

Guide to RS-232 commands on Digital View LCD Controller Boards

July 2023
www.digitalview.com

Disclaimer: This guide is provided 'as is' without warranty of any kind, either express or implied. Your use of this guide is at your own risk. Digital View is not responsible for any damage or loss caused by the use or misuse of this guide or the RS-232 commands it pertains to.

While every effort has been made to ensure the accuracy of this guide, errors can occur. Always refer to the latest version of the guide available on our website at www.digitalview.com (see LCD controller product pages).

Digital View LCD controller boards feature an RS-232 interface, enabling a range of commands to control and customize your LCD display. This guide will help you understand and utilize this command set, unlocking the full potential of your device.

1. Introduction

- a. Purpose of the guide
- b. A brief overview of the Digital View LCD Controller board
- c. Explanation of what RS-232 commands are and their importance

2. Getting Started

- a. Prerequisites (hardware and software requirements)
- b. Safety guidelines and best practices when using the Digital View LCD controller board
- c. Overview of the RS-232 interface on the controller board, along with an explanation of the role and function of each pin

3. Setting Up Your System

- a. Step-by-step instructions on how to connect the controller board to a PC
- b. Instructions on setting up any necessary software (drivers, terminal programs, etc.)
- c. How to verify a successful connection

4. RS-232 Commands

- a. Overview of basic commands
- b. A detailed explanation of each command, including what it does and when to use it
- c. Step-by-step instructions on how to send each command, including any necessary code or command line input
- d. Troubleshooting, common issues, and possible solutions

5. Commands via the Ethernet Port

- a. Overview of ethernet communication
- b. Setting up the Ethernet communication
- c. Sending commands over Ethernet
- d. Troubleshooting Ethernet commands

6. Example Applications

- a. Sample applications that demonstrate the use of various RS-232 commands
- b. Detailed explanations of how these applications work, and step-by-step instructions on how to run them

7. FAQs and Additional Resources

- a. Frequently asked questions about using RS-232 commands with the Digital View LCD controller board
- b. Links to additional resources, such as in-depth tutorials, forums, and the manufacturer's documentation

8. Contact Information

- a. How to contact support for further assistance

1. Introduction

a. Purpose of the guide

This guide aims to help users effectively utilize the RS-232 command set available on the Digital View LCD controller board. It's designed to assist in controlling and customizing LCD display settings to fit individual requirements.

b. Overview of the Digital View LCD controller board

The Digital View LCD controller board is a robust and reliable device capable of driving a broad range of LCD panels. The board's key feature is the incorporation of an RS-232 interface, a standard for serial communication transmission of data. This interface provides users with the flexibility to manage display functions, creating a customizable user experience.

c. Understanding RS-232 commands

RS-232 commands are a set of coded instructions transmitted through the RS-232 interface. These commands allow users to interact with and control various functionalities of the LCD controller board. Functions such as turning the display on or off, adjusting brightness, or changing input sources can be controlled via these commands.

2. Getting Started

a. Prerequisites

Before starting, ensure you have the following:

- A Digital View LCD controller board.
- An RS-232 cable to connect the board to your computer.
- A computer with a serial port or a USB to serial adapter.
- Terminal software (such as PuTTY or Tera Term) to send commands through the serial port.

b. Safety Guidelines and Best Practices

For safe and effective use of the Digital View LCD controller board, adhere to the following guidelines:

- Always power off the board before making or changing connections.
- Avoid touching the board's components directly to prevent static discharge.
- Always use the appropriate cables and connectors. Improper connections can cause damage.

c. Overview of the RS-232 Interface

The RS-232 interface on Digital Viuew LCD controller boards is typically CN8, a 6-pin JST connector; see the relevant controller board manual for details. This interface allows for serial communication between your computer and the board. Each pin has a specific function:

- RXD: Data input to the board.
- TXD: Data output from the board.
- VCC: +5V DC
- GND: Ground.

Note: Other pins' functionality may vary depending on the model of the controller board. Please refer to the board's specific datasheet for detailed pin-out information.

Once everything is properly connected and the terminal software is set up, you can verify the connection by sending a simple command, such as a command to get the firmware version. If the board responds correctly, the connection is successful.

3. Setting Up Your System

a. Step-by-Step Instructions on How to Connect the Controller Board to a PC

Follow these steps to connect your Digital View LCD controller board to your computer:

- Connect one end of the RS-232 cable to the RS-232 port on the Digital View LCD controller board.
- Connect the other end of the RS-232 cable to the serial port on your computer. If your computer does not have a serial port, use a USB to serial-adaptor.
- Power on your Digital View LCD controller board.

b. Instructions on Setting Up Necessary Software

After physically connecting the controller board to your PC, you will need to set up your terminal software:

- Download and install a terminal software program. Programs such as PuTTY or Tera Term are commonly used for RS-232 communication.
- Launch the terminal software and configure the settings to match the RS-232 specifications of the controller board. This typically involves setting the correct COM port, a baud rate of 9600, 8 data bits, no parity, and 1 stop bit.

c. How to Verify a Successful Connection

Once you've set up the hardware and software, you can check the connection:

- Send a simple command to the controller board via the terminal software. This could be a command to retrieve the board's firmware version or status.
- If the board responds with the expected output, the connection has been set up correctly. If not, double-check your connections and software settings, and try again.

4. RS-232 Commands

In this section, we'll introduce you to the basic RS-232 commands that can be used with the Digital View LCD controller boards. Please note that the exact command set can vary between different models of controller boards.

a. Overview of Commands

RS-232 commands typically allow you to control the functions of your LCD display system. This includes turning the display on or off, adjusting brightness, changing input sources, and retrieving status information from the controller board.

b. Detailed Explanation of Each Command

For specific details about each command, please refer to the user guide for your specific model of the Digital View LCD controller board. These user guides provide a comprehensive list of available commands, along with an explanation of what each command does and the appropriate context for its use. You'll also find information about any parameters that need to be included with each command and the expected responses from the controller board.

c. Instructions on Sending Each Command

Sending an RS-232 command typically involves the following steps:

- Open your terminal software and establish a connection with the controller board.
- Enter the command in the input area of your terminal software. Make sure to include any necessary parameters for the command.
- Send the command. The controller board should respond with an acknowledgment or the requested data.

Please note that the process for sending commands can vary depending on the terminal software you are using. Check the documentation for your software for specific instructions.

To access the user guide for your specific model and get a detailed list of available RS-232 commands, visit the LCD Controllers page on the Digital View website:
<https://www.digitalview.com/controllers/lcd-controllers-home.html>.

5. Commands via the Ethernet Port

Certain models of the Digital View LCD controller boards are equipped with Ethernet ports, which allow users to send RS-232 commands over an Ethernet connection. This opens up a host of possibilities, including long-distance control and integration into existing network infrastructures.

a. Overview of Ethernet Communication

Ethernet communication provides a reliable, high-speed method for transmitting RS-232 commands. By sending commands over an Ethernet connection, users can interact with the LCD controller board remotely or from a device that doesn't have a dedicated serial port.

b. Setting Up the Ethernet Connection

Setting up an Ethernet connection involves physically connecting the LCD controller board to your network, as well as configuring your system to send RS-232 commands over this connection. Specific steps will depend on your network configuration and the model of your LCD controller board.

c. Sending Commands Over Ethernet

Once the Ethernet connection is set up, you can send RS-232 commands over it using dedicated software or a networked terminal. The process is similar to sending commands over a

direct serial connection, but commands are sent to the controller board's network address rather than its COM port.

d. Troubleshooting Ethernet Commands

As with any networked communication, sending RS-232 commands over Ethernet can present its own challenges. This section would provide advice on diagnosing and resolving common issues, such as network connectivity problems or misconfigured settings.

Remember, not all Digital View LCD controller boards support Ethernet communication. Always refer to the user guide for your specific model for detailed instructions and capabilities. The user guide can be found on the relevant product page on the Digital View website.

6. Example Applications

This section will present some practical examples of using RS-232 commands to interact with your Digital View LCD controller board. These examples are designed to help you understand how you might use these commands in real-world situations.

a. Turning the backlight on/off, Windows PC

This example, provided in Appendix 1, shows a simple Windows application written in Python to toggle the backlight on/off.

b. Switching Display Inputs, PIC microcontroller

Many Digital View LCD controller boards support multiple input sources. Using a custom accessory board with a PIC microcontroller can enable function controls to match the needs of a display integration. For example using RS-232 commands to switch between different inputs, allowing you to control what content is displayed.

c. Remote Control via Ethernet

For LCD controller boards with an Ethernet port, this example will demonstrate how you can control your display from a remote location. By sending commands over your network, you can manage the display without needing to be physically present.

d. Retrieving Board Status Information

This example will show you how to use RS-232 commands to retrieve status information from the LCD controller board. This can be useful for diagnostic purposes or for monitoring the status of the board.

Remember, the availability and functionality of these commands can vary between different models of the Digital View LCD controller boards. Always refer to the user guide for your specific model for the exact commands and parameters to use. The user guide can be found on the relevant product page on the Digital View website.

7. FAQs and Additional Resources

This section covers frequently asked questions about using RS-232 commands with the Digital View LCD controller boards, as well as additional resources for further learning and troubleshooting.

a. Frequently Asked Questions (FAQs)

This subsection addresses common queries users might have when working with the LCD controller board and the RS-232 command set, such as:

- How can I check if my RS-232 command was successful?
- Can I send multiple commands in sequence?
- How do I reset my controller board to its default settings?

b. Additional Resources

In addition to this guide, there are numerous resources available to assist you in understanding and using the RS-232 commands:

- Digital View's Official Website: Offers product datasheets, user guides, and other useful information. <https://www.digitalview.com/index.html>
- Online Forums and Communities: Places like StackOverflow have numerous discussions that can provide insights and assistance.
- Technical Blogs and Articles: Various online articles cover RS-232 and its applications in detail.

8. Contact Information

If you encounter any issues not covered in this guide or need further assistance, you can contact Digital View's support:

Inquiry form on the website: <https://www.digitalview.com/support/support.html>

Phone:

- USA: +1.408.782 7773
- UK: +44.(0)20.7631.2150
- HK: +852.2861.3615

Hours - local timezones:

- UK and USA, 8:00am to 4:00pm
- HK, 9:30am to 6:00pm

Our support team is available to assist with any technical difficulties you may have, ensuring you get the most out of your Digital View LCD controller board.

Appendix 1 - Demo App, Backlight On/Off

The following is the Python code for a Windows based application to power on/off the LCD panel backlight via a Digital View SVX-4096 model LCD controller board.

Digital View SVX-4096 RS-232 notes:

- Serial command for turning the LCD panel on/off:
 - 0xe1, '0' to turn off the backlight and 0xe1, '1' to turn on the backlight
- RS-232 baud rate is 9600.
- The controller board automatically returns an acknowledgment after receiving a command.
- The acknowledgment response is the same as the command that was sent.

Notes about the following code:

- It has a simple GUI using tkinter which is typically included with Python
- The user can input the COM port
- The application will check that the COM port is valid
- Turning the backlight on/off is a toggle function

Code...

```
import tkinter as tk
from tkinter import messagebox
import serial
import time

# GUI Application
class Application(tk.Frame):
    def __init__(self, master=None):
        super().__init__(master)
        self.master = master
        self.pack()

        # Variables
        self.serial_port = tk.StringVar()
        self.serial_port.set("")
        self.backlight_state = False # False for Off, True for On
        self.keep_open = tk.BooleanVar()
        self.keep_open.set(False)
```

```
self.ser = None

# Widgets
self.create_widgets()

# Status box
self.status_text = tk.Text(self, height=5, state='disabled')
self.status_text.pack(side="top", fill='x')

# COM port validation
self.check_com_port()

def create_widgets(self):
    # Entry for COM port
    self.com_label = tk.Label(self, text='COM Port:')
    self.com_label.pack(side="top")

    self.com_entry = tk.Entry(self, textvariable=self.serial_port)
    self.com_entry.pack(side="top")

    # Checkbox for keeping port open
    self.keep_open_checkbox = tk.Checkbutton(self, text="Keep port open while application
running", variable=self.keep_open)
    self.keep_open_checkbox.pack(side="top")

    # Button for On/Off
    self.toggle_button = tk.Button(self, text="OFF", command=self.toggle_backlight)
    self.toggle_button.pack(side="top")

    # Check for changes in the COM port entry
    self.serial_port.trace('w', lambda *args: self.check_com_port())

def check_com_port(self):
    if self.ser and self.ser.is_open and not self.keep_open.get():
        self.ser.close()

    try:
        self.ser = serial.Serial(self.serial_port.get(), 9600, timeout=1)
        self.update_status('COM port is valid')
    except serial.SerialException:
        self.update_status('Invalid COM port')
```

```
self.ser = None

def toggle_backlight(self):
    if self.ser:
        if self.backlight_state:
            # If backlight is currently On, turn it Off
            self.send_command('0')
            if self.backlight_state == False:
                # Update button text only if the command succeeded
                self.toggle_button.config(text="OFF")
        else:
            # If backlight is currently Off, turn it On
            self.send_command('1')
            if self.backlight_state == True:
                # Update button text only if the command succeeded
                self.toggle_button.config(text="ON")
    else:
        self.update_status('Cannot send command, COM port is invalid')

def send_command(self, command):
    # Try/except block to handle errors
    try:
        # Turn on/off command
        self.ser.write(bytes([0xe1, ord(command)]))
        time.sleep(1)

        # Read response
        response = self.ser.read(1)

        # Error handling
        if response != bytes([0xe1]):
            self.update_status('Command failed')
        else:
            # Update backlight_state based on command
            if command == '0':
                self.backlight_state = False
            elif command == '1':
                self.backlight_state = True

        self.update_status('Command succeeded')
```

```
except Exception as e:  
    self.update_status('Error: ' + str(e))
```

```
def update_status(self, message):  
    self.status_text.configure(state='normal')  
    self.status_text.insert('end', message + '\n')  
    self.status_text.see('end')  
    self.status_text.configure(state='disabled')
```

```
root = tk.Tk()  
app = Application(master=root)  
app.mainloop()
```

```
*****
```