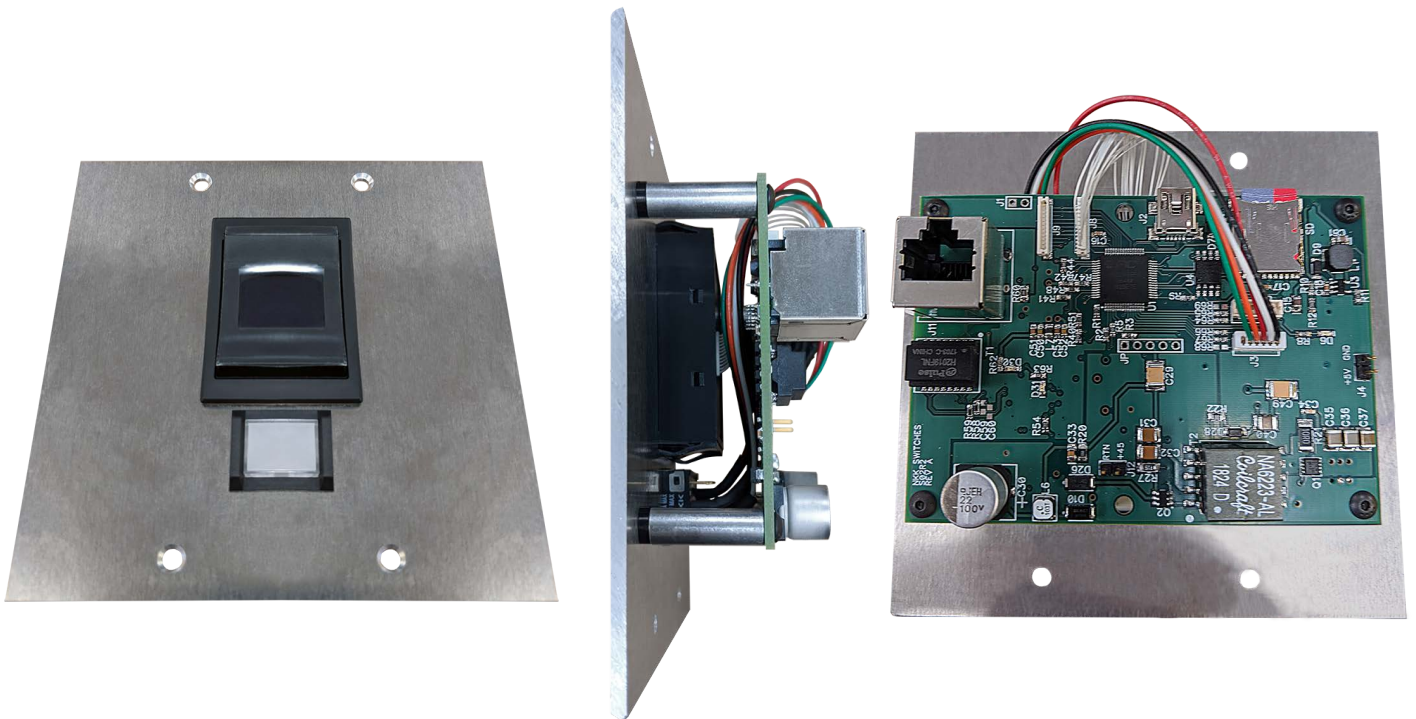# IS-70053 User Manual
Revision C

Power-Over-Ethernet OLED Rocker 2 Gang System

All Rights Reserved Worldwide

Table of Contents

1. OLED Rocker SmartDisplay Switch

The OLED Rocker SmartDisplay switch is a monochrome 96x64 OLED display mounted in the center of a 3-position rocker switch.

Please contact engineering@nkkswitches.com with your requirements for custom solutions.

NKK can supply subsystems with any configuration and number of SmartDisplays with any communication. SmartDisplay allows designers to dynamically change switch legends and images based on desired application functions. The system is ready to interface with a customer's application through USB and Ethernet. It can receive commands, send information, and update the SmartDisplay images.

SmartDisplay is ideal for use in applications with multiple, complex functions which would ordinarily require many dedicated switches and complex training. The dynamic nature of the system allows for instantaneous transitions from generalized lists of categories down to function specific actions. This reduces the need for complicated controls and shortens the time for training by only displaying relevant options and commands.

To help with development, NKK Switches provides free software, Engineering Kits Communicator, to save and erase images on the controller. Also, NKK Switches provides all the documentation necessary to get up and running quickly on our website: https://www.nkkswitches.com/SmartDisplay-resources/

2. General Features

The system is an ethernet-controlled display-on-pushbutton system in a compact form factor. Powered over ethernet, it can report switch presses from remote locations to a host. It comes with the following features:

Features:
- 1 OLED Rocker SmartDisplay and 1 UB pushbutton
- Fits into a single-gang electrical box
- USB or Ethernet controlled.
- Power over ethernet – 60 VDC max
- Images are loaded via Micro-SD
- 15 levels of brightness.
- Real-time control by host.
    - Show any saved image on any switch.
    - Reports switch activity to host.
    - Ability to send images directly to switches without saving to memory.

- o Write text on switches
- Controller board firmware can be customized based on customer requirements.
- Firmware field upgradable via USB.
- Windows based software is available for communication.
- The communication protocol can be modified to meet customer's requirements.
- Please contact the factory about custom builds and firmware modifications.

3. Electrical Specifications

Power Specs: Max 3 Watts.
                    USB +5 VDC
                    PoE +60 VDC max0

www.nkkswitches.com • engineering@nkkswitches.com   0922

4. Communication

The systems can communicate over USB and Ethernet. All commands and responses are detailed in the associated Command List. A non-inclusive list of commands is as follows:

- Acknowledge.
- Get/Set Aux light
- Get/Set Brightness
- Reset system
- Query version
- Set an image on the SmartDisplay
- Play a script (series of images)

The system shows up as a generic USB COM port. This allows quick testing, loading of images, and integration with customer software. For evaluation, the NKK provides test software

5.Images

Images can be created in any graphics software such as Paint, Photoshop, etc, or even user-created software. Images are to be saved on the Micro-SD card in the "images" directory.
To use this software, images must be saved in the proper format:

OLED 96x64	Monochrome bitmap (.bmp) 96x64 pixels	768 bytes per image

6. Operational Overview

Usage of the IS-70053 depends on a basic understanding of ethernet terms and concepts. The Micro-SD card must be configured before powerup. Using the example provided separately, the user can configure all of the ethernet settings, as well as the startup behavior. Images should be loaded in the "images" directory, and scripts should be loaded in the "scripts" directory. A script is a series of images, each displayed for a specific amount of time. Animations and videos can be played using this method.

The system takes no action after startup except loading the initial image or script (set in the config.txt file in the root directory). It then waits for commands from a host system. If a host IP address is set in the config file, the system will attempt to contact it at the (set port + 1). The system then waits for commands from the host. If a switch is pressed, the system reports it to the host in a format specified by the examples.

The format of commands and responses is in human-readable REST format. An example of the config.txt file is as follows:

```
{
        "config": {
                "name": "TEST",
                "device-ip-mode": "DHCP",
                "device-ip-type": "TCP",
                "device-ip-address": "C0A801D2",
                "device-port": "0050",
                "device-subnet": "FFFFFF00",
                "device-gateway": "C0A86401",
                "device-dns": "A2A86401",
                "host-ip-address": "C0A801CE",
                "multi-press-delay": "04FB",
                "startup-brightness": "09",
                "startup-image": "001.bmp",
                "startup-script": "start.txt"
        }
}
```

Note that all IP fields MUST be 8 characters of hexadecimal numbers. Missing character will cause undesirable behavior. All numbers are in hexadecimal format. ALL FIELDS MUST REMAIN IN THE DEFAULT ORDER.

These are the definitions of the fields
name – The name of the device

device-ip-mode – valid options are "Static" or "DHCP"
device-ip-type – valid options are "TCP" or "UDP"
device-ip-address – 4 hex bytes of the desired device IP address (Leave default if in static mode)
device-port – 2 hex bytes of the desired device port
device-subnet – 4 hex bytes of the desired device subnet address
device-gateway – 4 hex bytes of the desired device gateway
device-dns - 4 hex bytes of the desired device DNS
host-ip-address – 4 hex bytes of the address to send an acknowledgment packet to upon startup. The port connected to is the device-port + 1. (Change to 00000000 if not used)
multi-press-delay – 2 hex bytes in milliseconds. The system can be configured to accept multiple key presses before sending all of them to the host. This number is how long the system waits before reporting all pressed switches to the host. Set to 0001 if not used.
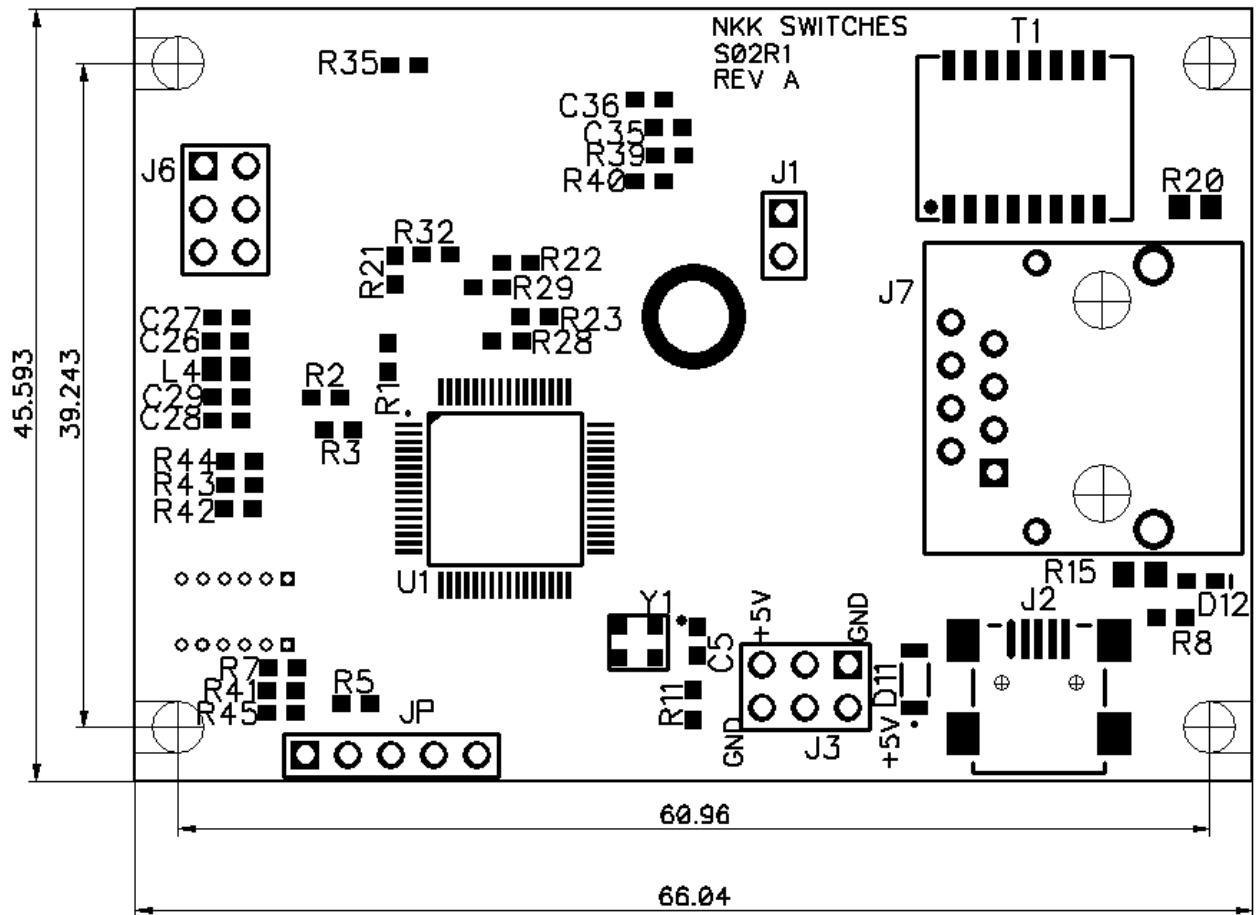**startup-brightness** – Brightness of the switches upon startup. Range from 00-0F
startup-image – The image file name to display upon startup. If a startup script is desired, there should be no value here (e.g. the values after the colon should be "",)
startup-script – The script file name to run upon startup. This is only used if there is no startup image specified

7.Board Dimensions

Typical dimensions.

8. Key Terms & Definitions

Host   Any computer, terminal, or other device that can communicate over the USB line.

Byte   An eight-bit hex value ranging from 00H to FFH (Decimal 0 to 255). The bit format of a byte is: (B7 B6 B5 B4 B3 B2 B1 B0) where B7 is most significant and bit B0 is least significant bit.

Nibble/Hex Digit   A four-bit value ranging from 0H to FH. A byte consists of two nibbles.

ASCII   A byte value representing a symbol.

Communication Format   There are two formats to transmit a byte:

1. **Hex format** - A hex byte is transmitted without any change to it. [**xxH**] will be used to denote this.

   All commands and some data are sent by using this format.

2. **ASCII HEX format** - Each nibble of the byte is converted to ASCII code and sent as a byte. [**xxAH**] will be used to denote this.

   For example, the hex byte 5AH is transmitted in two bytes, **35H** and **41H**. The ASCII value for **5** is **35H** and the ASCII value for **A** is **41H**.

   All addresses and most data are sent using this format.

Address   A two-byte value ranging from 0001H to 03E8H representing the 1000 memory locations for images on the flash memory.

9. Commands to the Controller

Please reference the external documents and demonstration files

---