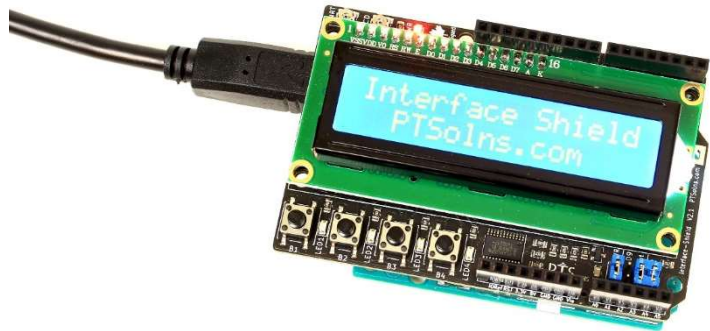


Interface-Shield Kit

1 DESCRIPTION

The PTSolns *Interface-Shield* is a shield designed to interface the user with their Arduino project. The *Interface-Shield* offers a range of features such as programmable buttons and LEDs, hardware and software debouncing filters, configurable addresses, dimmable LCD backlight (onboard potentiometer and PWM pin), interrupt, and much more.



The *Interface-Shield* communicates with the microcontroller via the I2C bus, and therefore allows the user to fully utilize all the digital and analog pins of the microcontroller, as none are occupied by the LCD. Signal quality is important whenever working with I2C. Therefore, the shield offers slotted large copper pads designed to connect the probes of an oscilloscope in order to view the I2C bus signals.

For tutorials and example sketches using the *Interface-Shield*, the user can install the custom library in Arduino IDE “PTSolns_InterfaceShield”, or is referred to: https://github.com/PTSolns/PTSolns_InterfaceShield

Table of Contents

1 DESCRIPTION	1
2 DOCUMENT REVISION HISTORY	2
3 PRODUCT FEATURES.....	3
3.1 Compatibility	3
3.2 I2C and Shield Address	4
3.3 Programmable Buttons and LEDs.....	4
3.4 Microcontroller Pin Access.....	5
3.5 LCD Backlight.....	5
3.6 Contrast.....	6
3.7 Interrupt.....	7
3.8 Power LED	7
3.9 Oscilloscope I2C Connection Pads.....	8
3.10 Mark of Authenticity	8
4 CUSTOM LIBRARY AND APPLICATION.....	9
5 KIT PACKAGE CONTENTS.....	10
6 RESOURCES.....	11

2 DOCUMENT REVISION HISTORY

Current document revision is Rev 0.

3 PRODUCT FEATURES

This section highlights the features of the *Interface-Shield*. Some of the features are graphically shown in Figure 1. More detail is provided in the corresponding sub-sections.

The *Interface-Shield Kit* is not fully assembled, and soldering is required. Note that all the SMD components are pre-soldered and only through-hole components are required to be soldered.

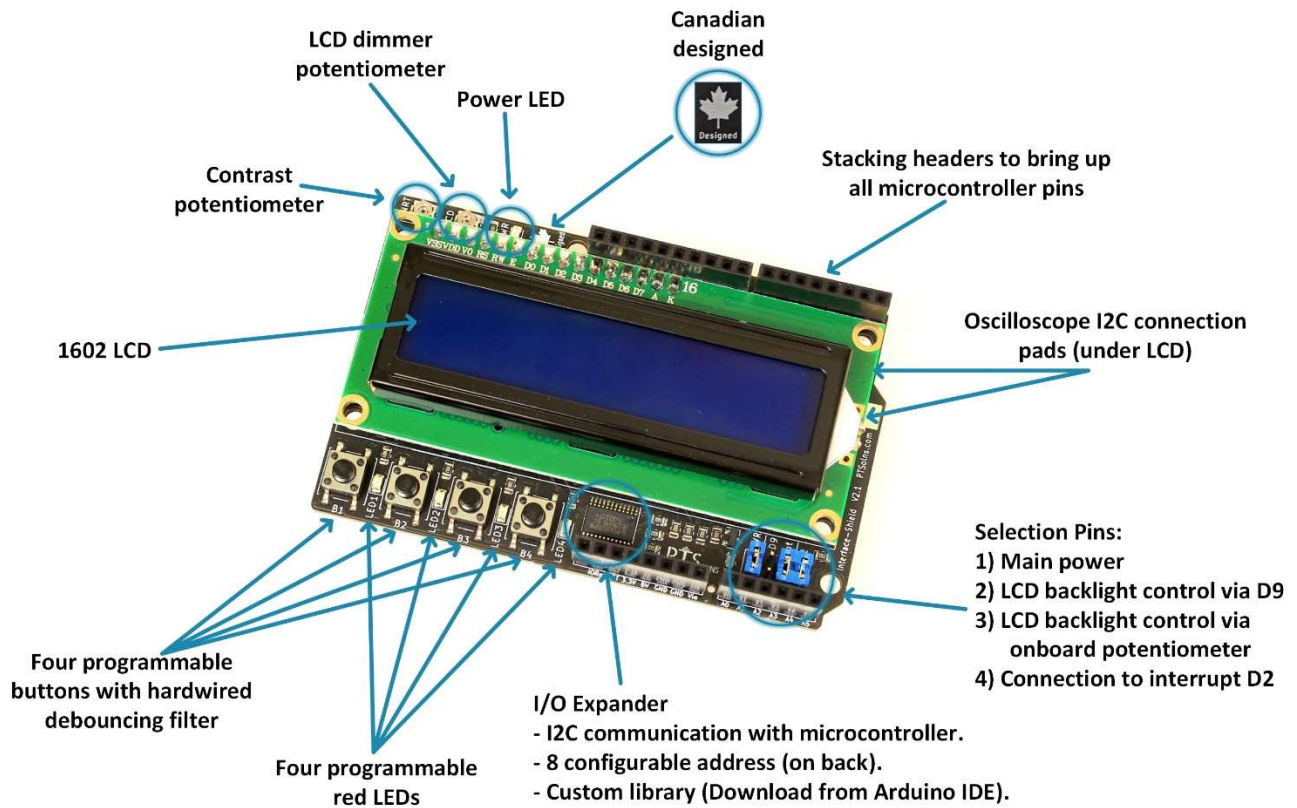


Figure 1: Features of the Interface-Shield.

3.1 Compatibility

The *Interface-Shield* is compatible with the following microcontrollers:

- Arduino Uno R3 (THT or SMD versions)
- Arduino Mega 2560 R3
- Arduino Leonardo
- Any other similar microcontroller with 5V logic and same pinout

3.2 I2C and Shield Address

The *Interface-Shield* communicates via the I2C bus with the microcontroller. Therefore, the shield has a specific address that the user must specify in their sketch. The default address is 0x27 but can be changed to a total of eight different addresses by bridging the appropriate solder jumper pads on the back of the shield, as shown in Figure 5. For example, if the jumper pads A0 and A2 are bridged, then the shield address becomes 0x22. For details of how to use the shield address the user is referred to the example Arduino IDE sketch titled “IO_Expander_Address.ino”.

To check the currently set shield address, the user can run the common “I2C scanner” sketch. The user should ensure that any other I2C device connected to their project does not have the same address as the one set on the *Interface-Shield*.

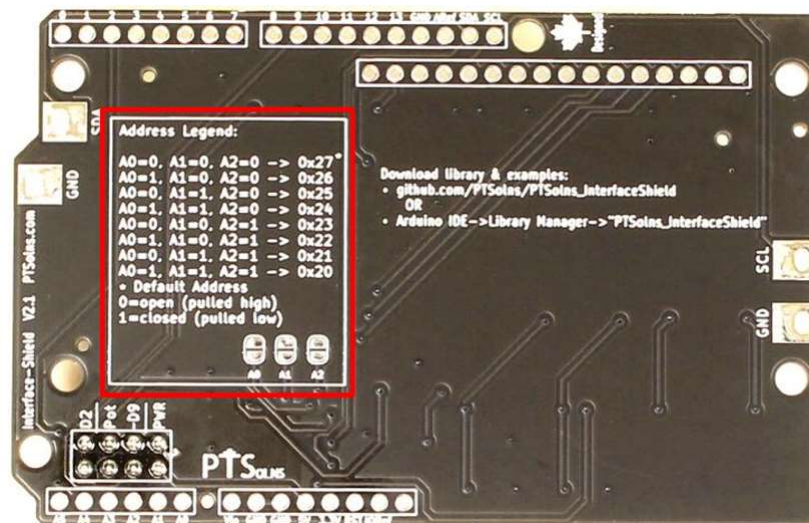


Figure 2: Address configuration of the Interface-Shield.

3.3 Programmable Buttons and LEDs

The *Interface-Shield* has four programmable buttons labelled B1, B2, B3 and B4, and four programmable LEDs labelled LED1, LED2, LED3 and LED4, as shown in Figure 3.

When programming these buttons or LEDs in an Arduino IDE sketch note that their names are already defined in the library (B1, B2, B3, B4, LED1, LED2, LED3, LED4). Therefore, the user should not define other items in the same sketch with the same names. For details of how to program these buttons and LEDs the user is referred to the example Arduino IDE sketch titled “Buttons_and_LEDs.ino”.

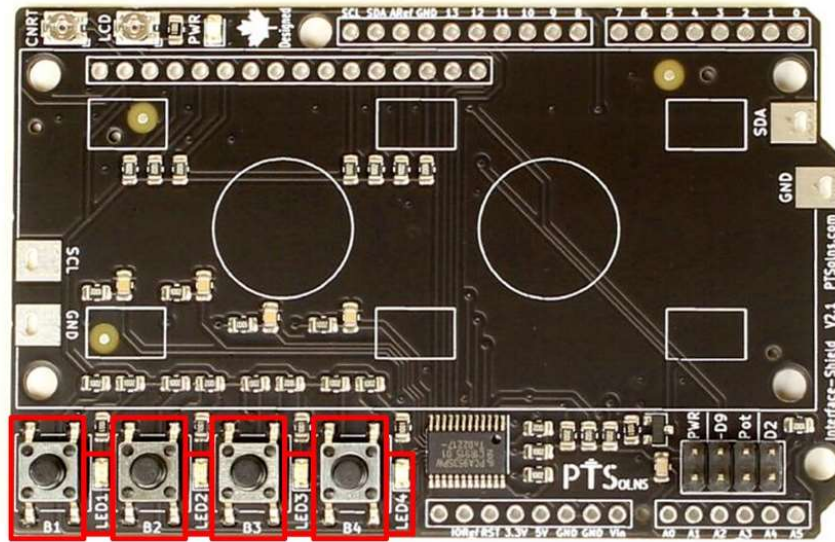


Figure 3: Programmable buttons and LEDs.

The four programmable buttons each have an onboard hardwired RC debouncing filter to smooth out button presses and limit accidental false triggering. Furthermore, the custom library has a command to enable a software debouncing filter. For details of how to program to use the software debouncing the user is referred to the example Arduino IDE sketch titled “Debouncing.ino”.

3.4 Microcontroller Pin Access

The *Interface-Shield* communicates with the microcontroller via the I2C bus. The user should be aware which two pins are used on the microcontroller for I2C communication. For example, on the Arduino Uno R3, these pins are A4 (SDA) and A5 (SCL). When using this shield on an Uno R3, these two pins will be used for I2C and therefore they cannot be used for any other tasks besides I2C communication. Similarly for other compatible microcontrollers that may assign different pins for I2C communication. Note one of the main advantages of the *Interface-Shield* communicating over the I2C bus is that no other digital pin or analog pin (besides the two used for I2C communication) are occupied. Therefore, the user has full flexibility which pins to use in their project.

In order to access all of the pins of the microcontroller, stacking female headers are provided in the *Interface-Shield Kit*, as shown in Figure 10. This makes quick prototyping of various I/O devices to the microcontroller easy and convenient.

3.5 LCD Backlight

The LCD backlight can be controlled by two settings, as shown in Figure 4, as follows:

- Jumper cap on “D9”
- Jumper cap on “Pot”

If the setting for D9 is selected, then the LCD backlight is powered by pin D9 on the microcontroller. This pin is a PWM capable pin and therefore the LCD becomes dimmable via software. For details of how to program pin D9 to dim the LCD backlight the user is referred to the example Arduino IDE sketch titled “LCD_Backlight_Dimming.ino”.

If the setting for Pot is selected, then the LCD backlight can be adjusted by the onboard potentiometer, labeled “LCD”.

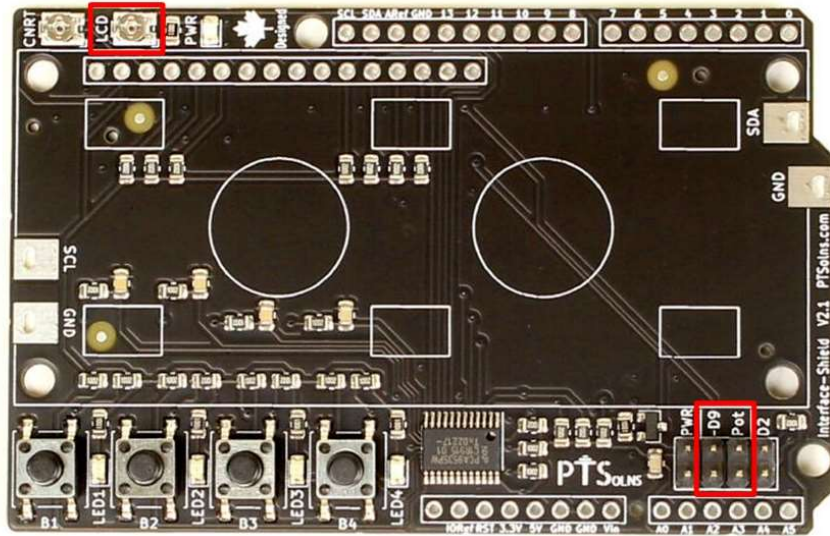


Figure 4: LCD backlight control of the Interface-Shield.

3.6 Contrast

The character contrast on the LCD can be adjusted by an onboard potentiometer, labelled “CNRT”, as shown in Figure 5.

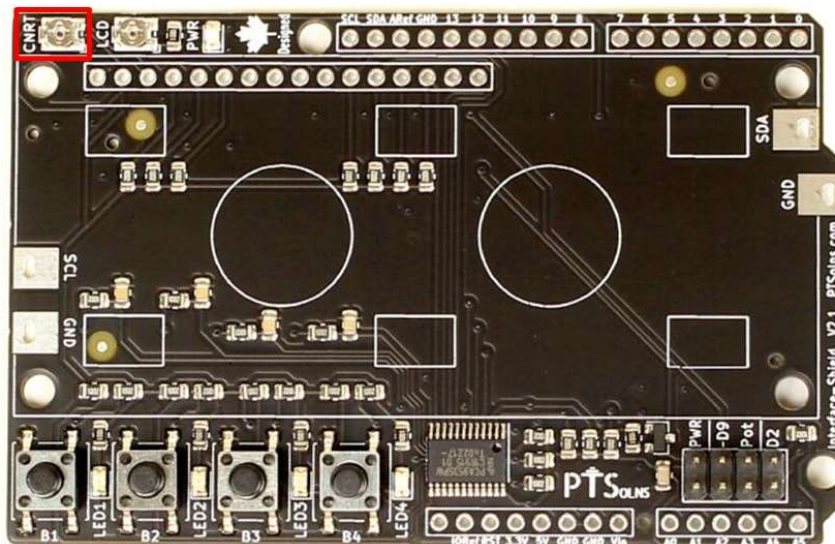


Figure 5: Contrast potentiometer onboard the Interface-Shield.

3.7 Interrupt

When one of the buttons is pressed, an interrupt signal can be sent to the microcontroller connected to pin D2 as shown in Figure 6. This can be done by placing a jumper cap on the D2 selection. If interrupt is not used, the user can simply leave these pins open. Note that there are limitations on how to implement this interrupt in the sketch, such as restricted function calls, etc. For details of how to use the interrupt feature the user is referred to the example Arduino IDE sketch titled “Interrupt.ino”.

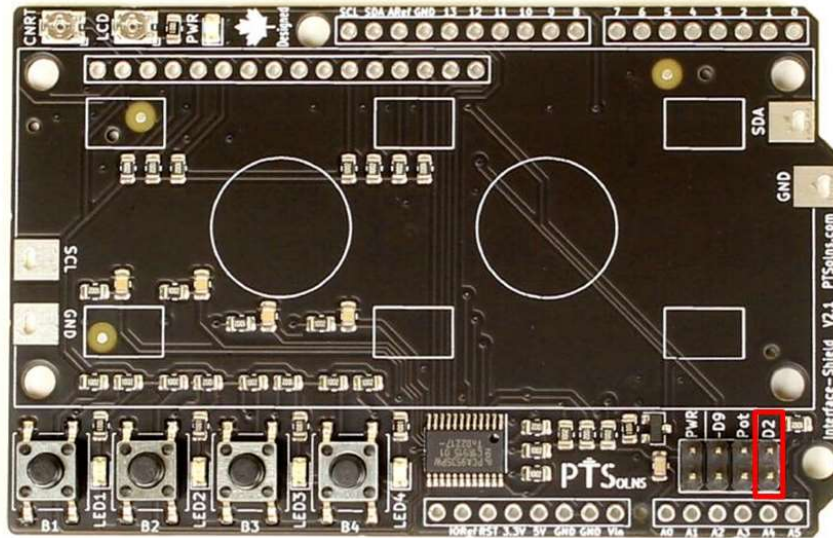


Figure 6: Interrupt jumper cap to D2 on Interface-Shield.

3.8 Power LED

The *Interface-Shield* has an onboard power LED, labelled “PWR”, as shown in Figure 7. The power LED will be ON whenever the jumper cap, also labelled “PWR” is closed and the microcontroller is supplying power.

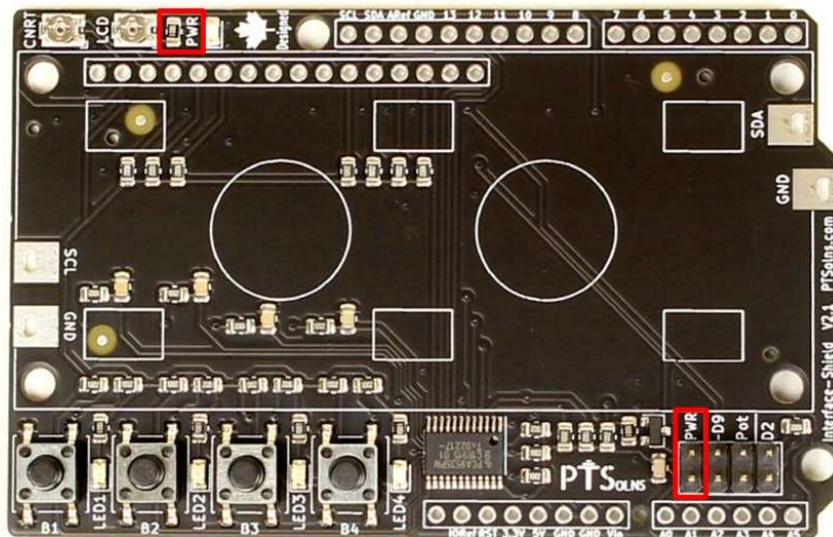


Figure 7: Power LED PWR on the Interface-Shield.

3.9 Oscilloscope I2C Connection Pads

The *Interface-Shield* has slotted large copper pads that can be used to connect probes of an oscilloscope to view the I2C bus, as shown in Figure 8.

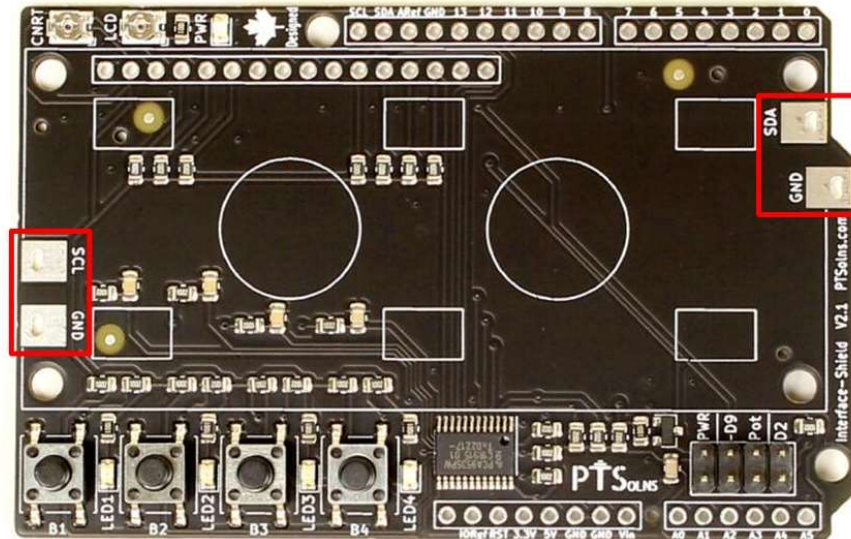


Figure 8: Oscilloscope I2C connection pads on the Interface-Shield.

3.10 Mark of Authenticity

Authentic PTSolns PCBs have a black solder mask color and are marked with the “PTSolns” logo in white silkscreen printing. The “Canadian Designed” symbol, consisting of the Canadian Maple Leaf with the word “Designed” underneath, can also be found on the PCB in white silkscreen printing. The “PTSolns” trademark and the “Canadian Designed” symbols are shown in Figure 9.

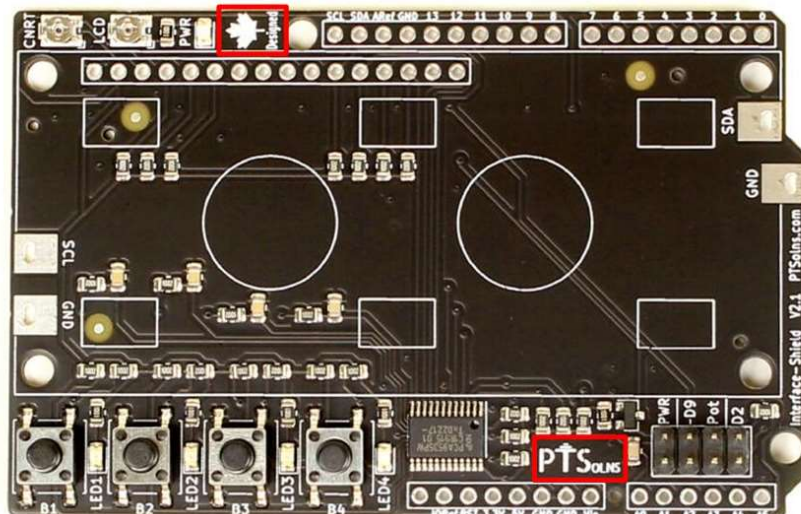


Figure 9: The “Canadian Designed” symbol found on authentic PTSolns PCBs.

4 CUSTOM LIBRARY AND APPLICATION

The *Interface-Shield* requires a custom library in order to communicate with a microcontroller. This library can be downloaded directly in the Arduino IDE by searching in the library section the name “PTSolns_InterfaceShield”. Installing this library also gives the user a range of example sketches that demonstrate how to use various features.

Alternatively, the custom library can be downloaded from Github with the following link:

https://github.com/PTSolns/PTSolns_InterfaceShield

Figure 10 shows a typical application of the *Interface-Shield*, in which the shield is plugged into an Arduino Uno R3. The microcontroller was programmed using the custom library such that a text is displayed on the LCD.

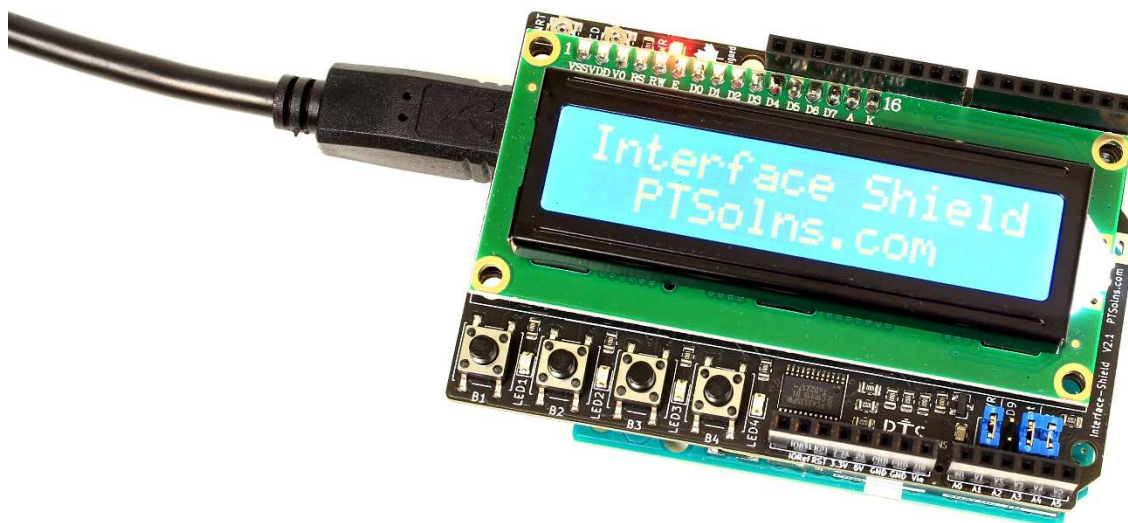


Figure 10: Typical application of the Interface-Shield.

5 KIT PACKAGE CONTENTS

The following items are included in the Interface-Shield Kit. **The kit comes unassembled, and soldering is required. Note that all the SMD components are pre-soldered and only through-hole components are required to be soldered.**

- 1pc PCB PTSolns *Interface-Shield*
- 1pc 1602 LCD
- 1pc 1X16 pin short leg male header
- 1pc 1X6 pin stacking female header
- 2pcs 1X8 pin stacking female header
- 1pc 1X10 pin stacking female header
- 3pcs jumper cap*
- 1pcs mini screwdriver*

* Color may vary.

6 RESOURCES

The following are resources relating to the *Interface-Shield*.

- In order to communicate with the *Interface-Shield* the user needs to install the corresponding Library.

If installing the library in Arduino IDE, search in the library section the name “PTSolns_InterfaceShield”. Once installed, several example sketches are available that present various features.

If installing the library elsewhere the user can download from Github using the following link.

https://github.com/PTSolns/PTSolns_InterfaceShield