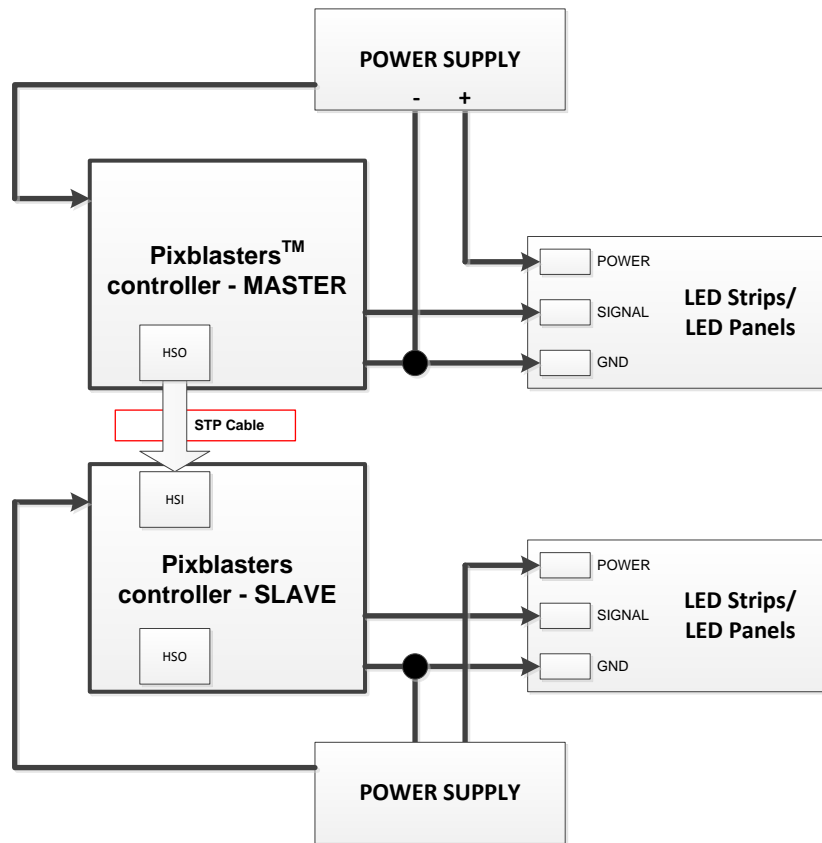
Figure 8.4 Multiple power supplies in Multi-controller topology

The power supply scheme shown in **Figure 8.2** is also applicable for multi-controller topologies. Again, if LED strips and Pixblasters controller use the same power supply, the Pixblasters controller should be connected to the power supply which powers the strip directly connected to the Pixblasters controller.

Figure 8.5 The STP (black) and the UTP (yellow) Cables

Some RGB LED strip types use +12 VDC, or even +24 VDC power supplies. As long as their expected data input voltage is +5 VDC, such LED strings can be successfully driven by the Pixblasters MS1 Video LED Controller.
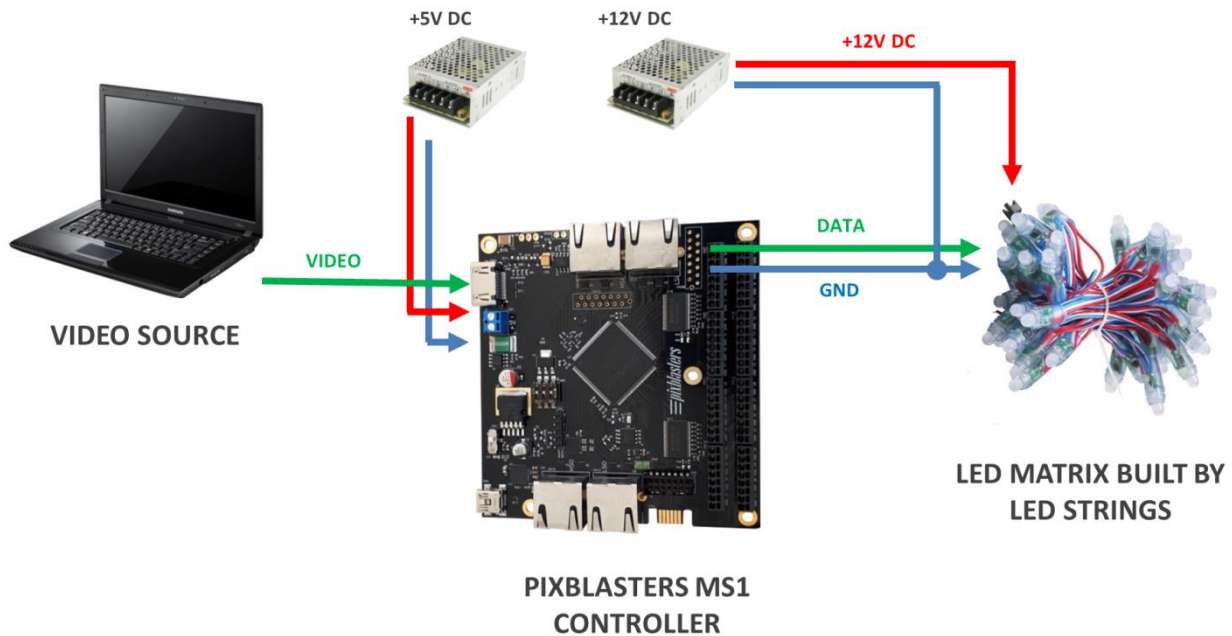
Figure 8.6 System Connections for LEDs with +12 VDC Input Voltage

> An example buildup of the demo display with WS2811 LEDs that use +12 VDC power supply: https://www.youtube.com/watch?v=o_5djzdhzHY&t=23s

> RGB LED strips that use higher power supply voltages enable longer strings with no additional power supplies, or no need to inserting additional power supply points along the string. Due to the higher voltage, lower currents flows through the string and the voltage drop is lower. It prevents LED "browning" that makes white colored LEDs to look brown due to the insufficient power supply.

9 USER CONTROL INTERFACE

Before the first use, or in a case of a significant change of the LED display controlled by an already initialized Pixblasters LED controller, the MS1 Video LED controller must be initialized through the embedded user's interface. Non-initialized LED controllers cannot display correct video.

The Pixblasters MS1 Video LED Controller controls very diverse LED display configurations and its control electronics must be programmed with the exact display's resolution, display's architecture (non-segmented, segmented, strips or LED panels, clock speed…), and with the exact part of the input video for the LED display. Additionally, by means of on-board DIP switches, the MS1 Video LED Controller must be configured for either master or slave operation modes

> The complete initialization process for use with WS2812B LED strips is described in this short video clip: https://youtu.be/6BcOaJHcRD8. Please set DIPs into 001 position, instead of 011 position shown in the video clip. Depending on the pre-programmed FW, some boards do not use 011 configuration. Currently used configurations are listed in the **Table 12.1: Selecting FPGA Configuration by DIP** Switches.
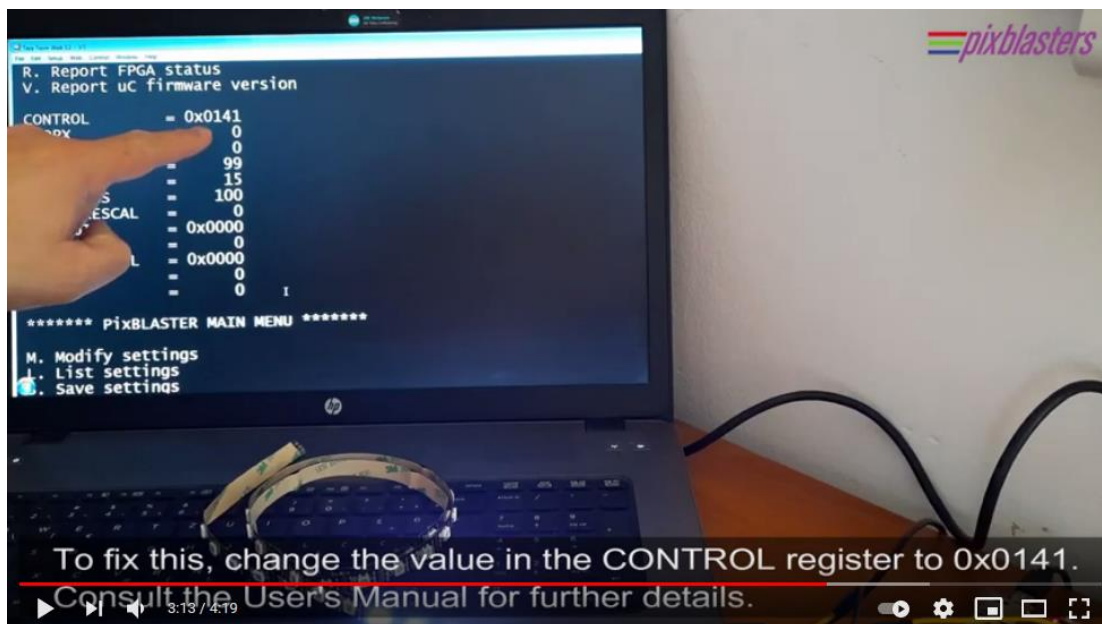
Figure 9.1 Video Clip Explaining the Pixblasters MS1 Initialization for LED Strips

> The complete initialization process for use with WS2812B RGB LED panels is described in this short video clip: https://www.youtube.com/watch?v=uhBz0K_IGW8. Please set DIPs into 101 position.
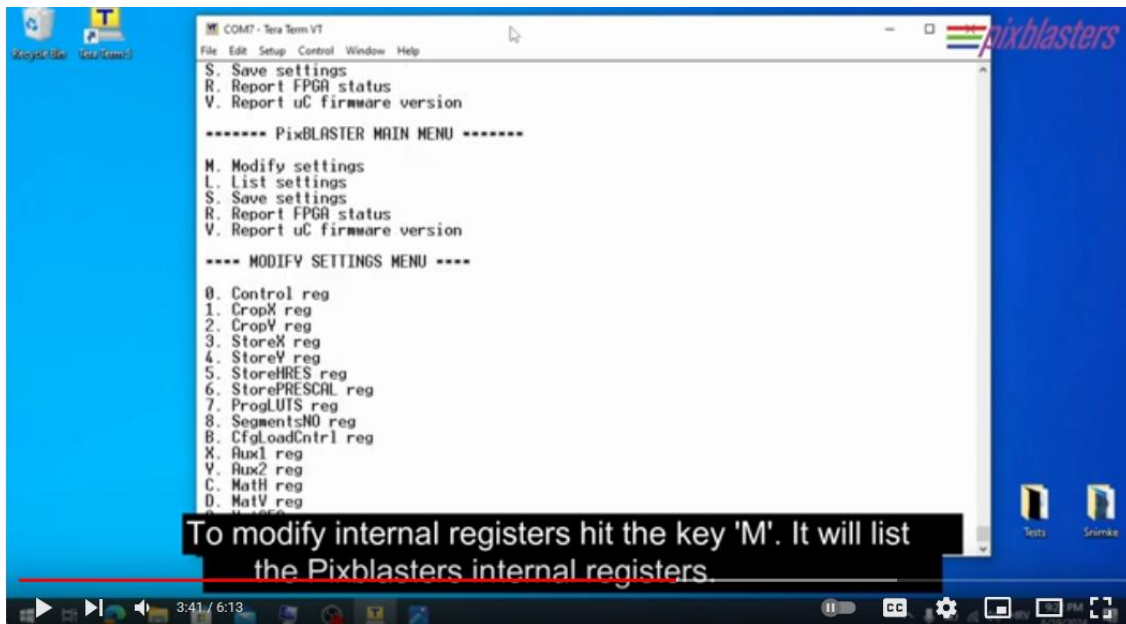
Figure 9.2 Video Clip Explaining the Pixblasters MS1 Initialization for LED Panels

9.1 Serial Connection to the Pixblasters MS1 Controller

To adjust the Pixblasters MS1 Video LED Controller to a specific display configuration, several controller parameters have to be set and permanently stored in the non-volatile flash memory. This procedure requires the USB serial connection between the Pixblasters controller and a PC, Mac or any other computer.

If the LED display utilizes multiple MS1 controllers, each controller has to be configured separately. Of course, if the configuration of the LED display never changes, the configuration process must be executed only once!

On Microsoft Windows PC machines, upon the Pixblasters controller connection, a new COM port will be visible in the **Control Panel\Device Manager**.

ℹ️   A tip from Pixblasters Mac user: https://www.decisivetactics.com/products/serial/

ℹ️   Application note PAPP002 explains the initialization process for LED strip video displays: http://pixblasters.com/wp-content/uploads/2021/05/papp002-Pixblasters-Quick-Start-Guide_v1.1.pdf.

ℹ️   Application note PAPP005 explains the initialization process for LED panel video displays: https://pixblasters.com/wp-content/uploads/2024/07/papp005-Pixblasters-Panels-Quick-Start-Guide_v1.0.pdf.
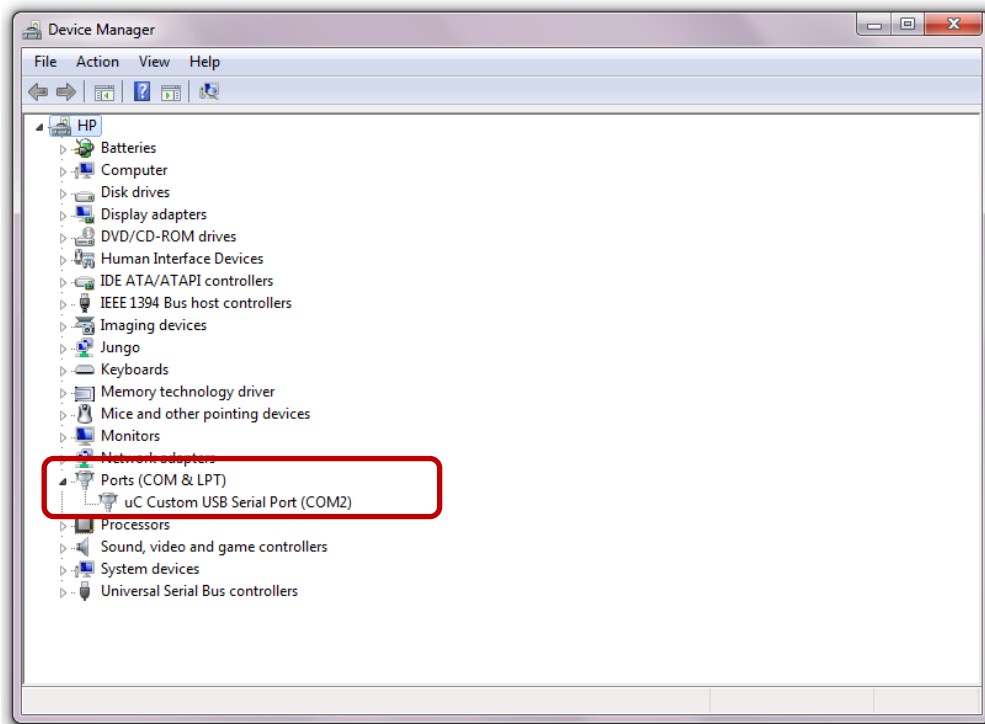
Figure 9.3 New USB Serial Connection Visible in the Windows Device Manager

The next step requires use of any terminal emulator application – the Pixblasters development team preference is the Tera Term, an open-source, free, software implemented, terminal emulator program. The Tera Term project home: https://teratermproject.github.io/index-en.html.

The Tera Term is used in all examples described within this document. Start the Tera Term (or equivalent program of your choice) and connect the Pixblasters controller through the related COM port (COM2 in this instance – see the Figure 9.3). The terminal emulator settings must be setup as shown in **Figure 9.4**.
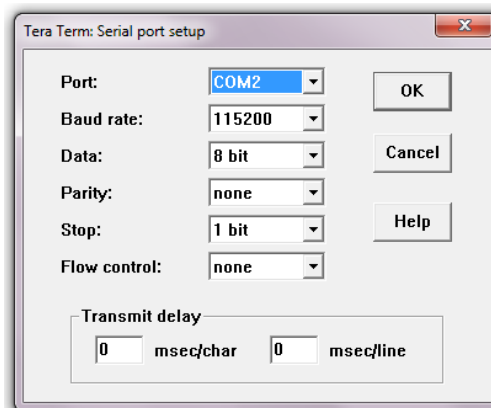
Figure 9.4 Serial Connection Parameters

After the established connection, hit the **Enter** key and the following menu should appear in the terminal emulator window:
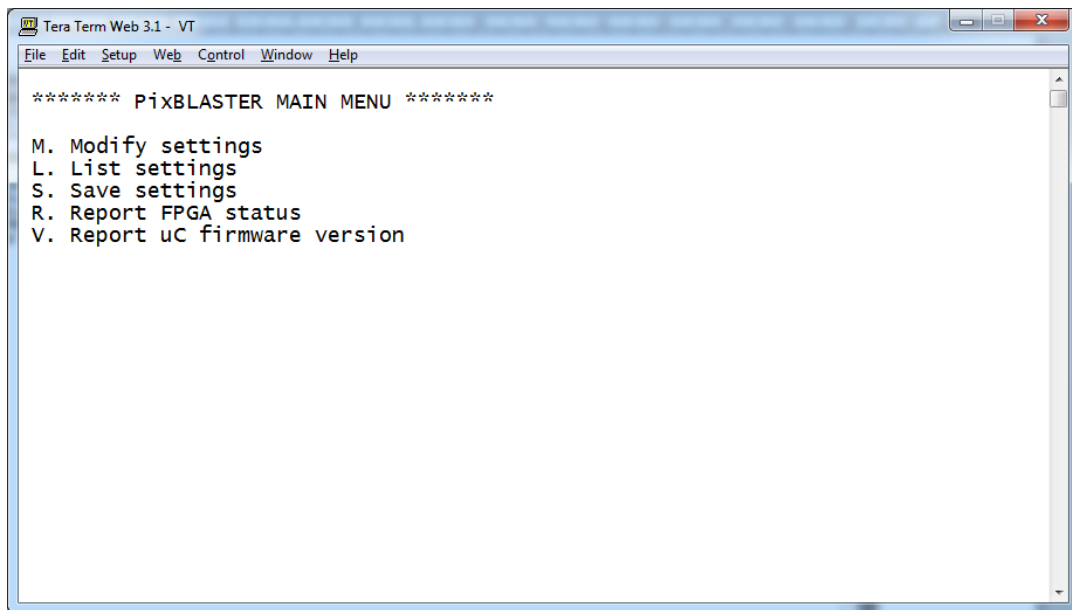
Figure 9.5 The Pixblasters Main Configuration Menu

9.2 Adjusting Pixblasters Controller Settings

Hit **M** or **m** key to enter the "Modify settings" menu, hit **E** or **e** to return to the main menu (Figure below).
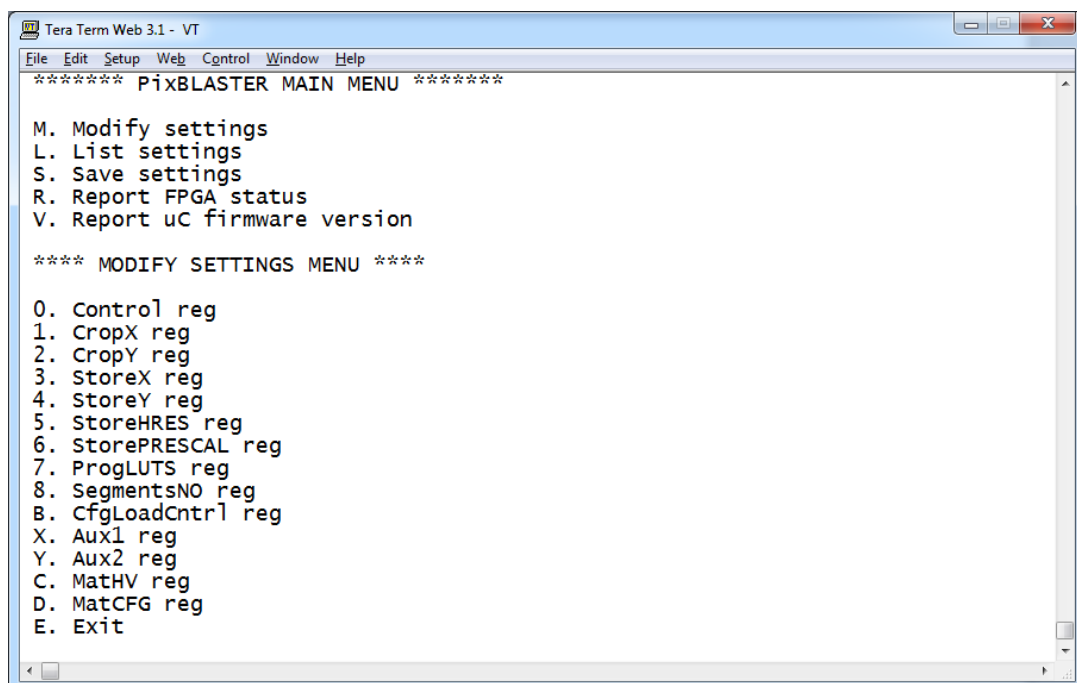
Figure 9.6 The Sub-menu for Internal Registers' Value Changes

⚠️ IMPORTANT: New register values will be active after the exiting from the "Modify settings" menu! To permanently store those values, please remember to save the configuration.

10 INTERNAL REGISTERS

10.1.1 Control Register Settings

Table 10.1: Control Register Description (Hexadecimal entry)

Bits	Name	Type	Reset value	Description
0	CONTROLLER ENABLE	R/W	1	Write:'0' – Controller disabled; '1' – Controller enabled Read: '0' – No valid video signal; '1' – Valid video detected The Pixblasters firmware automatically handles this bit depending on the valid input video signal. When changing the Control register's values, it is recommended to keep the current value of this bit.
1	MIRROR	R/W	0	**Valid for LED strips only!** '0' – Non-mirrored image '1' – Mirrored image For more information see the chapter Multi-Controller Topology.
3-2	PADDING	R/W	0	**Valid for LED strips only!** Used to set a number of LED strip pixels at the beginning of each strip that will not be used. First pixel of the video line will be sent to the LED strip pixel that follows the skipped LED strip pixels*.
5-4	LSEL	R/W	00	Pixel color order (low byte) "00" – Blue, "01" – Green, "10" – Red, "11" – Blue
7-6	MSEL	R/W	00	Pixel color order (middle byte) "00" – Green, "01" – Red , "10" – Blue, "11" – Green
9-8	HSEL	R/W	00	Pixel color order (high byte) "00" – Red, "01" – Green, "10" – Blue, "11" – Red
15-10	RESERVED	-	-	Do not change

This feature can be used in waterproof installations where strips are coming out from the sealed box. In such installations, the first strip pixels can be skipped to properly position the image on the visible display part.

<div align="center">**Table 10.2: Example Control Register Entries**</div>

Value (Hexadecimal)	Description
0x141	Start the controller. Colors pattern is GRB
0x001	Start the controller. Colors pattern is RGB
0x003	Start the controller. Image mirrored. Colors pattern is RGB
0x091	Start the controller. Colors pattern is RBG

HSEL, MSEL and LSEL bits determine the output pixel color order. Example: In order to change pixel color order from R (high byte), G (middle byte), B (low byte) to G (high byte), R (middle byte), B (low byte) pixel data, HLSEL, MSEL and LSEL bits must be programmed as follows:
LSEL[1:0] – "00"; MSEL[1:0] – "01"; HSEL[1:0] – "01"

10.1.2 CropX Register Settings

<div align="center">**Table 10.3: CropX Register Description (Decimal entry)**</div>

Bits	Name	Type	Reset value	Description
10-0	CROPX	R/W	0	Defines the exact X coordinate of the upper left corner of the cropping video window.
15-11	RESERVED	-	-	Do not change

For more information see chapter **11.1 Video Input Cropping**.

10.1.3 CropY Register Settings

<div align="center">**Table 10.4: CropY Register Description (Decimal entry)**</div>

Bits	Name	Type	Reset value	Description
10-0	CROPY	R/W	0	Defines the exact Y coordinate of the upper left corner of the cropping video window.
15-11	RESERVED	-	-	Do not change

For more information see chapter **11.1 Video Input Cropping**.

10.1.4   StoreX Register Settings

Table 10.5: StoreX Register Description (Decimal entry) – for LED STRIPS

Bits	Name	Type	Reset value	Description
9-0	STOREX	R/W	255	Defines the width of the cropping video window. The width is defined as STOREX + 1. It corresponds to the number of pixels in one horizontal line, even when one output drives multiple lines (segmented mode).
15-10	RESERVED	-	-	Do not change

Table 10.6: StoreX Register Description (Decimal entry) – for LED PANELS

Bits	Name	Type	Reset value	Description
9-0	STOREX	R/W	255	Defines the width of the cropping video window. It must be set as follows: (LED outputs per LED band * No. of panels * Panel Horizontal Res.) - 1
15-10	RESERVED	-	-	Do not change

For more information see chapter **11.1 Video Input Cropping**.

10.1.5   StoreY Register Settings

Table 10.7: StoreY Register Description (Decimal entry)

Bits	Name	Type	Reset value	Description
9-0	STOREY	R/W	31	Defines the height of the cropping video window. The width is defined as STOREY + 1.
15-10	RESERVED	-	-	Do not change

For more information see chapter **11.1 Video Input Cropping**.

10.1.6   StoreHRES Register Settings

Table 10.8: StoreHRES Register Description (Decimal entry) – for LED STRIPS

Bits	Name	Type	Reset value	Description
9-0	STOREHRES	R/W	512	Defines the length of the LED strip connected to a single LED output. In segmented displays it is a sum of all LEDs in multiple segments. In non-segmented displays it is a width of the display.
15-10	RESERVED	-	-	Do not change

Table 10.9: StoreHRES Register Description (Decimal entry) – for LED PANELS

Bits	Name	Type	Reset value	Description
9-0	STOREHRES	R/W	512	When driving LED panels, should be set to 512
15-10	RESERVED	-	-	Do not change

10.1.7 StorePRESCAL Register Settings

Defines the clock signal used to control the LED display. For more information see chapter **11.2 3-WIRE LED Display** Timing.

Table 10.10: StorePRESCAL Register Description – 3-WIRE RGB LEDs (Decimal entry)

Bits	Name	Type	Reset value	Description
9-0	STOREPRESCAL	R/W	0	0 – 800 kHz clock (DEFAULT); assures 60fps 1 – 400 kHz clock Defines the T_{SLOT} period (1 bit period) which is equal to the T_{STRIP}*(PRESCAL_REG + 1).
15-10	RESERVED	-	-	Do not change

Table 10.11: StorePRESCAL Register Description – 4-WIRE RGB LEDs (Decimal entry)

Bits	Name	Type	Reset value	Description
9-0	STOREPRESCAL	R/W	0	0 – 8 MHz clock; 1 – 4 MHz clock; **2 – 2 MHz (RECOMMENDED);** 3 – 1 MHz; 4- 0.5 MHz
15-10	RESERVED	-	-	Do not change

10.1.8 ProgLUTS Register Settings

In 3-wire LED configurations, this register is used for a selection of the LED bit encoding. In both, 3-wire and 4-wire LED configurations, bit 9 (PEM) is used to switch on and off the LVDA cable extender chip – 14 in **Figure 4.1 Pixblasters Controller Board Elements.**

Table 10.12: ProgLUTS Register Description – 3-WIRE RGB LEDs (Hexadecimal entry)

Bits	Name	Type	Reset value	Description
4 – 0	LUT_DATA	W	0x00	For selecting LED strip .driver type 0x00 - WS2812 0x01 - WS2811 0x10 - GS8208
8 – 5	RESERVED	W	0x0	Do not change.
9	PEM	W	0x0	For turning "On" and "Off" High-Speed port Pre-Emphasis. '0' – Off, '1' – On. For cables longer than 2 m, Pre-Emphasis should be turned "On".
15-10	RESERVED	-	-	Do not change

⚠ For driving the most popular 3-Wire WS2812B RGB LEDs, please setup LUT-DATA bits to 0b00000.

Table 10.13: ProgLUTS Register Description – 4-WIRE RGB LEDs (Hexadecimal entry)

Bits	Name	Type	Reset value	Description
8 – 0	RESERVED	W	0x00	Do not change.
9	PEM	W	0x0	For turning "On" and "Off" High-Speed port Pre-Emphasis. '0' – Off, '1' – On. For cables longer than 2 m, Pre-Emphasis should be turned "On".
15-10	RESERVED	-	-	Do not change

10.1.9 SegmentsNO Register Settings

Table 10.14: SEGMENTS Register Description (Decimal entry) – for LED STRIPS

Bits	Name	Type	Reset value	Description
9-0	SEGMENTS	R/W	0	Defines the number of output display lines driven by a single LED output. This number must comply with the following constraint: StoreHRES_REG * SEGMENTS_REG <= 512. Number of LED strip segments (display lines) = SEGMENTS. For more information see chapter **LED Strips Segmented Single Controller Topology**.
15-10	RESERVED	-	-	Do not change

Table 10.15: SEGMENTS Register Description (Decimal entry) – for LED PANELS

Bits	Name	Type	Reset value	Description
9-0	SEGMENTS	R/W	0	Defines the number of outputs that drive one LED band. It MUST BE divisible by two: 2, 4, 6…28, 30, 32
15-10	RESERVED	-	-	Do not change

10.1.10 CfgLoadCntrl Register Settings

Table 10.16: CfgLoadCntrl Register Description (Hexadecimal entry)

Bits	Name	Type	Reset value	Description
15-0	RESERVED	-	-	Do not change

> ⚠️ This is the register reserved for the service needs. Please do not use this register and do not alter its content.

10.1.11 Aux1 Register Settings

Table 10.17: Aux1 Register Description – 3-WIRE RGB LEDs (Hexadecimal value)

Bits	Name	Type	Reset value	Description
4-0	LED Brightness	R/W	0x1F	Defines the LED brightness. 0 is the minum and 31 is the maximum LED brightness.
5	RESERVED	-	-	Do not change
6	Gamma Bypass	R/W	0x0	0 – internal gamma correction tables used 1 – internal gamma correction bypassed
7	Gamma Select	R/W	0x0	0 – PC Gamma compensation 1 – MAC Gamma compensation
15-7	RESERVED	-	-	Do not change

Table 10.18: Aux1 Register Description – 4-WIRE RGB LEDs (Hexadecimal value)

Bits	Name	Type	Reset value	Description
4-0	LED Brightness	R/W	0x1F	Defines the LED brightness. 0 is the minum and 31 is the maximum LED brightness.
5	Continuous Clock	R/W	0x0	on/off continuous clock. By the 4-wire protocol, continous clock is not needed. However, the continous clock might relax transients in long wires and remove the unwanted flickering.
6	Gamma Bypass	R/W	0x0	0x0 – internal gamma correction used 0x1 – internal gamma correction bypassed
15-7	RESERVED	-	-	Do not change

10.1.12 Aux2 Register Settings

> ⚠️ 3-Wire WS2812B-like RGB LED types do not use this register!

Table 10.19: Aux2 Register Description – 4-WIRE RGB LEDs (Decimal entry)

Bits	Name	Type	Reset value	Description
4-0	Aux2	R/W	0	Defines the number of trailing 1s at the end of each line. The number of 1s is calculated as: Aux2 x 32; for example if the register value is 17, the number of trailing 1s is 17 x 32 = 544. Max. value for Aux2 is 31
15-5	RESERVED	-	-	Do not change

10.1.13 Mat_H Register Settings

This register is used when driving LED panels only. It defines the horizontal resolution of used LED panels.

Table 10.20: Mat_H Register Description (Decimal entry)

Bits	Name	Type	Reset value	Description
5-0	MATRIX_HOR	R/W	0	LED panel horizontal resolution – 1. For example, for 32x8 panel it is 31.
15-6	RESERVED	-	-	Do not change

10.1.14 Mat_V Register Settings

This register is used when driving LED panels only. It defines the vertical resolution of used LED panels.

Table 10.21: Mat_V Register Description (Decimal entry)

Bits	Name	Type	Reset value	Description
5-0	MATRIX_VER	R/W	0	LED panel vertical resolution – 1. For example, for 32x8 panel it is 7.
15-6	RESERVED	-	-	Do not change

Table 10.22: Example Mat_H and Mat_V Register Entries

Mat_H Value	Mat_V Value	Description
7	7	LED panel resolution 8x8
15	15	LED panel resolution 16x16
31	7	LED panel resolution 32x8
21	21	LED panel resolution 22x22
43	10	LED panel resolution 44x11

10.1.15 Mat_CFG Register Settings

This register is used when driving LED panels only. It defines the type of the used LED panel and the maximum number of panels per LED output.

Table 10.23: MatCFG Register Description (Hexadecimal entry)

Bits	Name	Type	Reset value	Description
3-0	MATRIX_DIST	R/W	0	Number of LED panels driven by a single LED output - 1. For example, for 8x8 panels it is 7
6-4	MATRIX_TYPE	R/W	0	LED panel type: - 000 – 8x8 - 001 – 16x16 - 010 – 32x8 - 011 – 22x22 - 100 – 44x11
15-7	RESERVED	-	-	Do not change

Table 10.24: Example Mat_CFG Register Entries

Value (Hexadecimal)	Description
0x07	LED panel resolution 8x8, 8 panels driven by a single output
0x11	LED panel resolution 16x16, 2 panels driven by a single output
0x21	LED panel resolution 32x8, 2 panels driven by a single output
0x40	LED panel resolution 44x11, 1 panel driven by a single output
0x30	LED panel resolution 22x22, 1 panel driven by a single output

11 OPERATIONAL DETAILS

> **For LED Strips** - Make sure to check the PAPP002 application note – the quick start guide: http://pixblasters.com/wp-content/uploads/2021/05/papp002-Pixblasters-Quick-Start-Guide_v1.1.pdf. A short 3-min video clip explains everything needed to successfully initialize and start the Pixblasters MS1 Video LED Controller.

> **For LED Panels** - Make sure to check the PAPP005 application note – the quick start guide: https://pixblasters.com/wp-content/uploads/2024/07/papp005-Pixblasters-Panels-Quick-Start-Guide_v1.0.pdf. A short 6-min video clip explains everything needed to successfully initialize and start the Pixblasters MS1 Video LED Controller.

11.1 Video Input Cropping

Cropping window position and sizing are programmable through the following registers: X_CROP, Y_CROP, X_STORE and Y_STORE. The following figure explains how to define the video cropping window.
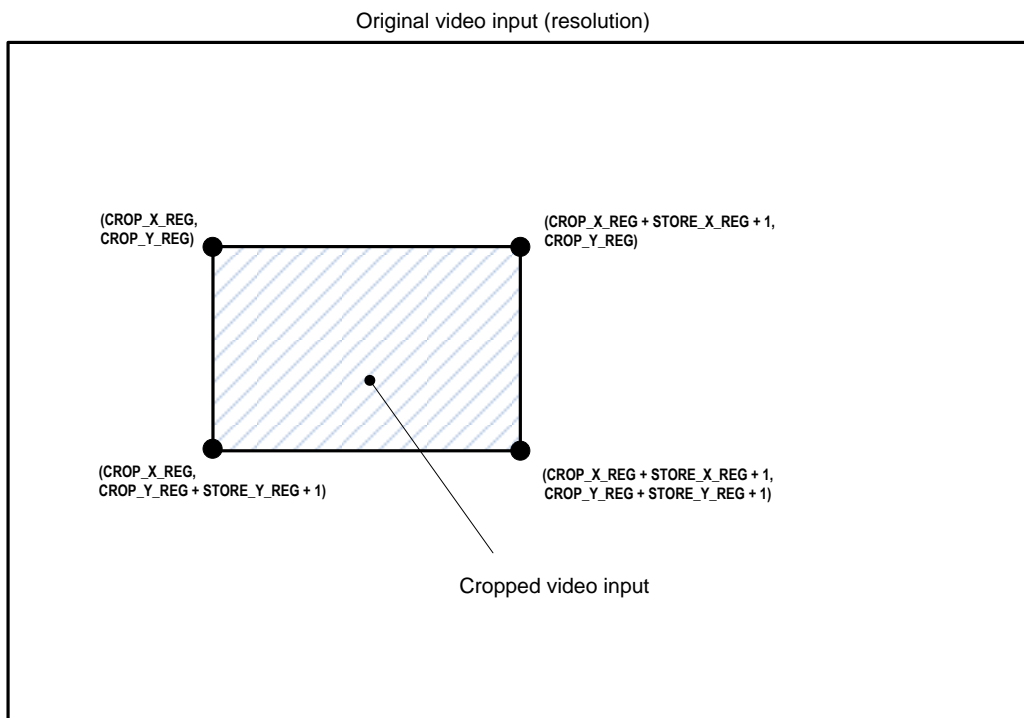
Figure 11.1 Definition of the Cropping Window Programming

Select **M. Modify Settings** (**Figure 9.6**) and setup the Cropping window position and sizing through the following registers: X_CROP, Y_CROP, X_STORE and Y_STORE. Once programmed, the configuration data will be permanently stored in the PIC18F26J60 microcontroller's internal flash memory, and at the next power up, the LED display will show the programmed image window.

11.2 3-WIRE LED Display Timing

> ℹ️ This also applies to the supported LED panels based on WS2812B RGBLEDs.

Figure 11.2 explains relationship between the Pixblasters controller's internal clock (f_{STRIP}), and the timing at the 3-wire WS2812B-like LED control outputs. f_{STRIP} is fixed to 25.6MHz. StorePRESCAL register should be set such that correct T_{SLOT} interval for the specific LED strip type is generated ($T_{SLOT} = T_{STRIP}*(PRESCAL\_REG + 1)$ where $T_{STRIP} = 1/f_{STRIP}$).

The NEW_PIXEL is fixed to 24 T_{SLOT} periods. This setup enables serial shift of the RGB888 or similar pixel formats to correctly drive the LED pixel.
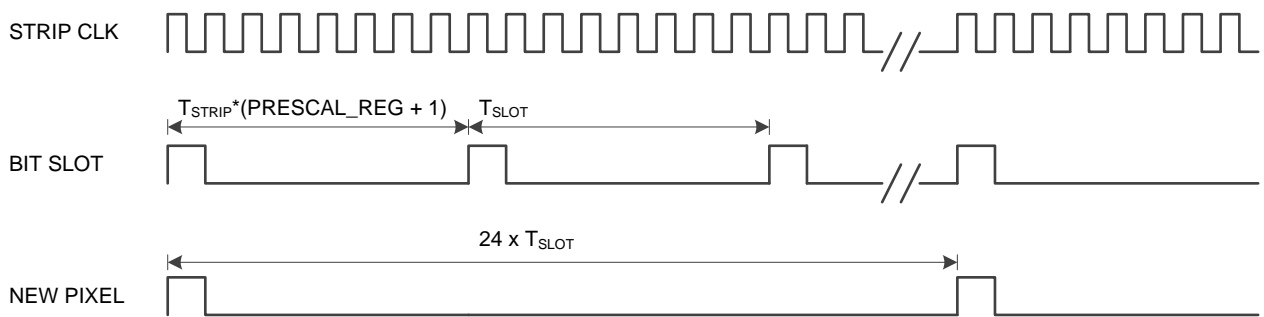
Figure 11.2 LED display clocking (3-wire LEDs)

11.3 3-WIRE WS2812B LED Pixel Bit Value Encoding

The Pixblasters MS1 controller uses the following bit value coding to drive 3-wire WS2812B-like RGB LEDs – please see the **Figure 11.3**. To activate this bit encoding, write LUT_DATA to 0x01 (**Table 10.12: ProgLUTS Register Description – 3-WIRE RGB LEDs**.

Bit encoding optimized for:
- T_{0H} = 0.35µs; T_{0L} = 0.8 µs
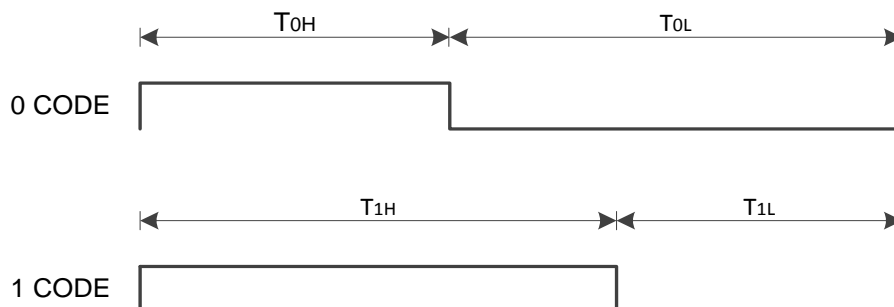- T_{1H} = 0.7µs; T_{1L} = 0.6 µs

Figure 11.3 Control 0 and 1 data encoding

11.4 3-WIRE GS8208B LED Pixel Bit Value Encoding

The Pixblasters MS1 controller uses the following bit value coding to drive 3-wire GS8208B LEDs. It is not compatible with the WS2812B encoding. T is 1/800kHz for both, GS8208G and WS2812B LEDs. To activate this bit encoding, write LUT_DATA to 0x10 (**Table 10.12: ProgLUTS Register Description – 3-WIRE RGB LEDs**.
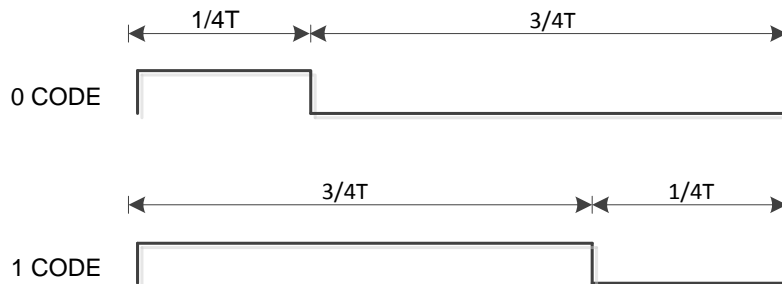

Figure 11.4 GS8208B Control 0 and 1 data encoding

11.5 3-WIRE WS2811 LED Pixel Bit Value Encoding

The Pixblasters MS1 controller uses the following bit value coding to drive 3-wire WS2811-like RGB LEDs. To activate this bit encoding, write LUT_DATA to 0x00 (**Table 10.12: ProgLUTS Register Description – 3-WIRE RGB LEDs**. This bit encoding is very similar to WS2812B bit encoding, but depending on the LED, the timing can be slightly different.

Bit encoding optimized for:
- T_{0H} = 0. 5µs; T_{0L} = 2.0 µs
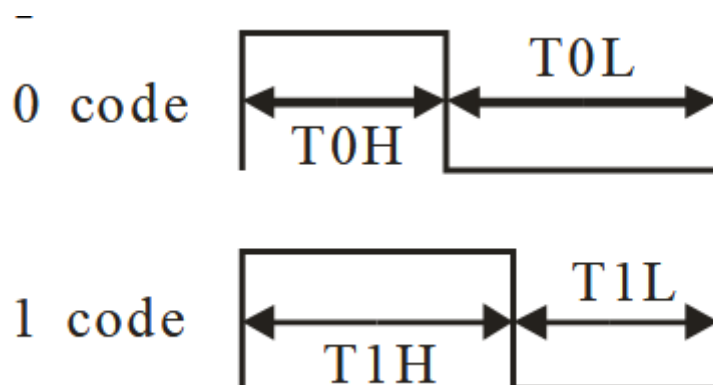- T_{1H} = 1.2µs; T_{1L} = 1.3 µs

Figure 11.5 GS8208B Control 0 and 1 data encoding

11.6 4-WIRE LED Display Timing

4-wire APA102-like addressable RGB LEDs can be controlled through the standard 4-wire LED interface that consists of the DATA/CLOCK pair, power and ground lines. The separated clock line makes the APA102 LEDs less timing sensitive than 3-wire LEDs, such as the WS2812B LED with the clock embedded within the data signal.

This feature makes them very popular among software developers and enables control of reasonable number of LEDs within multitasking operating system. However, applications with thousands of LEDs should be controlled by customized hardware peripherals – such as the Pixblasters MS1 Video LED Controller.

Figure 11.6 explains the APA102 interface. It is completely handled by the Pixblasters MS1 LED controller and the user does not have to care about the serial protocol details. To enable the APA102-like driving capabilities, user has to select the proper configuration from the
Table 12.1: Selecting FPGA Configuration by DIP Switches. The clocking frequency must be setup in the StorePRESCAL register – we recommend 2 MHz driving clock because it assures flicker-free 60 fps LED video refresh.
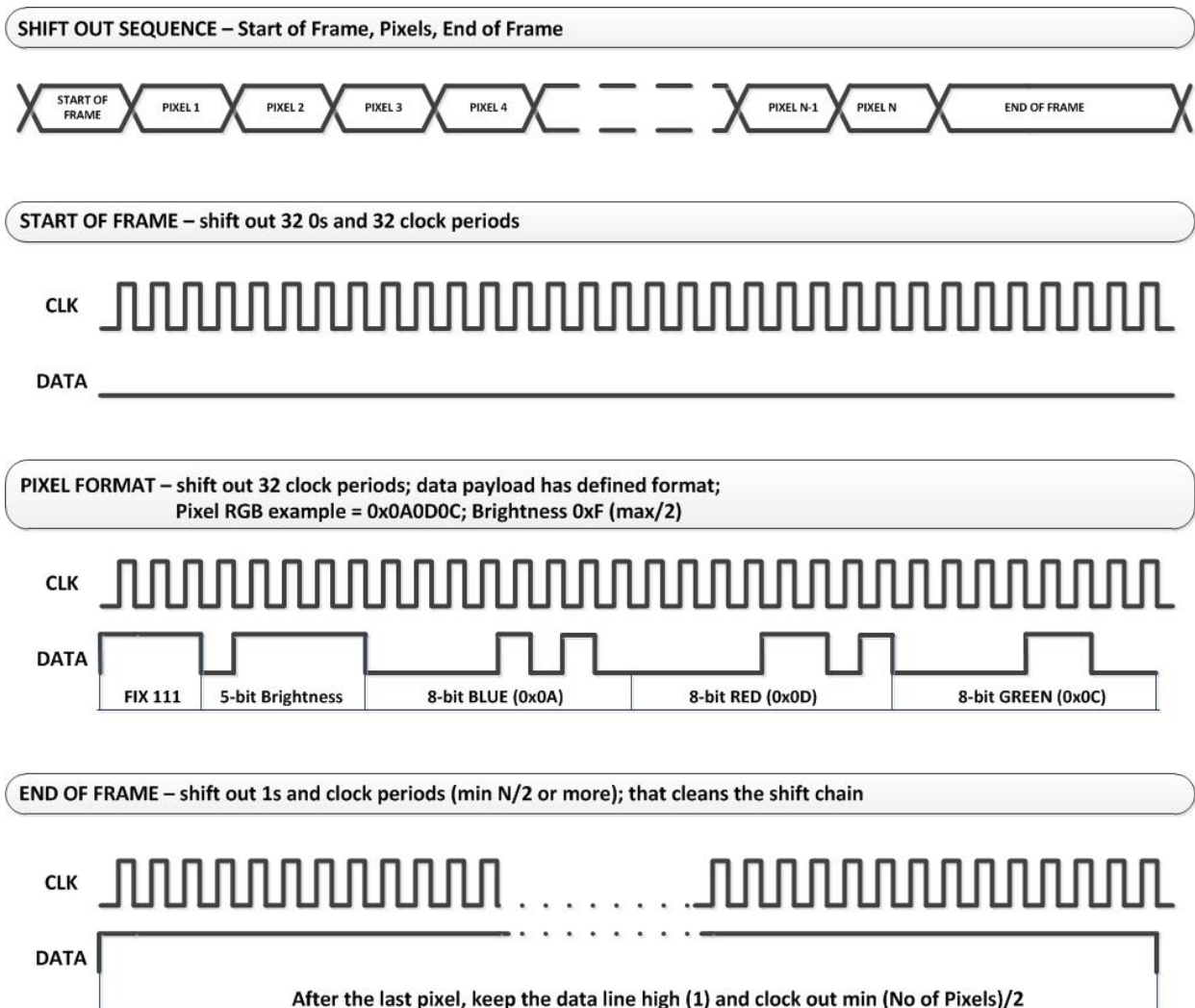

Figure 11.6 APA102 Control Interface Explained

11.7 Listing Pixblasters Controller Settings

To check configured controller's settings, hit **L** or **I** key and the following should appear:

```
M. Modify settings
L. List settings
S. Save settings
R. Report FPGA status
V. Report uC firmware version

CONTROL      = 0x0140
CROPX        =      0
CROPY        =      0
STOREX       =    255
STOREY       =     31
STOREHRES    =    512
STOREPRESCAL =      0
PROGLUTS     = 0x0000
SEGMENTSNO   =      0
CFGLOADCNTRL = 0x0000
AUX1         = 0x001F
AUX2         =      0
MAT_H        =     15
MAT_V        =     15
MAT_CFG      = 0x0011
```

Figure 11.7 Check-list of Programmed Internal Registers' Values (Default state)

> Please note that the actual values may be different from those shown in **Figure 11.7** since they reflect a specific controller's configuration. The MS1 controller set to configuration 1 (Master for LED strips) shows the defaults with MAT_H = MAT_V = MAT_CFG = 0 because this configuration does not have LED panel specific registers.

CONTROL	0x0041
CROPX	0
CROPY	0
STOREX	255
STOREY	31
STOREHRES	256
STOREPRESCAL	0x0000
PROGLUTS	0x0000
SEGMENTSNO	0
CFGLOADCTRL	0x0000
AUX1	0x001F
AUX2	0
MAT_H	15
MAT_V	15
MAT_CFG	0x0011

Table 11.1: Default (Pre-Programmed) Registers Values with Activated Configuration 5 for Driving LED Panels

11.8 Saving Pixblasters Controller Settings

"Save settings" menu is accessed by **S** or **s** key. Hit **Y** or **y** to save settings, or **N** or **n** to exit.
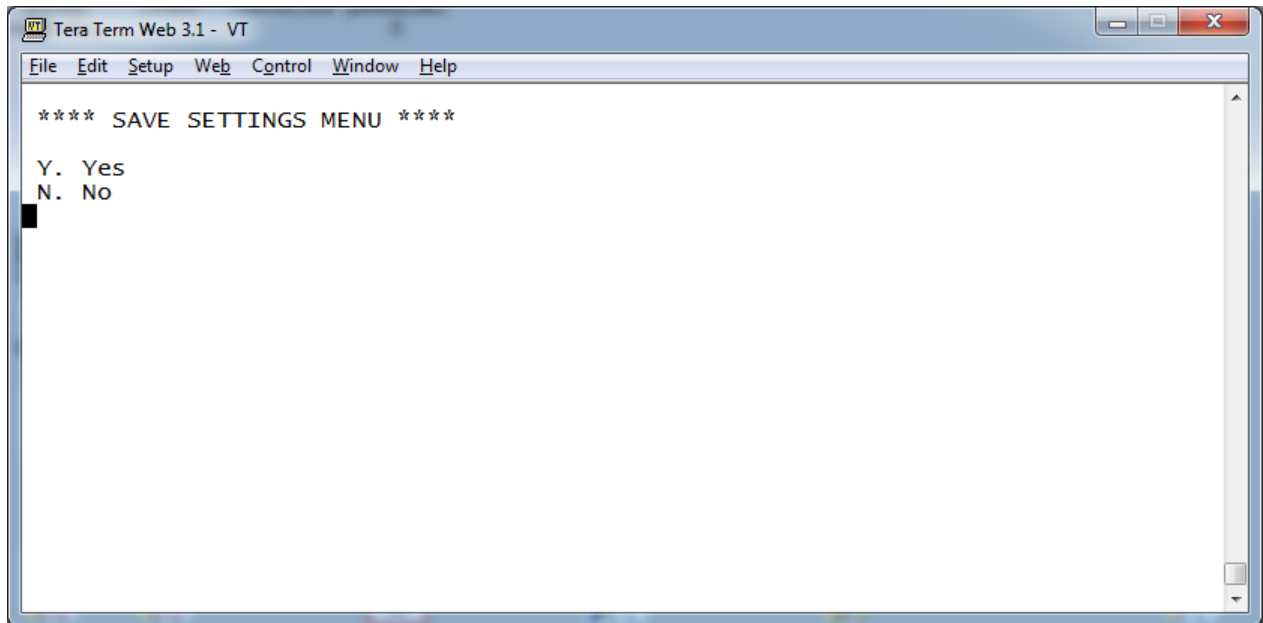

Figure 11.8 Permanently Save the Settings into Pixblasters MS1 Controller

11.9 Report FPGA Status

This menu is used for checking the FPGA status. Hit **R** or **r** to access the FPGA status. The following log shall appear:
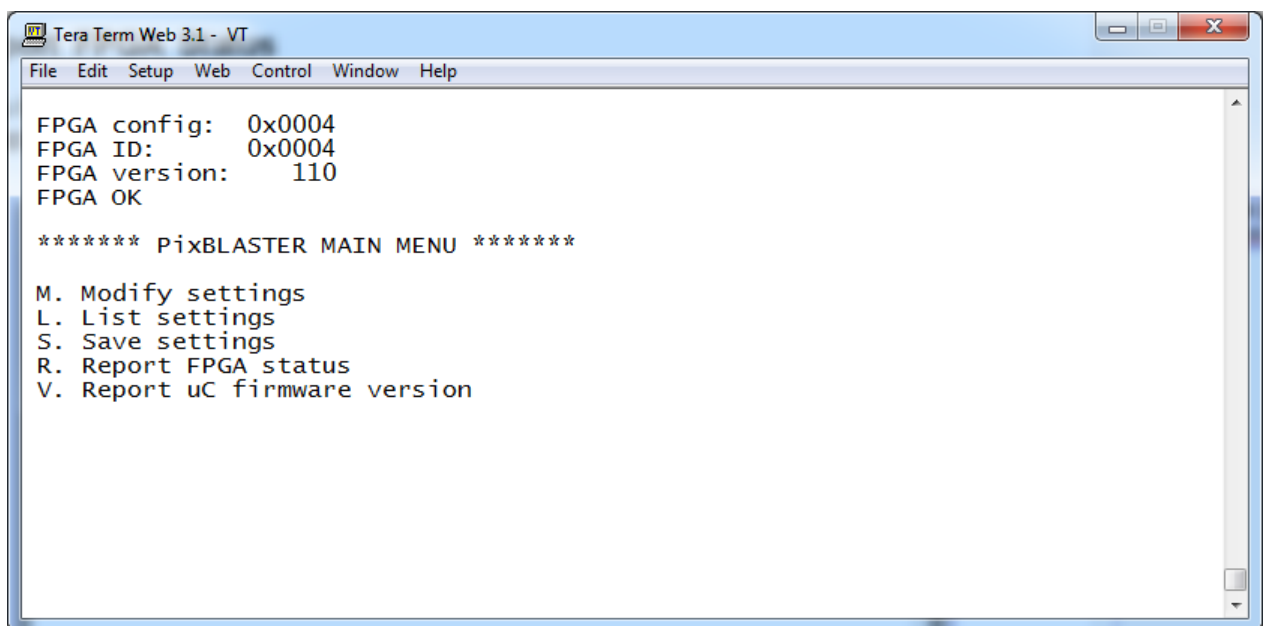

Figure 11.9 Check the FPGA Status

12 UPDATING FPGA CONFIGURATION

> ℹ️ For more details and a short instructional video, please check the PAPP003 application note – Pixblasters FPGA Update Guide: https://pixblasters.com/wp-content/uploads/2021/09/papp003-Pixblasters-FPGA-Update-Guide_v1.0.pdf
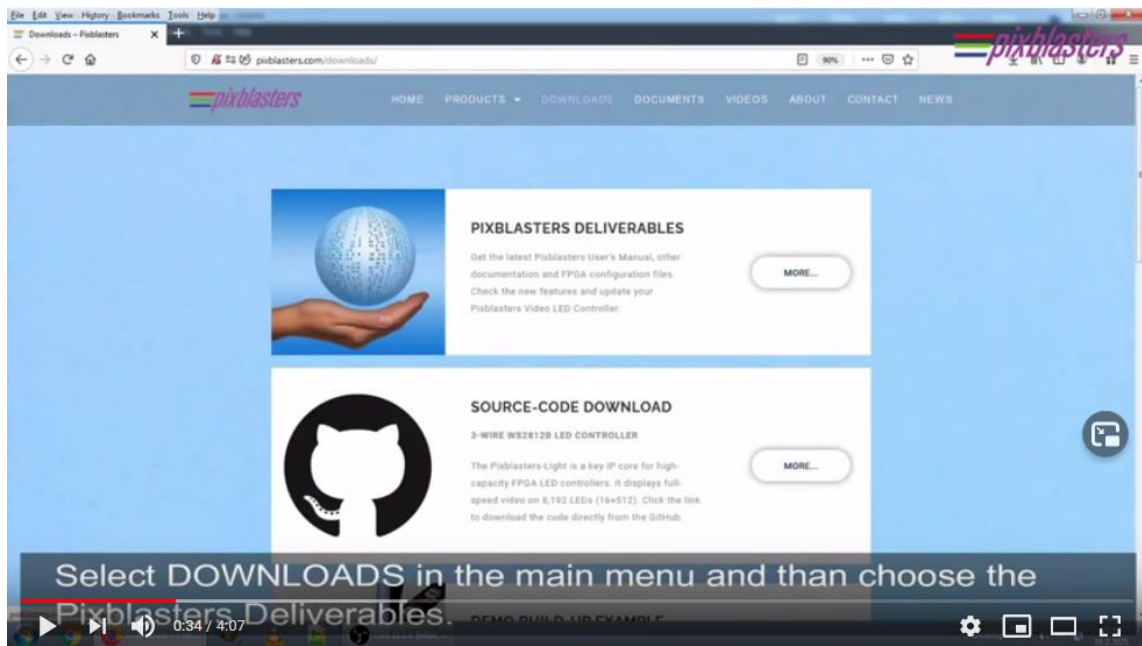

Figure 12.1 FPGA Update Guide – Instructional Video (Error! Hyperlink reference not valid.**)**

In order to be able to perform FPGA configuration update, the FPGA configuration '0' should be loaded first. That is accomplished by setting on-board DIP switches to 000, as shown in **Figure 12.2**.
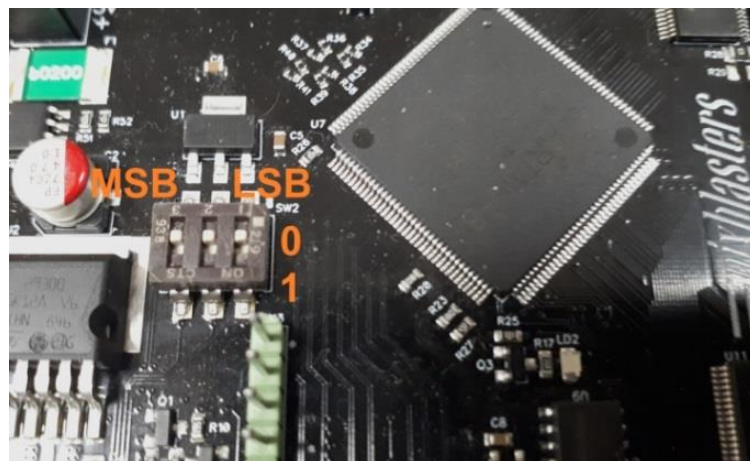

Figure 12.2 DIP switch – FPGA Configuration Selector

Table 12.1: Selecting FPGA Configuration by DIP Switches

Config. No.	DIPs [MSB, b1, LSB]	Description
Config. 0	000	Load configuration that enables FPGA configurations change
Config. 1	001	3-wire (WS2812B-like) Master Pixblasters MS1 configuration with 32 active LED strip outputs, each driving 512 pixels. Connect the video input to HDMI connector.
Config. 2	010	3-wire (WS2812B-like) Slave Pixblasters MS1 configuration with 32 active LED strip outputs, each driving 512 pixels. Connect the video input to HIS RJ45 connector.
Config. 3	011	4-wire (APA102-like) Master Pixblasters MS1 configuration with 16 active LED strip outputs, each driving 1024 pixels. Connect the video input to HDMI connector.
Config. 4	100	4-wire (APA102-like) Slave Pixblasters MS1 configuration with 16 active LED strip outputs, each driving 1024 pixels. Connect the video input to HIS RJ45 connector.
Config. 5	101	LED panels driving configuration – Master configuration. Connect the video input to HDMI connector.
Config. 6	110	LED panels driving configuration – Slave configuration. Connect the video input to HIS RJ45 connector.
Config. 7	111	LED panels – pixel mapping control matrix.

> *i* The latest versions of the User's Manual and FPGA configuration files can be found at: www.pixblasters.com/downloads/deliverables.

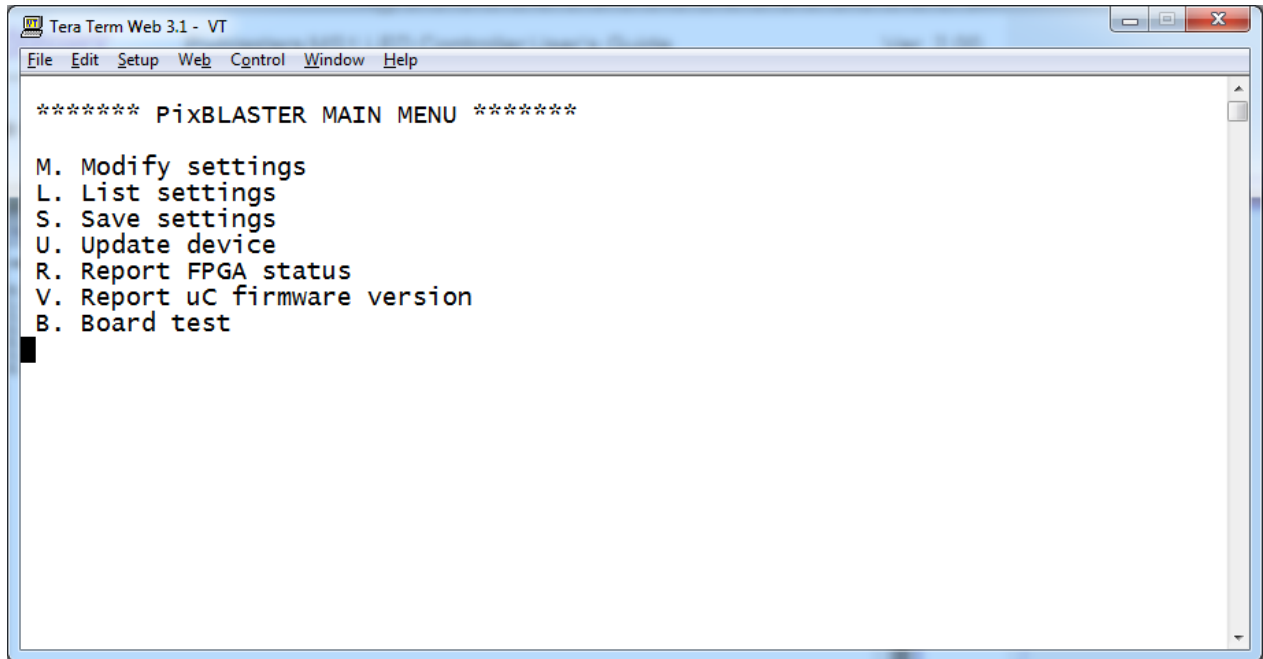When the configuration '0' is loaded, the following screen should appear:

Figure 12.3 Prepare for the FPGA Configuration Update

Hit **U** or **u** key to enter the "Update device" menu. The following shall appear:
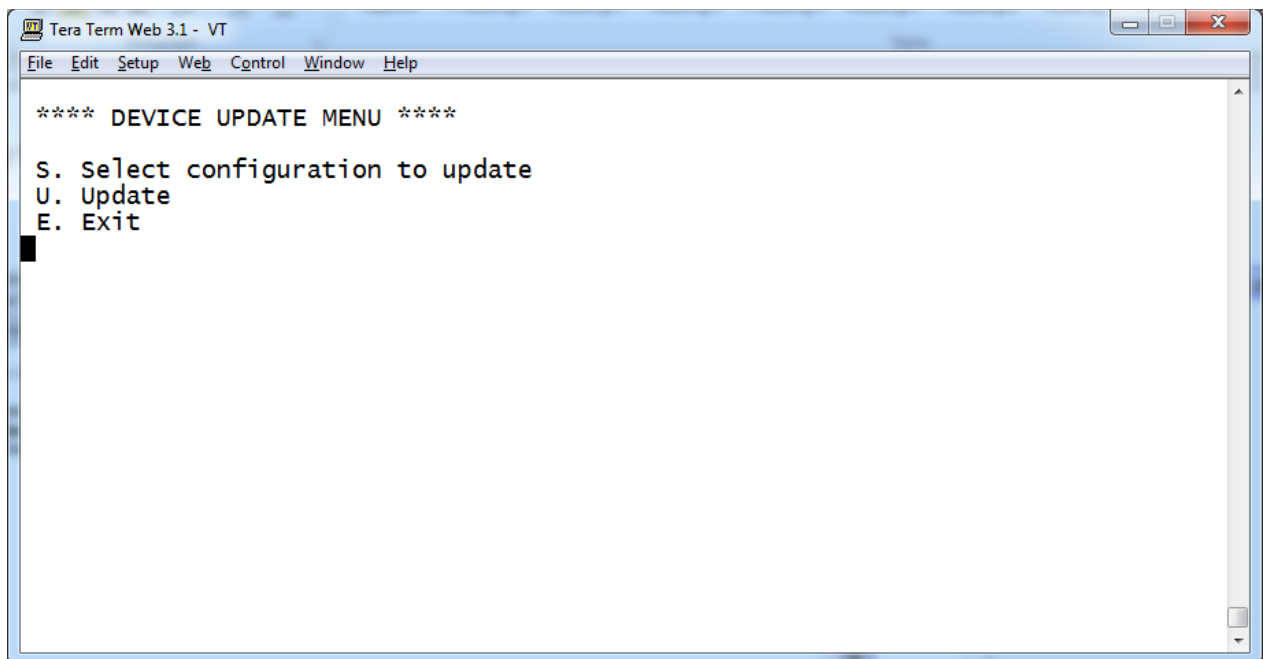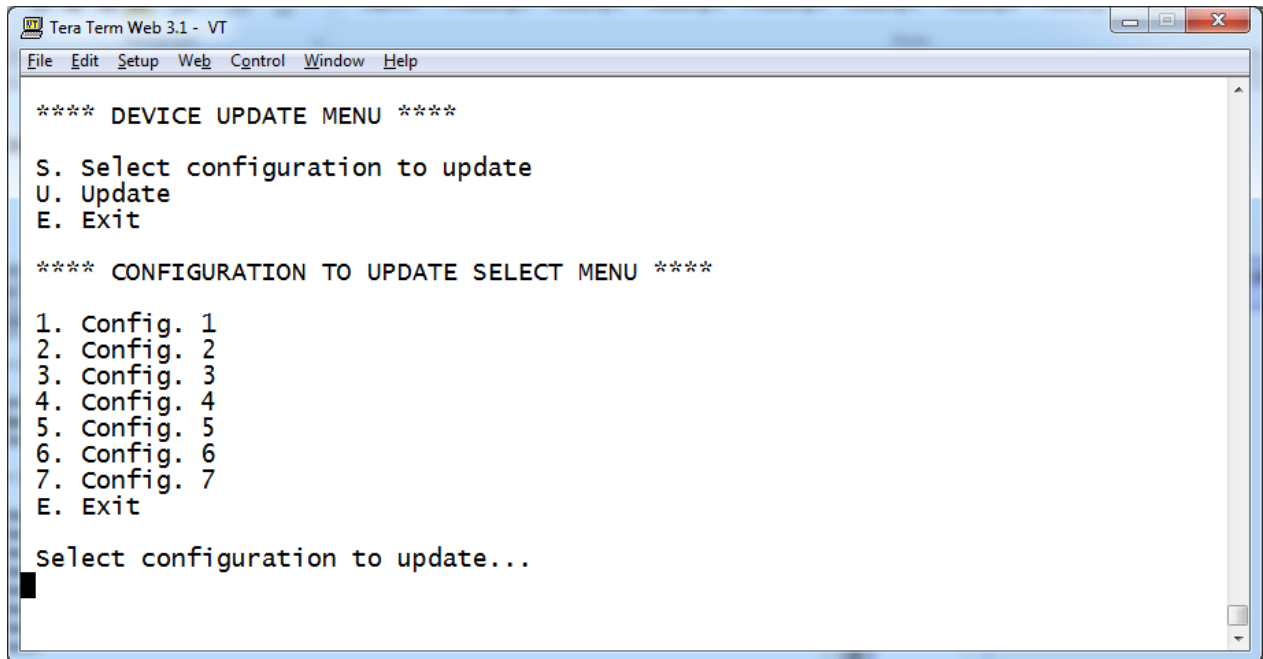

Figure 12.4 The FPGA Configuration Update Menu

Hit **S** or **s** to select configuration you wish to update. The following will appear:
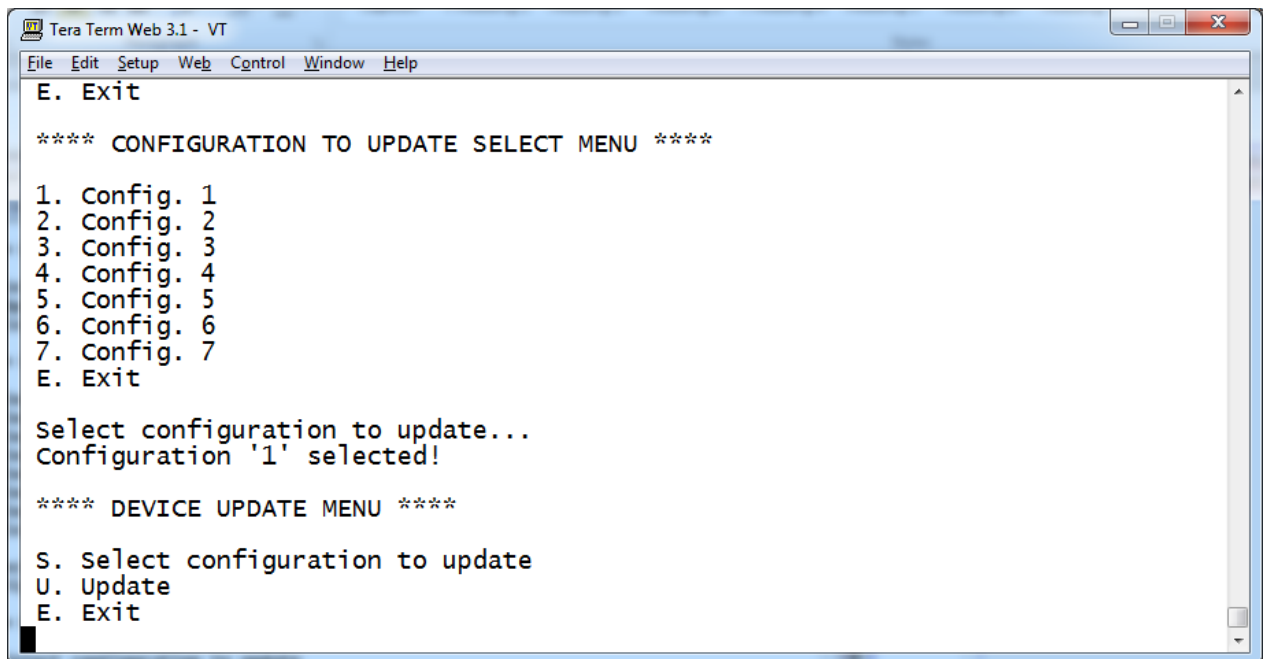
Figure 12.5 Select the FPGA Configuration for Update

Hit a number corresponding to a configuration you wish to update. If, for example, you select configuration '1' you should see the following:
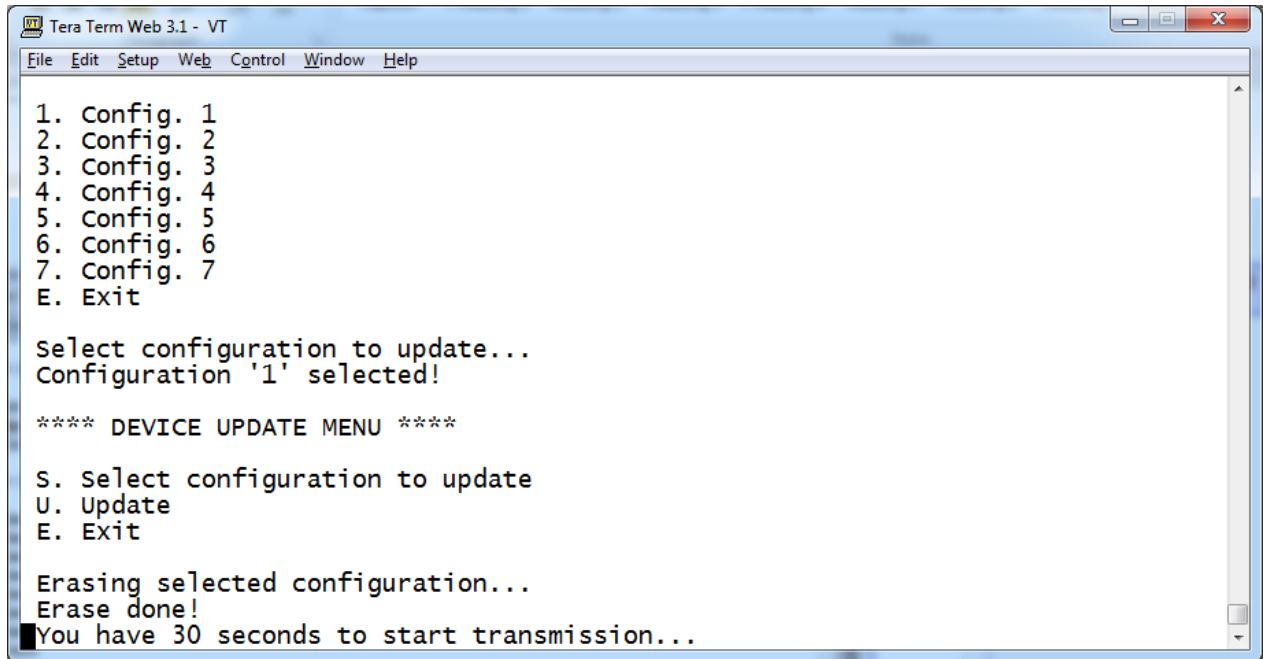
Figure 12.6 The FPGA Configuration for Update Selected

Hit **U** or **u** to start update procedure. The following shall appear:

Figure 12.7 Erasing the FPGA Configuration for the Update
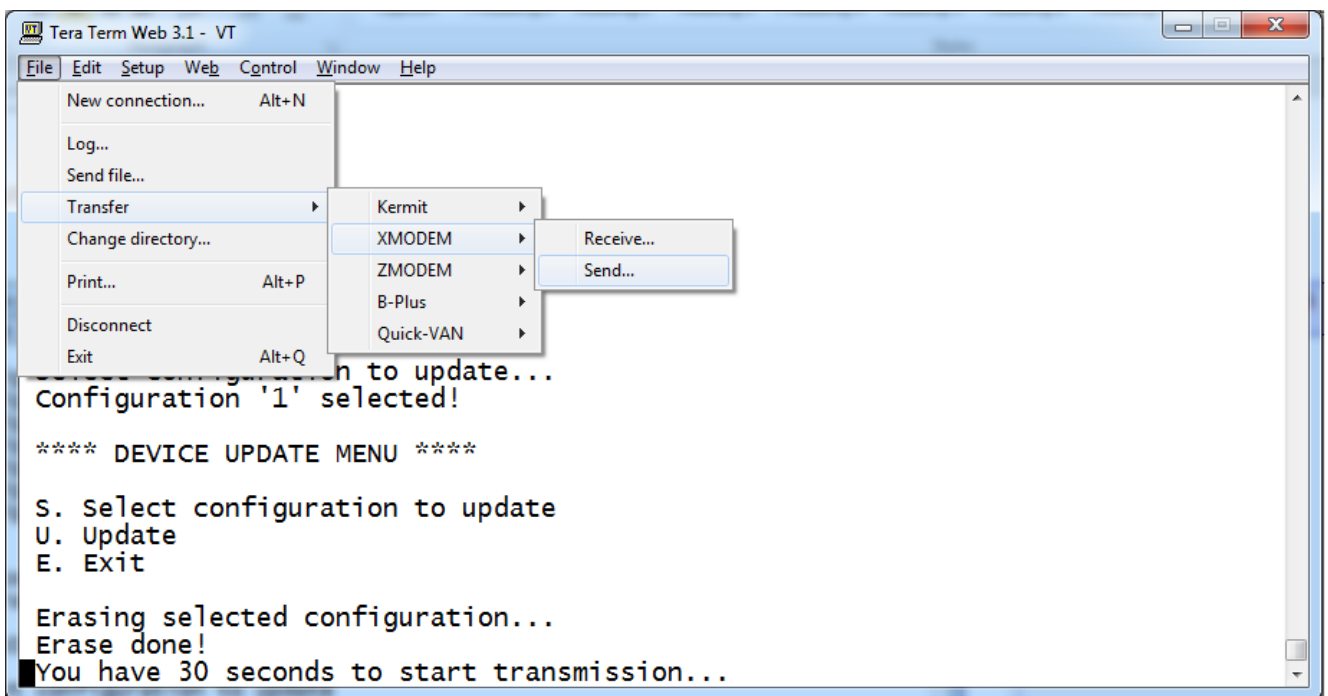
This message means that selected configuration has been erased, and that you have 30 seconds to select and start transmitting a file containing a new configuration.

File transmission is accomplished through the XMODEM protocol. If the Tera Term emulator is used, the XMODEM transmission is selected in the following way:

Figure 12.8 Setup the XMODEM Protocol in the Tera Term Serial Emulator

Then select the update .bin FPGA configuration file.
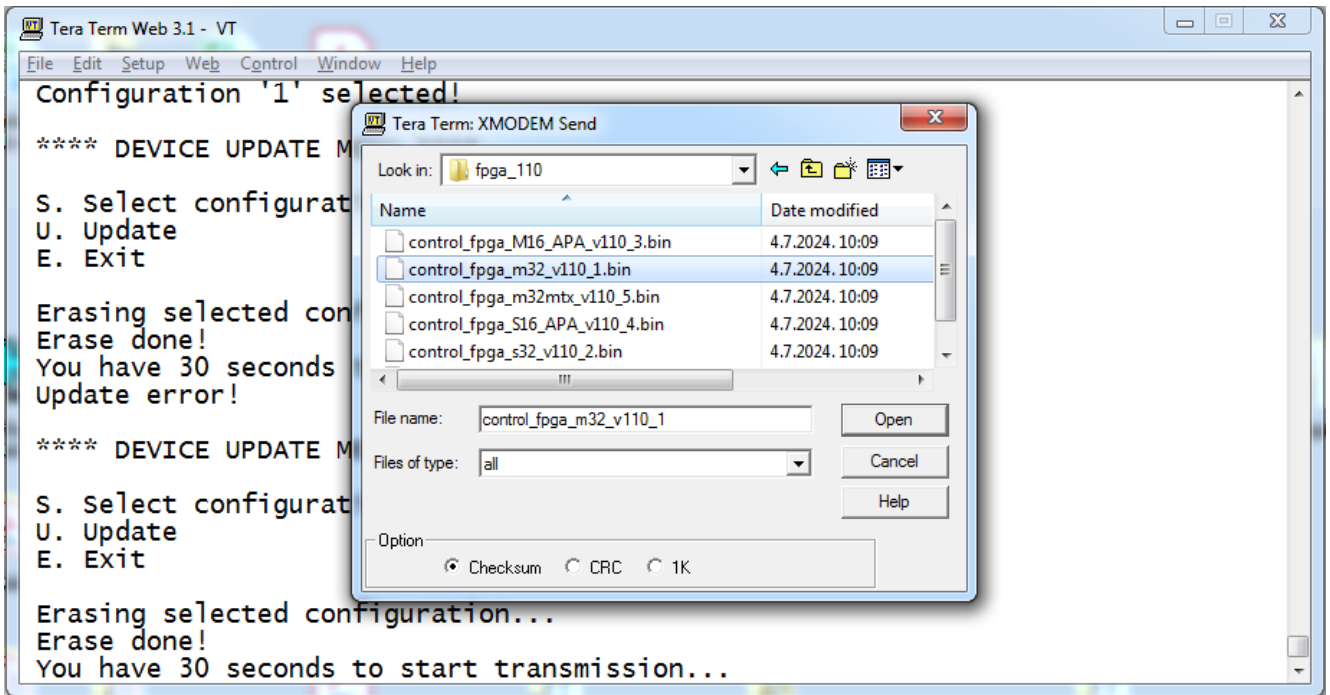
Figure 12.9 New FPGA Configuration Update File Selected

If selection was done within 30 seconds, the file update will start.
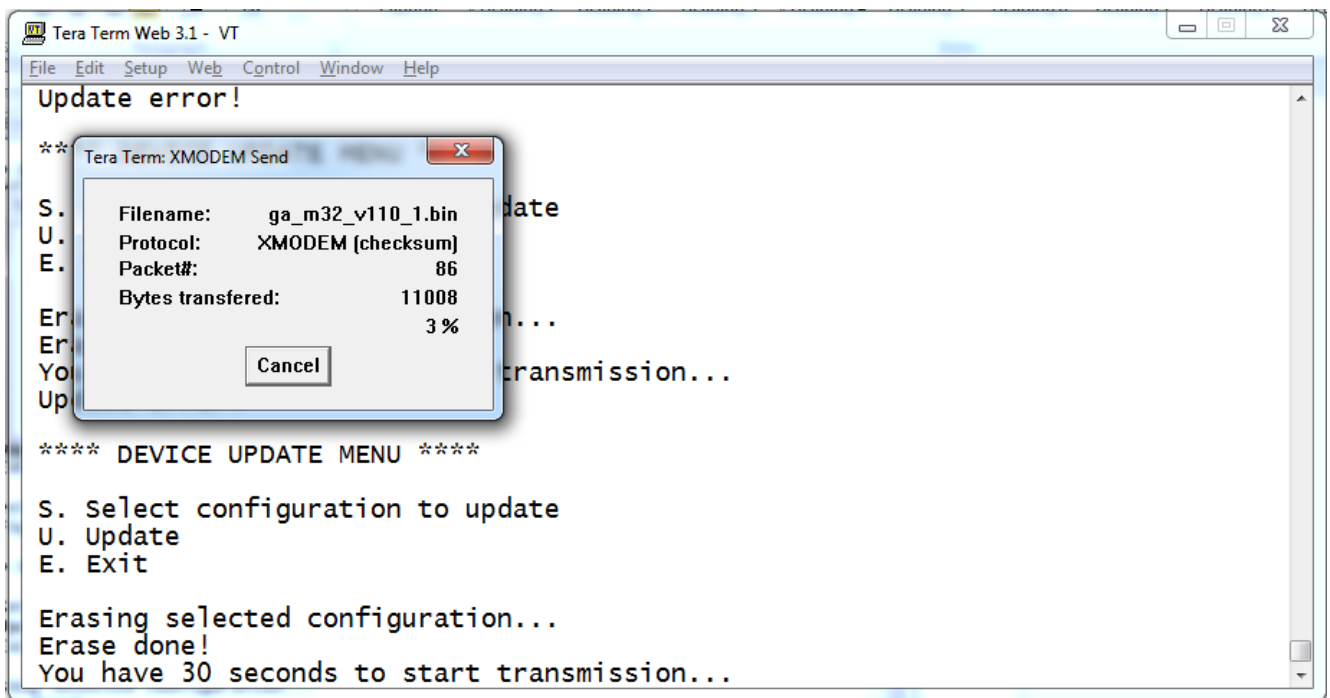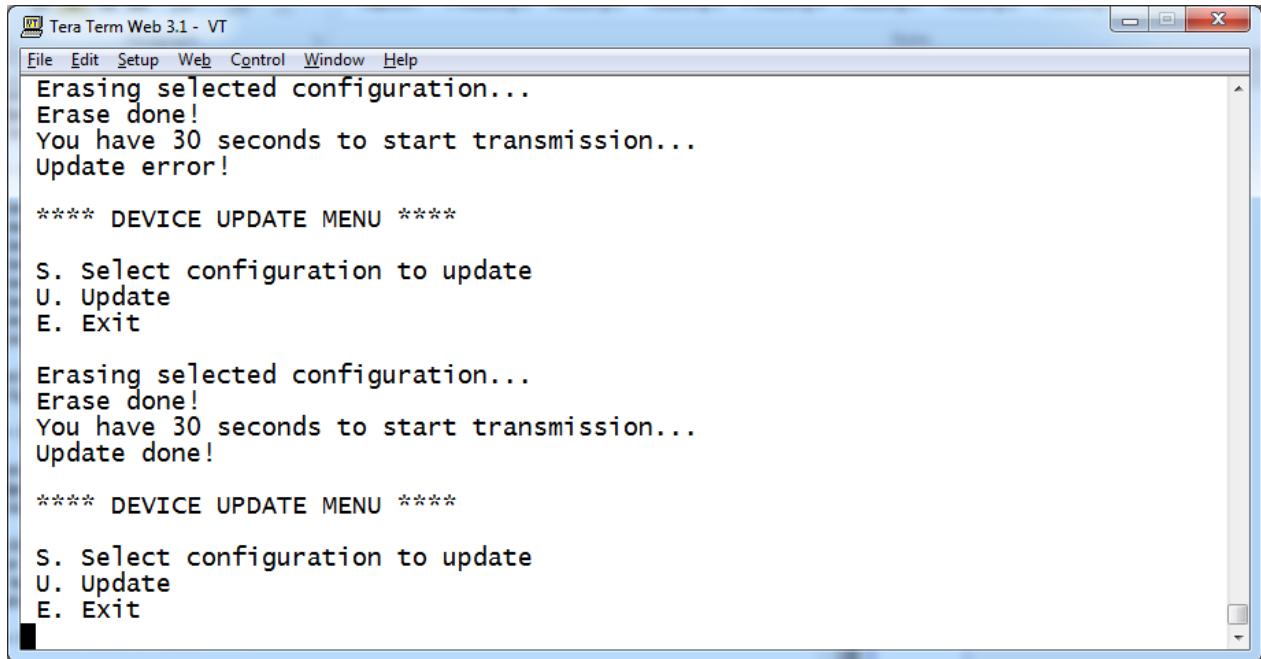
Figure 12.10 The FPGA Update in Progress

If the update was successful, the following log text will appear:

Figure 12.11 The FPGA Configuration Update Successful

- 75 -

13 UPDATING LED PANEL PIXEL MAPPING MATRIX

> ℹ️ For more details and a short instructional video, please check the PAPP005 application note – Pixblasters Quick Start Guide – Driving LED Panels: https://pixblasters.com/wp-content/uploads/2024/07/papp005-Pixblasters-Panels-Quick-Start-Guide_v1.0.pdf

Figure 13.1 FPGA Update Guide – Instructional Video
(https://www.youtube.com/watch?v=uhBz0K_IGW8)

LED panels come with different wiring and in order to be able to integrate many LED panels in a single high-resolution video LED display, the Pixblasters MS1 LED controller must be aware of the exact panels' wiring. Paragraph **6 BUILDING LED PANELS DISPLAY USING PIXBLASTERS CONTROLLER** provides more information about the supported LED panels.

> ℹ️ Pixblasters MS1 controller supports much more LED panel types. If you have a custom LED panel type, or encounter the LED panel that is currently unlisted, please contact us directly at info@pixblasters.com. We will prepare the new pixel mapping matrix and include it in a list of supported LED panel types.

14 ARRAY-SL LED HUB BOARD

14.1 Array-SL Board Specifications

Figure 14.1 The Array-S Auxiliary LED HUB Board

The Array-SL board enables releasable connections with the Pixblasters MS1 LED controller and easy building of the LED matrix. It distributes the video control signals and the power sourced from an external +5 VDC power supply. The board includes holes for easy wall mounting. Each controller supports up to two Array-SL Boards for a total of 32 strips per controller.

> ℹ️ The Array-SL board set is helpful, but not necessary for using the Pixblasters MS1 LED controller. **Figure 14.2** shows the display with different wiring terminals used.
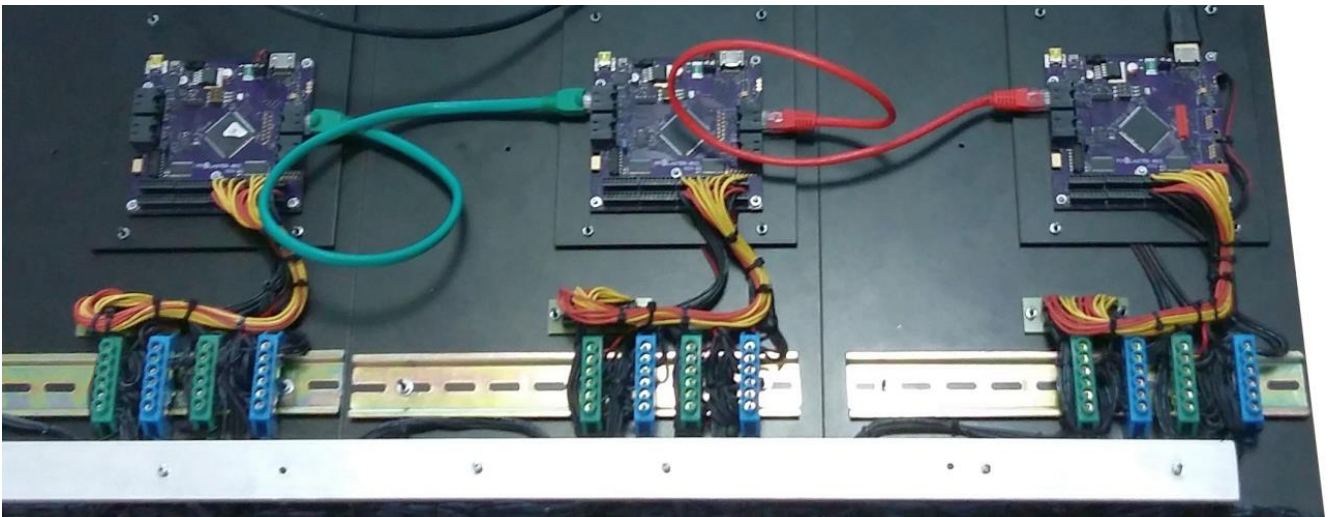
Figure 14.2 Example Wiring (Terminals) in the Multi-Controller Display Topology

LED HUB board is an optional component which has been designed to help users building their custom LED displays. It is a PCB that helps connecting LED strips, power supply, and Pixblasters controller together. Each LED HUB supports up to 16 LED strips.
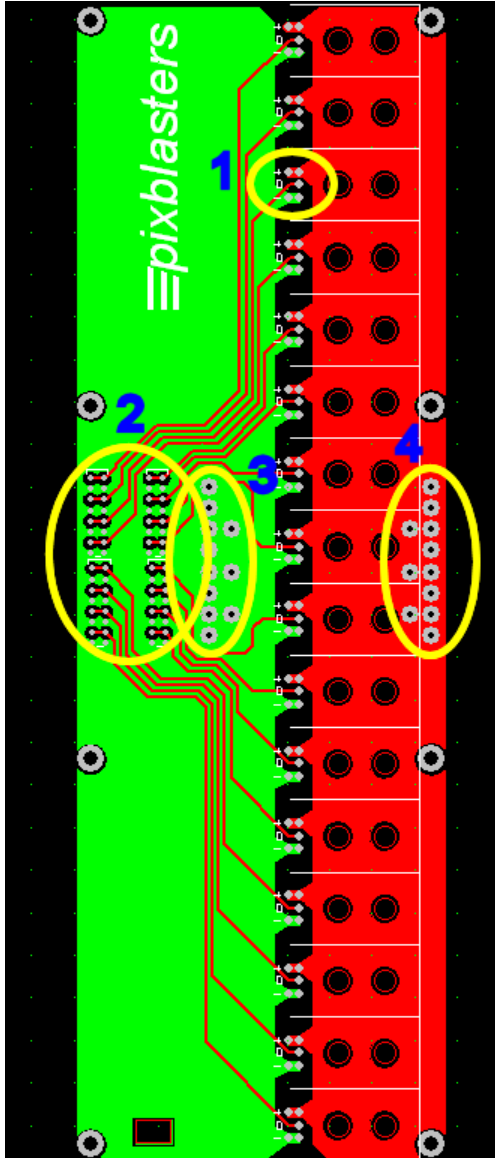
1 – LED strip port
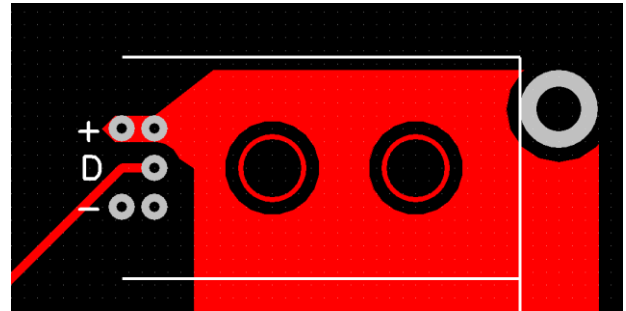
'+' - LED strip power
'D ' – LED strip data
'-' - LED strip ground

2 - Pixblasters controller port

Placeholder for connectors used for connecting to Pixblasters controller LED strip ports. Total of 4 Phoenix-1771017 connectors are supported.

3 – GND port

TH pads for connecting GND wires from power supply module. One pad may be used for connecting GND to Pixblasters controller.

4 – POWER port

The pads for connecting POWER wires from power supply module. One pad may be used for connecting POWER to Pixblasters controller.

Figure 14.3 Array-SL LED HUB Board's Layout
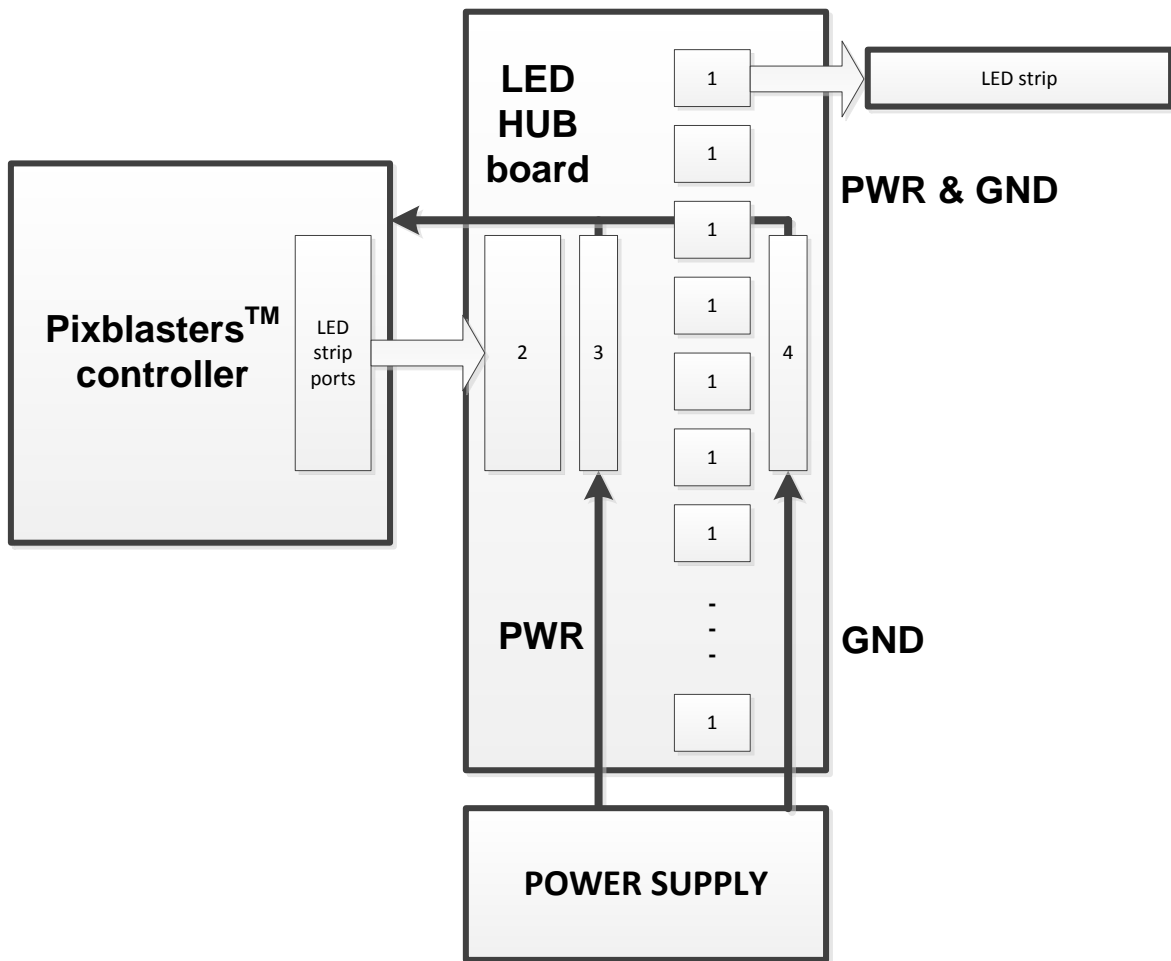
Figure 14.4 LED HUB Usage

> ℹ️ The Array-SL board can be connected to +12 VDC to power compatible LED strips. In that case, please make sure to power the Pixblasters controller from the separated +5 VDC power supply.

15 EXAMPLES – LED STRIPS

> Do not miss our PAPP004 application note: LED Strips Matrix – Non-Segmented and Segmented (Zig-Zag): http://pixblasters.com/wp-content/uploads/2021/03/papp004-Pixblasters-LED-Strips-Matrix-Explained_v1.0.pdf
>
> It brings several illustrative examples of the MS1 LED controller initialization.

15.1 Single Controller Topology, 32 LED Strips, Res - 256x64

This example shows how to configure the Pixblasters MS1 LED controller operating in the "Single controller topology", with 32 3-wire WS2812B LED strips connected in the segmented (zig-zag) arrangement and with a supported resolution of 256 x 64. Since the vertical resolution exceeds the maximum number of LED strips connections, the LED strips need to be segmented. Each LED strip should have 512 LEDs, and each will "handle" two 256 pixels lines (segments).

Control register
Read "Enable" bit first. Example assumes that controller is disabled (no valid video at the input).
0x00A0 – Controller disabled, No mirroring, No padding, GRB pixel order,

CropX
0015 – 15 pixels at the beginning of each line of the original video connected to video input will be skipped. 16th pixel in each line will be the first pixel sent to LED stripes.

CropY
0015 – 15 lines at the beginning of each frame of the original video connected to video input will be skipped. 16th line of each frame will be the first line sent to LED stripes,

StoreX
255 – 256 pixels will be taken from the original video connected to video input (starting at pixel 16 as defined in CropX) and sent to LED strips,

StoreY
0063 – 64 lines will be taken from the original video connected to video input (starting at line 16 as defined in CropY) and sent to LED strips,

StoreHRES
0512 – Each LED strip line will have 512 pixels in total,

StorePRESCAL
0x0000 – $T_{SLOT} = T_{STRIP} = 1/25.6MHz$,

SegmentsNO
001 – Two segments per strip

ProgLUTS

001 – WS2812 bit encoding selected

> ⓘ By default, on-board DIPs prevails the **CfgLoadCntrol** content if its LSB equals 0. We recommend use of DIP switches.

15.2 "Mirrored" Multi-controller Topology, 32 LED strips, Res - 1024x64

This example shows how to configure Pixblasters controllers operating in "Mirrored multi controller topology", with 32 LED strips connected to each controller and with resolution of 1024 x 64 **Figure 15.1**. Mirrored topology is used to double the maximum horizontal resolution. Required vertical resolution also exceeds the maximum number of LED strips that may be connected to a single controller, so two Pixblasters controllers are used per vertical column. Each LED strip should have 512 LEDs, and each will "handle" half of the line. Assumed the use of GS8208 RGB LEDs.
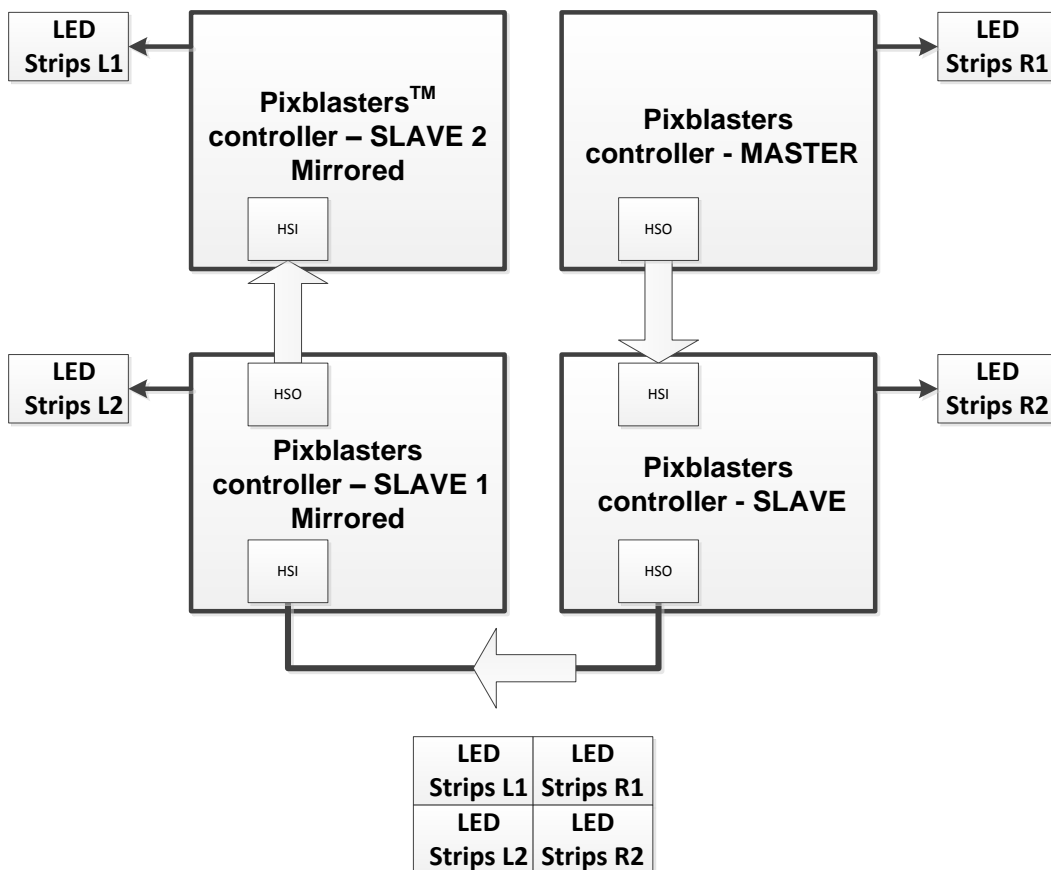
Figure 15.1 Example of Mirrored Multi-Controller Topology

Register content for each of four Pixblasters controllers is given in the remaining of the text.

15.2.1 MASTER Controller – Not Mirrored

Control register

Read "Enable" bit first. Example assumes that controller is disabled (no valid video at the input).

0x00A0 – Controller disabled, No mirroring, No padding, GRB pixel order,

CropX

511 – 512 pixels at the beginning of each line of the original video connected to video input will be skipped. 513rd pixel in each line will be the first pixel sent to LED stripes,

CropY

0000 – 0 lines at the beginning of the frame of the original video connected to video input will be skipped. First line of each frame will be the first line sent to LED stripes,

StoreX

0511 – 512 pixels will be taken from the original video connected to video input (starting at pixel 513 as defined in CropX) and sent to LED strips,

StoreY

0031 – 32 lines will be taken from the original video connected to video input (starting at first line as defined in CropY) and sent to LED strips,

StoreHRES

0512 – Each LED strip line will have 512 pixels,

StorePRESCAL

0 – T_{SLOT} = T_{STRIP} = 1/25.6MHz,

SegmentsNO

000 – No segments

ProgLUTS

002 – GS8208 bit encoding selected

15.2.2 SLAVE Controller – Not Mirrored

Control register

Read "Enable" bit first. Example assumes that controller is disabled (no valid video at the input).

0x00A0 – Controller disabled, No mirroring, No padding, GRB pixel order,

CropX

0511 – 512 pixels at the beginning of each line of the original video connected to video input will be skipped. 513rd pixel in each line will be the first pixel sent to LED stripes,

CropY

0031 – 32 lines at the beginning of the frame of the original video connected to video input will be skipped. First line of each frame will be the first line sent to LED stripes,

StoreX

0511 – 512 pixels will be taken from the original video connected to video input (starting at pixel 513 as defined in CropX) and sent to LED strips,

StoreY

0031 – 32 lines will be taken from the original video connected to video input (starting at line 33 as defined in CropY) and sent to LED strips,

StoreHRES

0512 – Each LED strip line will have 512 pixels,

StorePRESCAL

0 – $T_{SLOT} = T_{STRIP} = 1/25.6MHz$,

SegmentsNO

000 – No segments

ProgLUTS

002 – GS8208 bit encoding selected

15.2.3 SLAVE 1 Controller – Mirrored

Control register

Read "Enable" bit first. Example assumes that controller is disabled (no valid video at the input).
0x00A2 – Controller disabled, Mirrored, No padding, GRB pixel order,

CropX

0000 – 0 pixels at the beginning of each line of the original video connected to video input will be skipped. First pixel in each line will be the first pixel sent to LED strips,

CropY

0031 – 32 lines at the beginning of the frame of the original video connected to video input will be skipped. First line of each frame will be the first line sent to LED strips,

StoreX

0511 – 512 pixels will be taken from the original video connected to video input (starting at first pixel as defined in CropX) and sent to LED strips,

StoreY

0031 – 32 lines will be taken from the original video connected to video input (starting at line 33 as defined in CropY) and sent to LED strips,

StoreHRES

0512 – Each LED strip line will have 512 pixels,

StorePRESCAL

0 – T_{SLOT} = T_{STRIP} = 1/25.6MHz,

SegmentsNO

0 – No segments

ProgLUTS

002 – GS8208 bit encoding selected

15.2.4 SLAVE 2 Controller – Mirrored

Control register

Read "Enable" bit first. Example assumes that controller is disabled (no valid video at the input).

0x00A2 – Controller disabled, Mirrored, No padding, GRB pixel order,

CropX

0000 – 0 pixels at the beginning of each line of the original video connected to video input will be skipped. First pixel in each line will be the first pixel sent to LED stripes,

CropY

0000 – 0 lines at the beginning of the frame of the original video connected to video input will be skipped. First line of each frame will be the first line sent to LED stripes,

StoreX

0511 – 512 pixels will be taken from the original video connected to video input (starting at first pixel as defined in CropX) and sent to LED strips,

StoreY

0031 – 32 lines will be taken from the original video connected to video input (starting at first line as defined in CropY) and sent to LED strips,

StoreHRES

0512 – Each LED strip line will have 512 pixels,

StorePRESCAL

0 – T_{SLOT} = T_{STRIP} = 1/25.6MHz,

SegmentsNO

0 – No segments

ProgLUTS

002 – GS8208 bit encoding selected

- 84 -

16 EXAMPLES – LED PANELS

> *i* Here is a couple of different initialization examples for video LED displays built with RGB LED panels.

16.1 LED panel 8x8 example

8X8 LED panel test = 128x8 resolution		
CONTROL	0x0140	Top left corner of the cropped video input is (400, 400)..
CROPX	400	
CROPY	400	
STOREX	127	If STOREY /= 127, the image will not be correct. This configuration uses two LED outputs for the horizontal LED band that consists of:
STOREY	7	
STOREHRES	512	(8 panels per output * 8 LEDs horizontally) * two outputs = 128 LEDs horizontally.
STOREPRESCAL	0	
PROGLUTS	0x0000	STOREHRES must be always 512 to lit on the complete LED output for LED panels.
SEGMENTSNO	2	
CONTCFGLOADCNTRL	0x0000	
AUX1	0x001F	SEGMENTSNO = 2 defines two LED outputs for the hor. LED band
AUX2	0	
MAT_H	7	MAT_H and MAT_V defines panel's resolution.
MAT_V	7	MAT_CFG defines panel's type.
MAT_CFG	0x0007	

Table 16.1: 8x8 Panel Example Initialization

16.2 LED panel 16x16 example

8X8 LED panel test = 64x32 resolution		
CONTROL	0x0140	Top left corner of the cropped video input is (400, 400).
CROPX	400	
CROPY	400	
STOREX	63	If STOREY /= 63, the image will not be correct. This configuration uses two LED outputs for the horizontal LED band that consists of:
STOREY	31	
STOREHRES	512	(2 panels per output * 16 LEDs horizontally) * two outputs = 64 LEDs horizontally.
STOREPRESCAL	0	
PROGLUTS	0x0000	STOREHRES must be always 512 to lit on the complete LED output for LED panels.
SEGMENTSNO	2	
CONTCFGLOADCNTRL	0x0000	SEGMENTSNO = 2 defines two LED outputs for the hor. LED band
AUX1	0x001F	
AUX2	0	MAT_H and MAT_V defines panel's resolution.
MAT_H	15	MAT_CFG defines panel's type.

MAT_V	15	
MAT_CFG	0x0011	

Table 16.2: 16x16 Panel Example Initialization

16.3 LED panel 32x8 example

8X8 LED panel test = 128x32 resolution		
CONTROL	0x0140	
CROPX	400	Top left corner of the cropped video input is (400, 400).
CROPY	400	
STOREX	127	If STOREY /= 127, the image will not be correct. This configuration uses two LED outputs for the horizontal LED band that consists of:
STOREY	31	
STOREHRES	512	(2 panels per output * 32 LEDs horizontally) * two outputs = 128 LEDs horizontally.
STOREPRESCAL	0	
PROGLUTS	0x0000	STOREHRES must be always 512 to lit on the complete LED output for LED panels.
SEGMENTSNO	2	SEGMENTSNO = 2 defines two LED outputs for the hor. LED band
CONTCFGLOADCNTRL	0x0000	
AUX1	0x001F	MAT_H and MAT_V defines panel's resolution.
AUX2	0	MAT_CFG defines panel's type.
MAT_H	31	
MAT_V	7	
MAT_CFG	0x0021	

Table 16.3: 32x8 Panel Example Initialization

16.4 LED panel 44x11 example

8X8 LED panel test = 88x11 resolution		
CONTROL	0x0140	
CROPX	400	Top left corner of the cropped video input is (400, 400).
CROPY	400	
STOREX	87	If STOREY /= 87, the image will not be correct. This configuration uses two LED outputs for the horizontal LED band that consists of:
STOREY	10	
STOREHRES	512	(1 panels per output * 44 LEDs horizontally) * two outputs = 88 LEDs horizontally.
STOREPRESCAL	0	
PROGLUTS	0x0000	STOREHRES must be always 512 to lit on the complete LED output for LED panels.
SEGMENTSNO	2	SEGMENTSNO = 2 defines two LED outputs for the hor. LED band
CONTCFGLOADCNTRL	0x0000	
AUX1	0x001F	MAT_H and MAT_V defines panel's resolution.
AUX2	0	MAT_CFG defines panel's type.
MAT_H	43	
MAT_V	10	
MAT_CFG	0x0040	

Table 16.4: 44x11 Panel Example Initialization

16.5 LED panel 22x22 example

8X8 LED panel test = 44x22 resolution		
CONTROL	0x0140	Top left corner of the cropped video input is (400, 400).
CROPX	400	
CROPY	400	If STOREY /= 44, the image will not be correct. This configuration uses two LED outputs for the horizontal LED band that consists of:
STOREX	43	
STOREY	21	
STOREHRES	512	(1 panels per output * 22 LEDs horizontally) * two outputs = 44 LEDs horizontally.
STOREPRESCAL	0	
PROGLUTS	0x0000	STOREHRES must be always 512 to lit on the complete LED output for LED panels.
SEGMENTSNO	2	SEGMENTSNO = 2 defines two LED outputs for the hor. LED band
CONTCFGLOADCNTRL	0x0000	
AUX1	0x001F	MAT_H and MAT_V defines panel's resolution.
AUX2	0	MAT_CFG defines panel's type.
MAT_H	21	
MAT_V	21	
MAT_CFG	0x0030	

Table 16.5: 44x11 Panel Example Initialization

17 APPENDIX A – ABOUT LED STRIPS BASED DISPLAYS

Pixblasters LED controller enables creation of LED display installations that are:

1. Flexible enough to support practically any method of generating digital content, including the use of professional digital signage software
2. Easy to use and with the hardware architecture fully transparent to end users
3. Creatively different from other common products on the market
4. Large, or possibly enormous in size, and
5. Affordable to build and operate

The Pixblasters MS1 video LED controller's ability to connect to any computer and any operating system as an ordinary monitor successfully fulfills the first two requirements from the list. The last three project requirements are linked to a common question regarding Pixblasters - **why do we support LED strips and not the fixed LED modules?**

> Pixblasters now supports LED panels with addressable RGB LEDs. WS2812-like LEDs offer individual control of each LED, whereas fixed LED modules are designed for uniform, large-scale displays and need external electronics to control pixel dimming.

The Pixblasters team has recognized specific situations and applications where LED strips-based video screens can offer certain benefits and fill the holes in the existing LED display feature sets.

17.1 LED Strips are More Affordable and Manageable

The high-end module based LED displays are huge "electronic monsters" that implement hundreds and thousands of driving chips, carefully calibrated LED lots, air-conditioners to keep the whole system thermally stable, complex internal networking based on proprietary communication standards, etc. Due to the complexity that comes at high costs, those huge, shiny, beautiful video displays have always been beyond the available price range and beyond the area of technical expertise of the vast majority of enthusiasts and small business owners.

The Pixblasters MS1 video LED controller wants to change it and to make big video LED displays affordable to everyone. In combination with LED strips, this video controller assures the most important LED display features like the smooth video reproduction, appealing color content and big displays size.

With the LED strip pricing in a range of $2 - 5 per (source: Alibaba.com), LEDs for the 8 meters long and 0.3 meters high (480 x 16 resolution) would cost between $256 – 640.
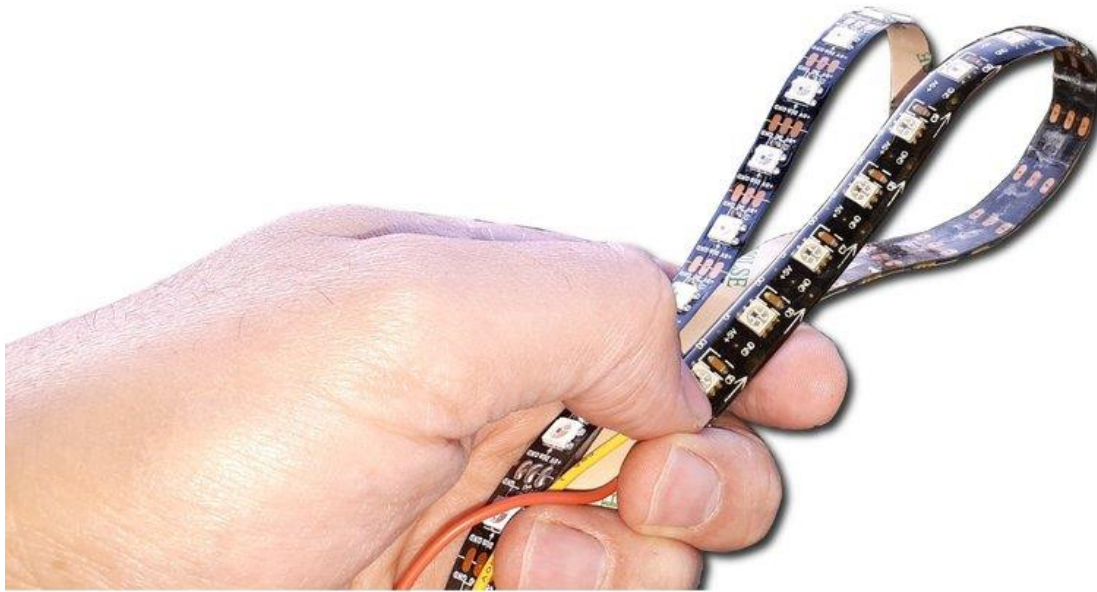
Figure 17.1 LED Strip (left) with Low Water Resistance and the Waterproof LED Strip (right)

17.2 LED Strips are Flexible, Lighter, and Fits to a Variety of Surfaces

Responsive and LED strips based video displays offer some features that cannot be attained by standard LED modules. The LED strips can be curved in different shapes. Wrap them around the round pillars, glue them as a transparent curtain in the shopping window glass, embed the animated strips in big printed jumbo posters, cover the room's ceiling or even the complete street vault by video LED strips display, and new video installations that are hard or impossible to build with LED modules become reality!

Lightweight LED strips do not need complex mechanical supports. They can be easily glued or stapled to different surfaces, inserted in LED pipes, and combined in various lattice structures, etc. Literally within minutes, LED strips can be glued to a plain smooth wall to convert it into a video screen, which can be wirelessly controlled through the Internet.

The strips come with different numbers of LEDs per meter/foot, with LEDs of different size, in different PCB colors, and with different waterproof ratings. The **Figure 17.1** shows both types, the RGB LED strip with a very low water resistance and the waterproof LED strip that is fully covered by the silicon coating and suitable for outdoors applications.

17.3 LED Strips are More Durable, and Easier to Repair if They do Break

It sounds a bit funny, but the **LED strips based video screens can be cut and repaired by ordinary scissors.** The non-functional part of the LED screen can be easily cut off and replaced with new pieces of the LED strip! Much easier than in more complex LED module based video displays that often have scrambled screen parts while waiting for repair.

17.4 LED Strips Can be Adjusted for Pixel Pitch and Density

Chained Pixblasters Video LED Controllers can drive LED displays with HD (High Definition) video resolutions, but we strongly believe that great and engaging video results can be also achieved by LED displays, which have smaller video resolution and are great in size. It is somehow related to the way humans see – our eyes and brain connect a smaller number of visible dots (pixels) in a smooth video image.

A bigger pixel pitch (distance between two pixels either horizontally or vertically) and the lower pixel density significantly decrease assembly and operational costs of such displays. With the LED strips, the pixel pitch and pixels density can be freely changed, which cannot be done by predefined and fixed pitch LED modules. A school gym scoreboard visible from a big distance may have a significantly bigger pixel pitch than a room's ceiling decoration that should be visible from a closer distance. Please note that the Pixblasters MS1 controller drives serially connected addressable RGB LEDs, which can be supplied as the off-the-shelf LED strips with a horizontally fixed pitch, or self-made LED strips (single pixels wired) with an arbitrary setup pixel pitch.

As a conclusion, a modules based video LED display that i.e. shows news with letters of human height size, requires much more LEDs than the LED strips based video display. Of course, the first one usually provides a higher quality image, but the strips based one generates a comparable video effect at a much lower cost!

A single Pixblasters controller directly drives 16,384 LEDs, which in displays built up from the 30 LEDs/m (pitch 33 mm), turns into an active video surface of approx. 17 square meters (~ 183 ft2). Multiple Pixblasters controllers can be chained to build economically viable video installations that can span entire buildings. Potentially it can turn the complete shopping mall's façade into responsive and remotely controlled video display that shows any multimedia content.

On the other side of the display size-spectrum are ordinary illuminated signs on shops, coffee bars, gas stations… that can be easily converted into video displays connected to the Internet. Such conversion only requires the exchange of the existing background LEDs by the addressable RGB LEDs driven by the Pixblasters and a small computer like the Raspberry Pi Zero, or similar.

A number of used LEDs is not the only significant cost driver. The LED modules require a higher number of driver chips and each LED module contains such driving electronics.

17.5 The Pixblasters MicroSign Demo

The LED strips based displays have significantly simpler construction and all electronic functions are contained in the Pixblasters MS1 video LED controller. It can be easily illustrated by our Pixblasters MicroSign (**Figure 17.2**) display, which was available only during the crowdfunding campaign for demonstration purposes.

The photo shows all necessary display parts. The Raspberry Pi computer generates the video output connected to the Pixblasters MS1 video LED controller. The demo uses only a half of available 32 digital outputs and drives 120 x 16 (1,920) LEDs on 16 LED strips attached through the Pixblasters Array-S terminal board. This board can be easily exchanged by any kind of wire terminals. There is also a power supply for the complete video display.
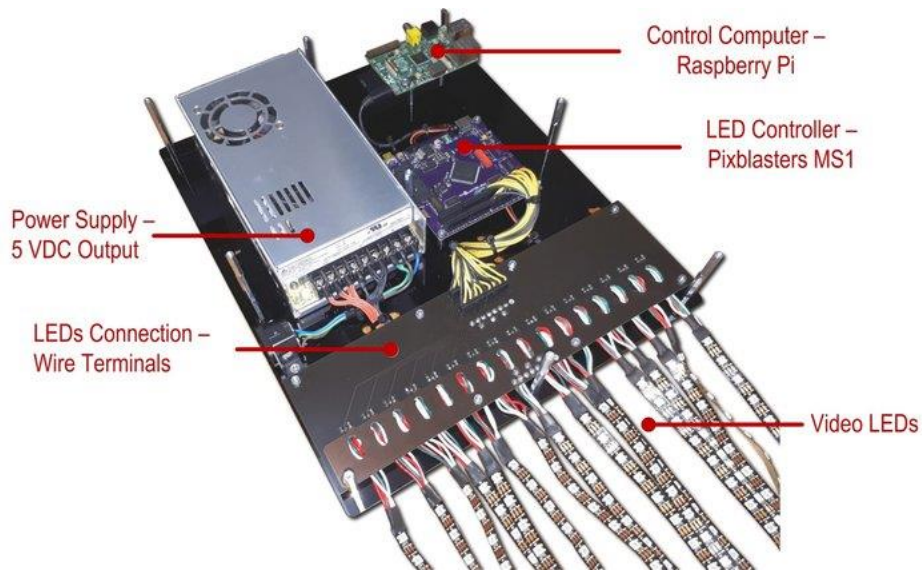
Figure 17.2 The Pixblasters MicroSign Demo (resolution 120x16, display size 2 x 0.3 m)

(not available as a product from Pixblasters!)

Display like this one can be built in one day! It was named the MicroSign due to a number of connected LEDs. But, the LED driving capabilities of this simple electronic assembly are much, much higher!
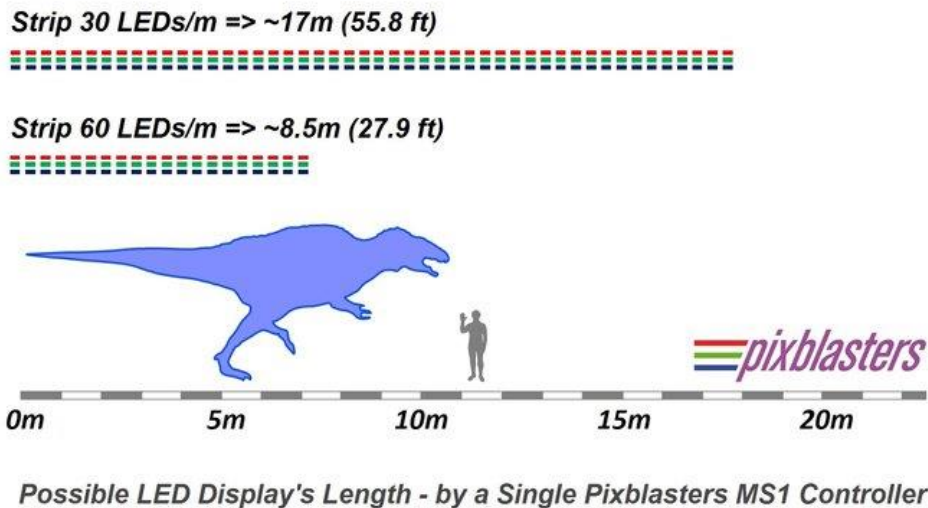

Figure 17.3 Achievable LED Display Lengths by a Single Pixblasters MS1 Video Controller

Figure 17.3 illustrates the length of LED displays that can be controlled by the MicroSign, and that length is simply impressive. Additionally, the unused digital outputs can effectively double the vertical display's resolution and the number of LEDs. Of course, the bigger number of LEDs requires more power and the power supply unit from the **Figure 17.2** would not be sufficient for more than 2,000 LEDs. The MicroSign LED displays with higher resolutions would require additional power supply units, but the LED control electronic remains the same.

The following video clip shows the Pixblasters MicroDemo in action. We think that this simple display construction, with the activated Internet access and the digital signage software, can transform i.e. dull street shop kiosks into multimedia hot spots. Glue the waterproof LED strips around the kiosk's top and it will light up!

Please note that the MicroSign display demo use the Pixblasters open-source FPGA. The code is available from here: https://github.com/PixiGreen/Pixblasters-MicroDemo

Figure 17.4 Pixblasters MicroSign - Open Source LED Display Linux Test Controller

(*https://www.youtube.com/watch?v=CLNaBz24IXY*)

18 APPENDIX B - µC AND FPGA DIRECT PROGRAMMING

🚫 Programming steps described in this paragraph may **PERMANENTLY DISABLE** the Pixblasters functionality and should be executed only by skilled developers who want to use the MS1 board hardware in ways that differ from its original purpose.

🚫 Pixblasters **do not provide PIC18F26J50 firmware** for download to the end users. Once deleted, it becomes **PERMANENTLY LOST** to the end user! The same applies to the FPGA configuration 0. If the EEPROM containing FPGA configurations is completely erased, user will lose ability to update remaining FPGA configurations!

Revision History

Version	Date	Note
1.00	11.03.2020.	Initial release
1.01	09.02.2021.	Document updated to describe new features (Addition of LVDS Buffer/Repeater with Pre-Emphasis, change of default values and removal of 16 stripe configurations).
1.02	24.02.2021.	CROPX and CROPY registers extended to 11 bits. Default register values updated.
1.20	14.07.2021.	Added support for 4-wire APA102 LEDs. Described control interface changes. Added references to educational video clips and application notes. (Unpublished)
1.30	16.09.2021.	Published version with the added support for 4-wire APA102 LEDs. Described control interface changes. Added references to educational video clips and application notes. Valid for FPGA ver 107 and SW ver 104.
1.40	25.01.2022.	Changed Aux1 register description for 3-WIRE RGB LEDs
2.00	25.04.2024.	Added information about support for LED panels.
2.10	08.07.2024.	The first release with the integrated support for LED panels.