7850 East Gelding Drive • Scottsdale, AZ 85260-3420

OLED Engineering Kits User Manual

Revision E

Firmware Version 1.X

NKK SWITCHES
7850 E. Gelding Drive
Scottsdale, AZ 85260

Toll Free 1-877-2BUYNKK (877-228-9655)
Phone 480-991-0942
Fax 480-998-1435
e-mail <engineering@nkkswitches.com>

NKK Switches makes no warranty for the use of these products and assumes no responsibility for any errors, which may appear in this document, nor does it make a commitment to update the information contained herein. Smart Switch is trademark of NKK Switches. All Engineering Kits are tested 100% and programmed with the default images and attributes. As the firmware for these kits can be modified by the customer, any damage caused by a customer mistake is not warrantied.

7850 East Gelding Drive • Scottsdale, AZ 85260-3420

Table of Contents

Toll Free 1.877.2BUYNKK (877.228.9655) • **Phone** 480.991.0942 • **Fax** 480.998.1435
www.nkkswitches.com • **Email** engineering@nkkswitches.com

7850 East Gelding Drive • Scottsdale, AZ 85260-3420

1. General Features

The Engineering Kits (IS-ENG-KIT-x) are designed to accelerate the incorporation of the SmartSwitch family of products into real world applications by, not only familiarizing design engineers to the SmartSwitch products, but also as a development platform for experimentation and design viability testing.

The controller used for OLED SmartSwitches is S02L1. The schematic is at the end of this document. The C language source codes and a Windows based communication software, Engineering Kits Communicator, can be downloaded from www.nkkswitches.com. The software is designed to download images, attributes, and manual commands as well as display responses from the kit.

Features of S02L1:

- Controls up to two OLED SmartSwitches or one OLED Rocker.
- 16M on-board NOR flash memory to store images, attributes or any other information.
- Serial communication via USB (115.2K, 1 start bit, 8 bit, 1 stop bit).
- Auxiliary port with 7 I/O pins of microcontroller to control/sense other applications.
- Stand-alone or real-time operation.
- Preprogrammed for basic operation.
- Reports via USB the switch closures and releases, timer expirations, and addresses of the images displayed.
- Programmable for changing the displayed images based on switch actuations and user defined timers.
- Schematics and firmware are provided for user firmware customization.
- User can program the Engineering Kits using a PICKIT3 debugger and pin adaptor.
- The kits come with a 6-foot USB 2.0 A to Mini-B cable (IS-USB1).
- Power Specs: 3.5V to 5.5V.
- Maximum current: 150mA@5VDC.
- Can be powered either by USB or by optional 5V power connection (.100” spacing).
- Window based software is provided for communication.
    - Accepts bitmap files, extracts the images and download them to the controller.
    - Allows typing of commands and downloading to the controller.
    - Extracts HEX or ASCII data from Excel files and downloads to the controller.
    - Messages to and from the controller are displayed in different colors.

7850 East Gelding Drive • Scottsdale, AZ 85260-3420

2. Available OLED SmartSwitches

There are 4 products in OLED family: Standard size switch full color Frameless OLED 96x64, Standard size switch full color OLED 64x48, full color OLED 52x36 display and monochrome OLED 96x64 Rocker. The 3 color OLEDs use the same OLED controller on-board

Part Number	Description	Socket
ISC01P	65k color per pixel 52x36 Display	AT9704-085M
ISC15ANP4	65k color per pixel 64x48 switch	AT9704-085L
ISF15ACP4	65k color per pixel 96x64 frameless switch	AT9704-085L
IS18WWC1W	Monochrome 96x64 Rocker (3 switches)	Panel mount

3. Possible Engineering Kits Using the S02L1 Controller

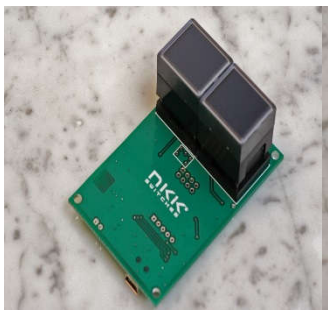
Many Engineering Kits can be made with the S02L1 controller simply by soldering on various sockets (Table below). Only the part numbers in green are marketed and kept in stock. However, all of them can be ordered. All the Engineering Kits come with switches/display on the sockets and a USB cable.

Part Number	Switch 1 / Display 1		Switch 2	
IS-ENG-KIT-7-DS	ISC01P	OLED Display 52X36	ISC15ANP4	OLED Switch 64X48
IS-ENG-KIT-7-FS	ISF15ACP4	Frameless 96X64	ISC15ANP4	OLED Switch 64X48
IS-ENG-KIT-7-FF	ISF15ACP4	Frameless 96X64	ISF15ACP4	Frameless 96X64
IS-ENG-KIT-8-R	IS18WWC1W	SW3: Cable AT715 and Connector AT097		
IS-ENG-KIT-7DF	ISC01P	OLED Display 52X36	ISF15ACP4	Frameless 96X64
IS-ENG-KIT-7SS	ISC15ANP4	OLED Switch 64X48	ISC15ANP4	OLED Switch 64X48

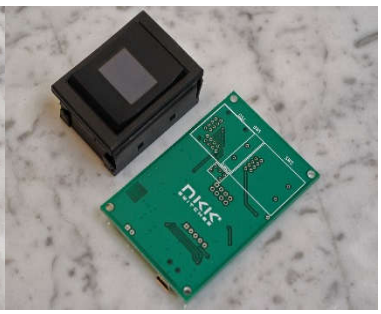
IS-ENG-KIT-7-DS

IS-ENG-KIT-7-FS

IS-ENG-KIT-7-FF



IS-ENG-KIT-8-R



Toll Free 1.877.2BUYNKK (877.228.9655) • **Phone** 480.991.0942 • **Fax** 480.998.1435
www.nkkswitches.com • **Email** engineering@nkkswitches.com

7850 East Gelding Drive • Scottsdale, AZ 85260-3420

4. Firmware Overview

The firmware is written such that it is easy to modify or add features. Only basic features are implemented in the factory default to keep the firmware easy to follow. As such it is not optimized. For example, the current firmware can send about 10 frames of Frameless OLED images per second. With the same controller, it is possible to send about 50 frames per second.

OLED Rocker is controlled independently of the color OLED versions and has its own set of attributes.

All OLEDs have the same on-board controller. However, they have different initializations and pixel fields. A code, (Eng Kit code), set in the firmware, specifies the type of OLED product at each position. This code determines that each position is initialized with the proper OLED initialization and images for the associated OLED product.

IS-ENG-KIT-7-FF	Frameless switch 1 and Frameless switch 2	00H
IS-ENG-KIT-7-FS	Frameless switch 1 and Standard switch 2	01H
IS-ENG-KIT-7-DS	Display switch 1 and Standard switch 2	02H
IS-ENG-KIT-8-R	Rocker switch	03H

It is very important to have this code correct as wrong initialization can damage the OLED product. This code is saved in the EEPROM of the microcontroller. If the EEPROM of the microcontroller has not been programmed, the default value from firmware code is saved to the EEPROM and used (found in the GetCurrentDevKit() function) which is 00H. Make sure that when modifying the firmware that the default code in the firmware is selected to match the Engineering Kit in use.

The firmware communicates to a host computer via RS232 to USB. Any data sent from the host are received via interrupt and placed in a receive buffer. The data get processed in the main program. Any transmit datum is put in a transmit buffer. Each timer interrupt transmits a byte from the transmit buffer if it is not empty. The firmware also handles read and write operations from and to the external flash memory, as well as controls the color OLEDs and the OLED Rocker. The communication protocol is outlined in following sections.

The 16M flash memory is a NOR flash and as such, when writing to the memory, it can only turn an ON cell to the OFF state. The erase command turns all the cells to the ON state. If images or other data on the flash memory needs to be changed then it is required that the erase command be sent first. The flash memory has partial erase commands available, however, as of the first firmware release date only the complete erase is implemented in the firmware.

The 7 I/O pins in the auxiliary port are initialized as inputs with weak pull-ups. These pins can be used to sense or control in user-defined custom applications.

The flash memory is set up to save up to 511 images and attributes each for all 4 switch types (Frameless, Standard, Display, Rocker). Two bytes are needed to address an image or attributes.

Address	OLED 52x36	OLED 64x48	OLED 96x64	OLED color Attributes	OLED Rocker 96x64	OLED Rocker Attribute
0001	Image	Image	Image	Attribute Block	Image	Attribute Block
0002	Image	Image	Image	Attribute Block	Image	Attribute Block
---	Image	Image	Image	Attribute Block	Image	Attribute Block
---	Image	Image	Image	Attribute Block	Image	Attribute Block
01FE	Image	Image	Image	Attribute Block	Image	Attribute Block
01FF	Image	Image	Image	Attribute Block	Image	Attribute Block
Size	3,744 bytes	6,144 bytes	12,288 bytes	16 bytes	768 bytes	16 bytes

OLED Standard/Frameless/Display Attribute Block										
Name	Current Address	End address for the loop	Timer1 for the loop	Timer2 for the loop	Next address for SW1 upon sw press	Next address for SW2 upon sw press	Jump address for SW1 at the end of the loop	Jump address for SW2 at the end of the loop	Reserved	Reserved
Range	0000 to 01FF	0000 to 01FF	00 to FF	00 to FF	0000 to 01FF	0000 to 01FF	0000 to 01FF	0000 to 01FF	Reserved	Reserved
# of bytes	2	2	1	1	2	2	2	2	1	1

OLED Rocker Attribute Block										
Name	Current Address	End address for the loop	Timer1 for the loop	Timer2 for the loop	Next address upon top press	Next address upon middle press	Next address upon bottom press	Jump address at the end of the loop	Reserved	Reserved
Range	0000 to 01FF	0000 to 01FF	00 to FF	00 to FF	0000 to 01FF	0000 to 01FF	0000 to 01FF	0000 to 01FF	Reserved	Reserved
# of bytes	2	2	1	1	2	2	2	2	1	1

Each switch has an associated active Attribute Block in the RAM. Additionally each switch is associated to an active display address. If the system detects an image needs changing, the image from the flash memory address of the active address is transferred to the switch .

An address is assigned to a switch via switch press, timer expiration, or upon power-up. Before an address is assigned to a switch, the controller checks the Attribute Block in the flash at the assigned address. If the first

two bytes are equal to the assigned address, the Attribute Block from flash is loaded to the active Attribute Block. After loading, the active display is loaded based on the attributes. If Timer1 does not equal zero, the loop timer starts running. If the first two bytes of the attribute block are not equal to the assigned address, no action is taken.

If Timer1 does not equal zero, a loop timer is started. The loop timer is calculated by multiplying Timer1 and Timer2. When the loop timer expires the displayed address is compared to the End Address. If they are equal, and the jump addresses for SW1/SW2 do not equal zero, then the jump addresses become the new active addresses for switch1 and switch2. Then the attributes and images are loaded accordingly. If the jump addresses of SW1/SW2 are equal to zero, there is no change to the active addresses and the loop restarts for the associated switch. If the displayed address and the End Address are not equal, then the displayed address + 1 is loaded as the displayed address and the image is sent to the switch. The loop timer then restarts.

When a switch is pressed, if the press addresses for SW1/SW2 of the active Attribute Block are not equal to zero they become the active addresses for switch1 and switch2, and are sent to the switches. If the press addresses of SW1/SW2 are equal to zero, there will not be any update or change to the active addresses.

5. Operational Overview

Power-up Sequence:
Upon power-up the controller execute the following steps and then enters main program.

1. Initializes all the active attributes blocks to zero.
2. Checks the current dev kit setup. If not set, defaults to two frameless switches and saves to EEPROM.
3. Sets the active address of switch#1 and the OLED rocker to 0001H (if applicable).
4. Checks the active address of switch #2 and the brightness from the setup attribute. If there is no setup attribute it defaults to address 0002H and maximum brightness (0FH).
5. Loads the attributes for both switches into memory.
6. Sends the active images to the switches.
7. Set the timer interrupt for every 0.5ms.
8. Set up the UART (RS232) communication.
9. Transmit 11H to host.

OLED Color Main Program:
The main program continuously goes through the following steps:

1. Check and execute commands if there are data in the RS232 receive buffer.
2. If the flag for switch scan is set, scan the switches and set flags.
3. If switch #1 timer has expired, the timer is stopped, report 83H and process according to the active attribute of the switch#1.
4. If switch #2 timer has expired, the timer is stopped, report 84H and process according to the active attribute of the switch#2.
5. If switch#1 is pressed, report 81H and process according to the active attribute of the switch#1.
6. If switch#2 is pressed, report 82H and process according to the active attribute of the switch#2.
7. If switch#1 is released, report C1H.
8. If switch#2 is released report C2H.
9. If switch#1 image changes, report FFH followed by switch#1 active address and update the memory for switch#1 with image from flash at the active address for the switch#1. Starts timer for switch#1 if timer 1 is non-zero.
10. If switch#2 image changes, report FEH followed by switch#2 active address and update the memory for switch#2 with image from flash at the active address for the switch#2. Starts timer for switch#2 if timer 1 is non-zero.
11. Go to step1.

Rocker Main Program:
The main program continuously goes through the following steps:

1. Check and execute commands if there are data in the RS232 receive buffer.
2. If the flag for switch scan is set, scan the switches and set flags.
3. If the switch timer has expired, the timer is stopped, report 83H and process according to the active attribute of the switch.
4. If the top switch is pressed, report 91H and process according to the active attribute of the switch.
5. If the middle switch is pressed, report 92H and process according to the active attribute of the switch.
6. If the bottom switch is pressed, report 93H and process according to the active attribute of the switch.

7. If the top switch is released, report B1H.
8. If the middle switch is released report B2H.
9. If the bottom switch is released report B3H.
10. If the switch image changes, report FDH followed by the switch active address and update the memory for the switch with image from flash at the active address for the switch. Starts timer for the switch if timer 1 is non-zero.
11. Go to step1.

Timer-Interrupt

Timer-interrupt is set for every 0.5ms. It does the following functions:

1. Increment the master half millisecond timer that the main program uses for all timers. All timers use this for a starting time and to check if the stop time has passed.
2. If the RS232 transmit buffer is not empty transmit a byte
3. Exit

7850 East Gelding Drive • Scottsdale, AZ 85260-3420

6. Communication Protocol

The controller communicates with the host via USB serial communication (115.2K, 1 start bit, 8 bit, 1 stop bit).

Communication Initiated by the Controller

The controller transmits the switch activities, timer expirations, and the addresses of images when they are displayed. The protocol for the code transmitted are explained in the Main Program in the Operational Overview.

Communication Initiated by the Host

The controller receives the data via a serial interrupt routine that places the data in the circular receive buffer. When the controller detects data in the circular receive buffer in the main program, the controller reads one byte and executes according to the following scenario.

A. If the byte is 01H the controller responds by putting 61Hin the transmit buffer and exits.
B. If the byte is a 20H to 2FH the controller responds by putting 61H in the transmit buffer. The controller checks the command procedure and processes it accordingly. If the command procedure exists and all the data is proper, the controller puts 79H in the transmit buffer and exits. If the command procedure does not exist, the data is not acceptable, or consecutive bytes are not received within 100ms then the controller puts 6EH in the transmit buffer and exits.
C. If the byte is not 01H, or 20H to 2FH, it is ignored.

7850 East Gelding Drive • Scottsdale, AZ 85260-3420

7. Commands to the Controller

Command to reboot the controller

This command reboots the controller to power-up state and transmit 11H when the reboot is complete.

Command format:	**24H**
Transmit format:	(xxH)

Command to check communication

This command is used to check if the controller is on-line.

Command format:	**01H**
Transmit format:	(xxH)

The controller transmits 61H back to the host:

Command to change brightness level

This command changes the brightness level for both switches. The brightness level changes immediately and stay in effect until reboot or power off

Command format:	**27H**	4EH	[Brightness level]
Transmit format:	(xxH)	(xxH)	(xxAH)

[Brightness level] is one byte transmitted in ASCII HEX. The acceptable range is 00H to 0FH, where 0FH is the brightest level.

Command to query the controller for mode, controller, and firmware version

This command queries the controller for the mode as set by the mode select switch, the controller name, and the firmware version installed:

Command format:	**26H**	52H	58H
Transmit format:	(xxH)	(xxH)	(xxH)

Example: The command is sent. The controller responds with the following:

61	xx	53 30 32 4C 31	31 30	79
61H	[dev kit]	[Controller name]	[Version]	79H

[dev kit] is one byte. The possible responses are listed in section 4 of this document
[Controller name] is five bytes. 53H 30H 32H 4CH 31H (S02L1)
[Version] is two bytes. For example, a 31h 30H would be version 1.0 (converted from ASCII hex)

7850 East Gelding Drive • Scottsdale, AZ 85260-3420

Command to designate the current Engineering Kit type

This command changes the current Engineering Kit being used and saves the changes to EEPROM.

Command format:	**26H**	44H	[Eng Kit code]
Transmit format:	(xxH)	(xxH)	(xxAH)

[Eng Kit code] is one byte transmitted in ASCII hex. The following options are currently supported:

IS-ENG-KIT-7-FF	Frameless switch 1 and Frameless switch 2	00H
IS-ENG-KIT-7-FS	Frameless switch 1 and Standard switch 2	01H
IS-ENG-KIT-7-DS	Display switch 1 and Standard switch 2	02H
IS-ENG-KIT-8-R	Rocker switch	03H

A reboot is required after sending this command. If an invalid code is sent, the system defaults to two frameless switches (00H).

Command to manually set images for Switch#1 and switch#2

Command format:	**2DH**	[Switch Identifier]	[Address for switch#1)	[Address for switch#2]
Transmit format:	(xxH)	(xxH)	(xxAH)	(xxAH)

[Address for switch#1] is two bytes transmitted in ASCII HEX. The acceptable range is 0001H to 01FFH.
[Address for switch#2] is two bytes transmitted in ASCII HEX. The acceptable range is 0001H to 01FFH.

The controller will assign the desired addresses to the switches.

Commands that Disable Switch and Timer Execution

Upon transmitting any of the following commands, the timers stop running and attributes for the switches activity do not execute. However, the switches are still scanned and reported. The attributes execution is enabled upon reboot, power-up.
The reason for disabling attribute execution is to enable faster download of images and attributes.

Command to disable switch and timer execution

This command disable switch and timer execution

Command format:	**26H**	51H	5AH
Transmit format:	(xxH)	(xxH)	(xxH)

Command to erase the flash memory

This command erases the memory by turning all the cells ON. Before proceeding, the host needs to wait for the 79H to be received from the controller, indicating the memory is erased. The erase time for this flash memory is approximately 2 minutes.

Command format:	**21H**	55H	AAH	52H	52H
Transmit format:	(xxH)	(xxH)	(xxH)	(xxH)	(xxH)

Command to download setup data

This command downloads the setup data. The setup data is saved on the flash memory:

Command format:	**2AH**	56H	[0000H]	[Switch 2 start address]	Brightness level]	[Reserved]
Transmit format:	(xxH)	(xxH)	(xxAH)	(xxAH)	(xxAH)	(xxAH)
Number of bytes	1	1	2	2	1	11

[Brightness Level] is one byte sent in ASCII HEX format. Range 00H to 0FH
[Switch 2 start address] is two bytes with the value of 0001H to 01FFH sent in ASCII HEX format. This is the address for the image that switch#2 displays upon power-up or reboot.
[Reserved] is 11 bytes of 00H sent in ASCII HEX format. This is for future use. The customer can configure the last two bytes as needed for their application. The rest of the reserved bytes should be zero. The controller saves the setup to flash memory.

This setup data can be read using the attribute upload command.
On startup the controller checks if the setup attribute is valid, and if not, the setup is not used.

7850 East Gelding Drive • Scottsdale, AZ 85260-3420

Command to download an image

This command downloads an image from the host to the flash memory location:

Command format:	**28H**	[Switch Identifier]	[Address]	[Image]
Transmit format:	(xxH)	(xxH)	(xxAH)	(xxAH)

[Switch Identifier] is one byte with a value depending on switch type as shown below:

OLED64X48 Standard Switch	53H
OLED52X36 Display	54H
OLED96X64 Rocker	55H
OLED96X64 Frameless Switch	56H

[Address] is two bytes with value of 0001H to 01FFH sent in ASCII HEX format.
[Image] is transmitted in ASCII HEX format.

Display Image Bytes	
Byte	**Description**
1	Top left pixel of the display
2	
•	
•	
•	
3743	
3744	Bottom right pixel of the display

Standard Image Bytes	
Byte	**Description**
1	Top left pixel of the display
2	
•	
•	
•	
6143	
6144	Bottom right pixel of the display

Toll Free 1.877.2BUYNKK (877.228.9655) • **Phone** 480.991.0942 • **Fax** 480.998.1435
www.nkkswitches.com • **Email** engineering@nkkswitches.com

Frameless Image Bytes	
Byte	**Description**
1	Top left pixel of the display
2	
•	
•	
•	
12287	
12288	Bottom right pixel of the display

Rocker Image Bytes	
Byte	**Description**
1	Top left pixel of the display
2	
•	
•	
•	
767	
768	Bottom right pixel of the display

Command to upload an image

This command uploads an image from the flash memory location to the host:

Command format:	**29H**	[Switch Identifier]	[Address]
Transmit format:	(xxH)	(xxH)	(xxAH)

[Switch Identifier] is one byte with a value depending on switch type as shown below:

OLED64X48 Standard Switch	53H
OLED52X36 Display	54H
OLED96X64 Rocker	55H
OLED96X64 Frameless Switch	56H

[Address] is two bytes with value of 0001H to 01FFH sent in ASCII HEX format.

The controller transmits the image bytes back in ASCII HEX format.

Toll Free 1.877.2BUYNKK (877.228.9655) • **Phone** 480.991.0942 • **Fax** 480.998.1435
www.nkkswitches.com • **Email** engineering@nkkswitches.com

Command to download attribute block

This command downloads an attribute block from the host to the flash memory location:

Command format:	**2AH**	[Switch Identifier]	[Attribute block]
Transmit format:	(xxH)	(xxH)	(xxAH)

[Switch Identifier] is one byte with a value depending on switch type as shown below:

OLED Rocker	55H
OLED All Color Types	56H

[Attribute Block] is 16 bytes transmitted in ASCII HEX format.

The first two bytes of the Attribute Block are the desired address of the block. The Attribute Block is stored in flash memory according to these first two bytes.
When the controller reads the Attribute Block, it compares the desired address to the saved address. If they do not match, all attributes are deemed invalid and do not get loaded.

Command to upload an attribute block

This command uploads an attribute block from the flash memory location to the host:

Command format:	**2BH**	[Switch Identifier]	[Address]
Transmit format:	(xxH)	(xxH)	(xxAH)

[Switch Identifier] is one byte with a value depending on switch type as shown below:

OLED Rocker	55H
OLED All Color Types	56H

[Address] is two bytes with the value of 0000H to 01FFH sent in ASCII HEX format.

The controller transmits the 16 bytes Attribute Block via RS232 in ASCII HEX format.

8. Hardware Considerations

The power can be supplied to the Engineering Kits via USB port or J2 header. The voltage range can be 4.5V to 5.5V. Maximum current consumption is 125mA. A fuse and a Zener diode are used for high voltage input protection.

The PIC18F46K40 microcontroller is used in the Engineering Kits. The VDD for the microcontroller is 3.3V. A reset chip at the location PC1 is implemented to reset the microcontroller at 2.9V to comply with the OLED switch turn OFF requirement. Once the voltage gets to the 2.9V the microcontroller is reset causing the charge pump circuit disabling the OLED voltage. The OLED logic voltage is acceptable till 2.4V. The charge pump provides the OLED voltage and it is normally disabled with a pull-down resistor on the enable pin. The microcontroller enables it by setting the enable pin to high.

The auxiliary ports I/O are directly connected to microcontroller pins without any safety circuits. Care must be taken not to exceed VDD voltage.

9. Programming Setup

The Engineering Kits are designed to be easily programmed and have a 5-position SIP footprint for this purpose (JP1). They can be programed by a variety of different programmers. Our recommendation is to use the PICkit 3 with the IS-PA2 adaptor and a USB power source. The IS-PA2 is an adaptor with a 5-position SIP header on one side and a 5-position, spring-loaded header on the other side.

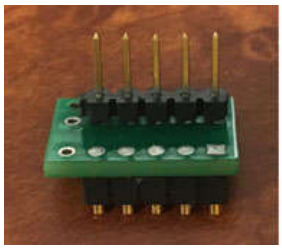
Note: Position 1 of the PICit3 must be at position 1 on the PCB or damage could occur. The 5V power must be supplied; usually through the USB cable although there are through-hole pads on the PCB for an alternative 5V power connection.

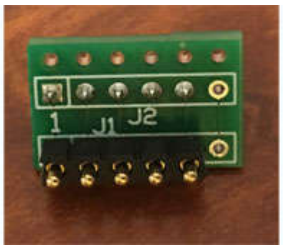
Connector callout of the IS-PA2 and controller JP1.

J1	J2	Controller JP1	
1	1	1	MCLR
2	2	2	VDD
3	3	3	GROUND
4	4	4	PGD (data)
5	5	5	PGC (clock)

IS-PA2 on a control board:

IS-PA2 Board photos:





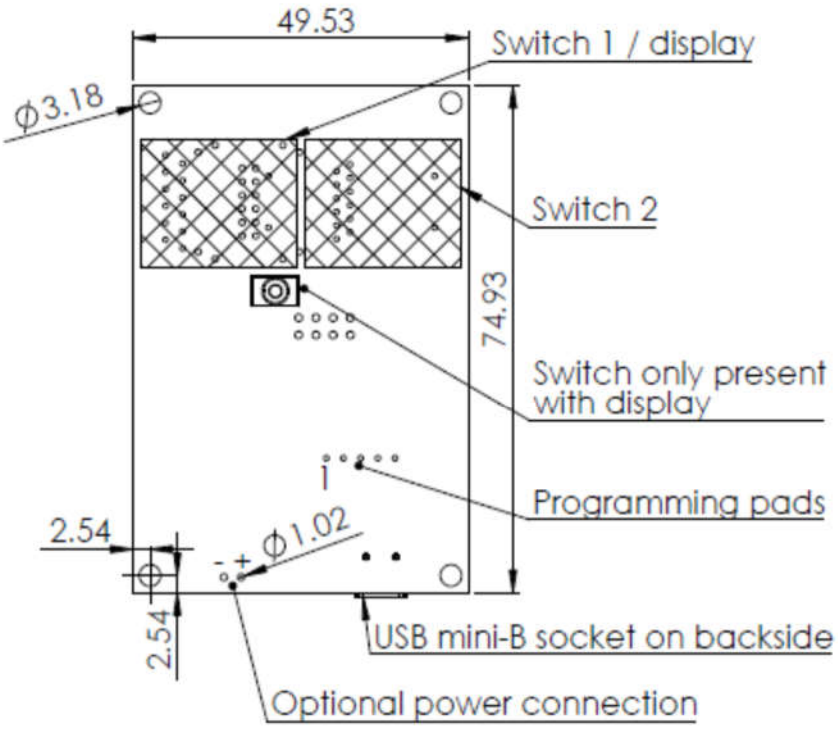
10. Modifying the Firmware

1. Download MPLAB-X from www.microchip.com.
2. Install MPLAB-X to the default location with the default options.
3. Download the MPLAB XC8 compiler from www.microchip.com.
4. Install the MPLAB XC8 compiler to the default location with the default options
5. Extract the firmware source code to a location of your choice.
6. Open MPLAB-X IDE (Integrated Development Environment).
7. Press the open project button and navigate to the directory the source code was extracted to, then click the directory and click “Open Project”.
8. Click the “Production” menu at the top, and select “Clean and Build Project”. Wait for the build to complete.
9. Attach a PICKIT3 (or equivalent) to the PCB board. This can be done using a IS-PA2 adapter (available for purchase on the NKK website) or by soldering a header into JP1 and attaching the PICKIT3.
10. Press the debug project button and wait for the output window to say “Running”.
11. Observe the program is now running on the board and images appear on the SmartSwitch.
12. Press the stop button. You are now ready to start developing for the SmartSwitches.

7850 East Gelding Drive • Scottsdale, AZ 85260-3420

11. Board Dimension

Typical dimensions. Sockets and components vary with kit type.



Dimensions in mm.

Toll Free 1.877.2BUYNKK (877.228.9655) • Phone 480.991.0942 • Fax 480.998.1435
www.nkkswitches.com • Email engineering@nkkswitches.com

7850 East Gelding Drive • Scottsdale, AZ 85260-3420

12. 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**.

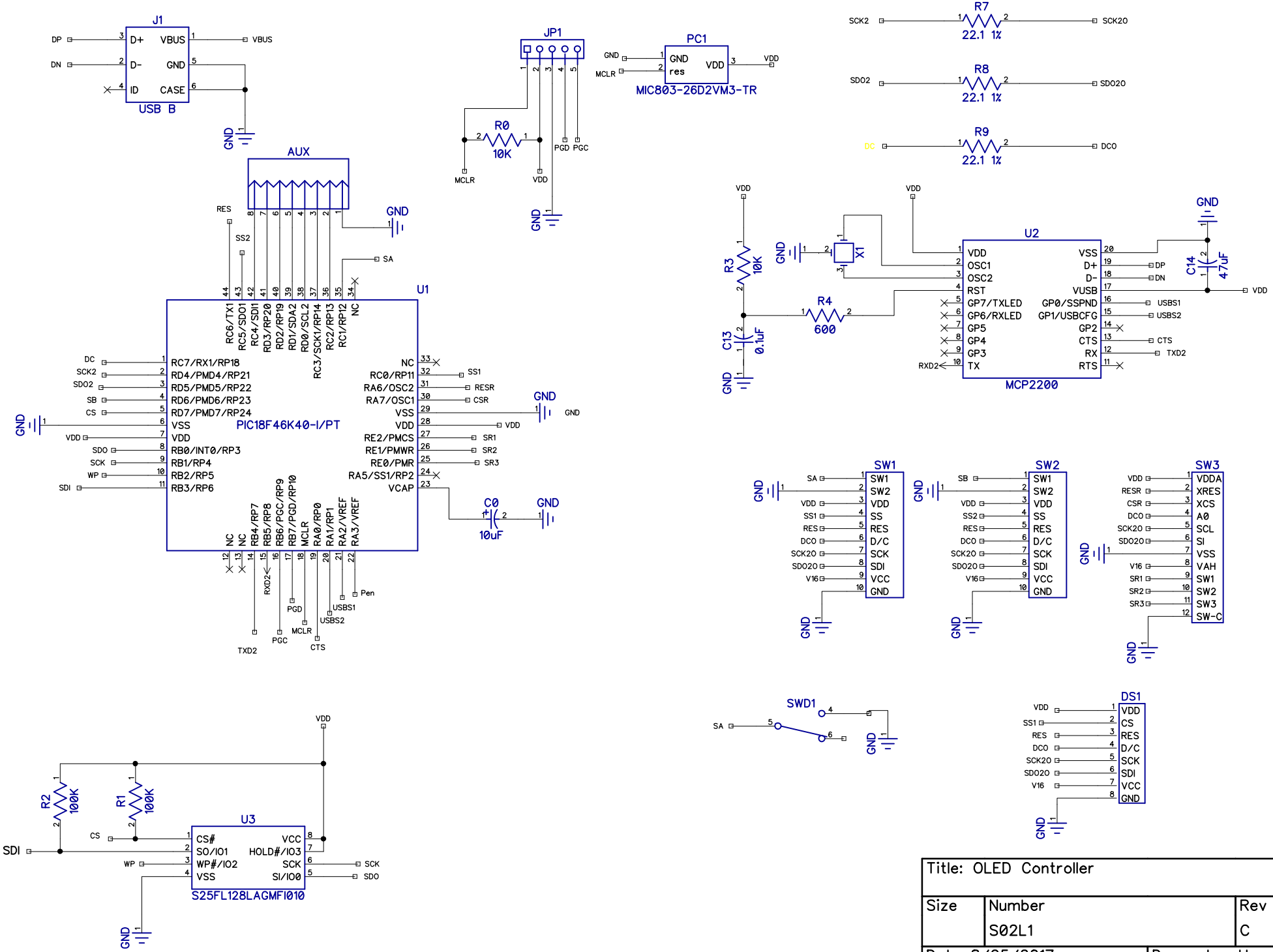
   All addresses and most data are sent using this format.

Address — A two-byte value ranging from 0001H to 01FFH representing the 511 memory locations for pictures and attributes on the flash memory.

13. Schematics

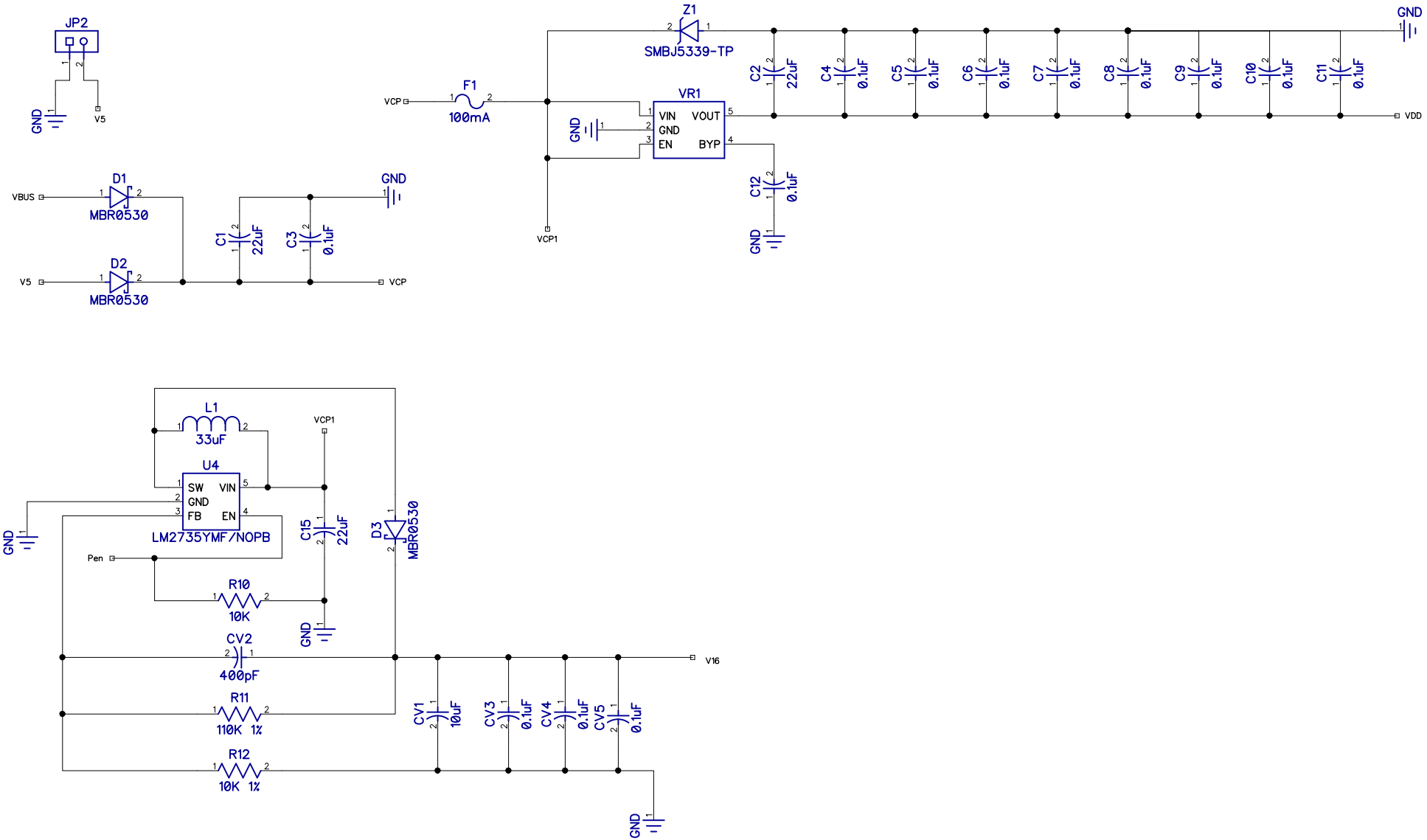
See schematics on next page(s).

RevNo	Revision note	Date	Signature	Checked



Title: OLED Controller		
Size	Number S02L1	Rev C
Date 8/25/2017		Drawn by: Hassan
Filename		Sheet 1/2

RevNo	Revision note	Date	Signature	Checked



Title: OLED Controller		
Size	Number S02L1	Rev C
Date		Drawn by
Filename		Sheet 2/2