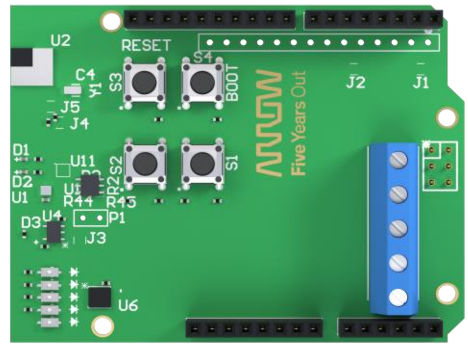
SLASH1000A User guide



1 Overview

The SLASH1000A is a development and prototyping tool built by Arrow Electronics with the intent to speed time to market and increase ease of development for Bluetooth integrated products. This shield provides a host of useful components suitable for integration with the common Arduino prototyping environment. The shield is designed to work with Arduino Uno based boards as well as Arduino MKR based boards as the host processor. Libraries are also provided to aid in reducing software development times. The SLASH1000A is broken into three categories: Core, Sensing, and Control.

The SLASH1000A Core showcases the BGX13S from Silicon Labs. This Wireless Xpress product enables Bluetooth 5 connectivity with no wireless programming required. The host processor communicates with the SLASH1000A using a UART and several configurable General Purpose Input Output (GPIO) options to facilitate a robust and reliable Bluetooth connection to other BGX devices or smart phone applications. The functionality provided by the Bluetooth Xpress chipset is detailed in Section 2.

Several common sensors are included on the SLASH1000A. A SI7050 temperature sensor from Silicon Labs allows for simple low power environmental monitoring. A VEML6030 Ambient Light sensor allows for high resolution ambient light monitoring as well as feedback from the integrated lighting controllers. For details on the usage of these subsystems see Section 3.

The Control category is comprised of onboard LEDs and a 0-10V lighting output for easy testing and integration with existing lighting products. Section 4 details the design, operation, and configuration of this system. A full bill of materials (BOM) and schematic files can be found at the end of this document.

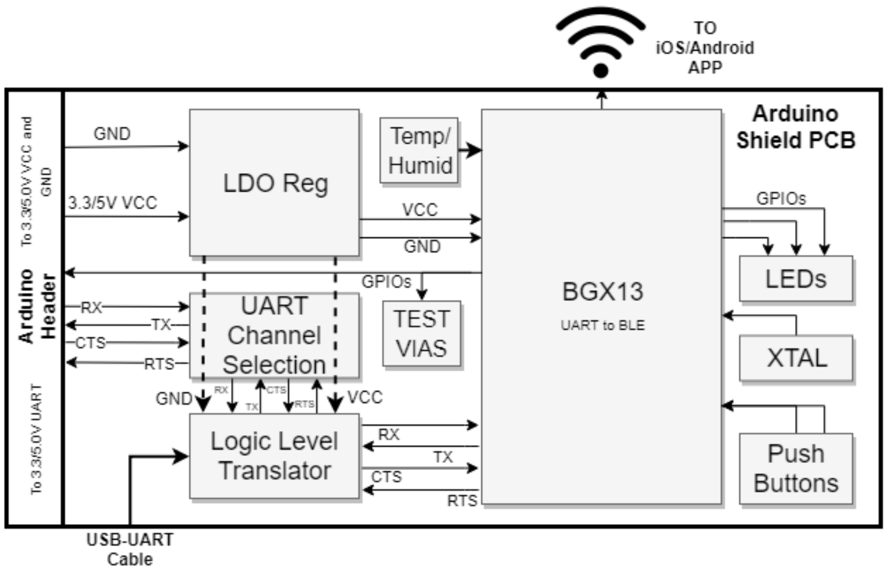


Figure 1 – SLASH1000A Block Diagram

2 Contents

3 BGX13

The BGX13 is the core of the SLASH1000A Shield. This serial replacement module eliminates the complexity of Bluetooth Firmware development while adding the connectivity needed by many applications.

3.1 Background

The BGX13 from Silicon Labs can serve as device-to-device cable replacement or enable smartphone connectivity through a simple Xpress Bluetooth Library. The device is Bluetooth 5 compliant in order to allow for a long service life. The BGX13 is available as a System-in-Package (SIP) Module or a PCB Module. The SIP package is used in the SLASH1000A to demonstrate the smaller form factor. The module functions with the help of only a 32.768kHz crystal. The BGX13S antenna leverages a copper clearance area to further reduce the number of external components needed.

3.2 Connections

The BGX13S is connected to the host processor (as detailed in Section 3.3). To facilitate host processors of higher voltage levels, logic level translators are included on the SLASH1000A as per Section 6. The connections between the BGX and SLASH1000A/Host are detailed in Table 1 below. All pins connecting to an Arduino pin are equipped with logic level translators to ensure safe inter-operability. For more details on the possible configurations and uses of the BGX13 GPIOs, please see the BGX13S Datasheet.

BGX13		SLASH1000A	
NAME	PIN #	UNO PIN #	MKR PIN #
GPIO0	10	7	7
GPIO1	11	6	6
GPIO2	12	8	8
GPIO3	17	9	9
GPIO4	18	LED D1	
GPIO5	26	LED D2	
GPIO6	27	PUSH BUTTON S1	
GPIO7	28	PUSH BUTTON S2	
UART_TX	13	5	14
UART_RX	14	4	13
UART_CTS	15	3	N/A
UART_RTS	16	2	N/A
RESETn	44	PUSH BUTTON S3	
BOOT	33	PUSH BUTTON S4	

Table 1 - BGX13S Connections

3.3 Arduino + BGX13

The SLASH1000A is designed to be used with genuine Arduino Boards and mates with both Arduino UNO and Arduino MKR formats. The officially supported models, which the libraries and examples have been tested on and developed for are: Arduino UNO R3, Arduino MKR1000, Arduino MKR1010. Due to the Arduino MKR boards featuring a reduced set of GPIO connections, some features such as control flow are not enabled. All core functionality is retained for both MKR and UNO boards. The libraries and examples will run on both; custom applications or modification are best run on the MKR platform due to the significantly larger memory and enhanced stability.

It is possible to use the SLASH1000A with other boards that use the standard Arduino footprints; however, these boards' specifications and operating parameters will vary and are not officially supported.

4 Sensors

The SLASH1000A has two environmental sensors included onboard to facilitate data aggregation and transfer over the BGX13S. The sensors interface directly with the Arduino over I2C. An Arduino library and an example sketch are supplied to provide a framework for interacting with these chips.

4.1 Lux

The VEML6030 from Vishay provides ambient and white light sensing. The 16-bit digital sensor is read over I2C and is intended for easy implementation. The default 7-bit address I2C is 0x10. Various configurations are available and can be achieved by writing to specific memory locations at this address. These registers and their functions can be found in the VEML6030 datasheet.

The configurations needed for operation as part of the SLASH1000A as well as the conversions to understand the digital readout as a lux (lx) value are outlined below. All referenced functions are members of the VEML6030 class in the ArrowBGXShield library. See Section 7.1 for details.

4.1.1 Configurations

The VEML6030 is configurable through six registers written via I2C interface. These registers program the gain, integration time, interrupts, and other settings. Basic operation is possible with many of these settings left in their default state or tuned off. However, the chip defaults to sleep mode and must be set to active by writing ‘b0’ (‘b’ denotes binary representation) to bit zero (the Least Significant Bit - LSB) of command register zero: “ALS Configuration Register 0”. Writing ‘0x00’ (‘0x’ denotes hexadecimal representation of a number) to register zero sets the integration time to 100ms, gain to 1x, and the power to on. This configuration is ensured in the provided demo software and any time the VEML6030 “DefaultConfig” function is called. All other configuration modifications are optional and facilitated through library functions and pre-defined variables.

4.1.2 Reading

A memory register internal to the VEML6030 contains the latest ambient light sensing reading. This is accessed by a read command issued over the I2C bus. This read is packaged within the “getALS” or “getWhite” functions provided in the C++ library files. Once the device has been configured using the “DefaultConfig” function or custom configuration settings, these functions will return the most recent sensor value and can be converted to lux values according to Section 4.1.3 or Vishay Application Note 84367.

4.1.3 Conversion

The lux value is calculated by multiplying the ALS readout (as a decimal value) by a predetermined scalar based on the gain and integration time. This scalar is the resolution of the reading. These values are seen below in Table 2. Reference Vishay Application Note 84367 for additional details.

	Gain 2	Gain 1	Gain 1/4	Gain 1/8
Integration Time	**Typical Resolution**			
800ms	0.0036	0.0072	0.0288	0.0576
400ms	0.0072	0.0144	0.0576	0.1162
200ms	0.0144	0.0288	0.1152	0.2304
100ms	0.0288	0.0576	0.2304	0.4608
50ms	0.0576	0.1152	0.4608	0.9616
25ms	0.1152	0.2304	0.9616	1.8432

Table 2 - Resolution for VEML6030

4.2 Temperature

The SI7050 from Silicon Labs provides real time temperature monitoring with ±1.0°C accuracy. Other members of the SI705X line share this common footprint and control methods but offer accuracies of up to ±0.1°C. This 14-bit I2C controlled sensor has a default 7-bit I2C address of 0x40. A single register is available to control the resolution of the temperature reading. No changes to the power-on settings are needed. The function of the control register is detailed in the SI7050 datasheet and changes can be made using I2C writes from the host controller.

Below, the procedure followed to attain a temperature reading and convert it to a human readable format is outlined. Any referenced software functions are from the SI7050 class in the ArrowBGXShield library files as per Section 7.1.

4.2.1 Temperature Readings

The temperature sensor value is attained in a two-step process. This first step is to issue a command to the SI7050 to perform a temperature measurement. This is accomplished by issuing a single byte (‘0xF3’) to the device. The function “startTempMeasure” handles this operation. The subsequent step is reading two bytes from the sensor to attain the temperature data (“Temp Code”). The 16-bit value attained from this process must be converted to human-readable temperature values as detailed in Section 4.2.2.

4.2.2 Temperature Conversion

The following formulas can be used to convert the temp code to standard temperature units:

$$\text{Temperature (°C)} = \frac{175.72 * Temp_{Code}}{65536} - 46.85$$

$$\text{Temperature (°F)} = \left(\left(\frac{175.72 * Temp_{Code}}{65536} - 46.85\right) * 1.8\right) - 32$$

These conversions and the I2C read required to attain a temperature value are handled within “getTempMeasureC” and “getTempMeasureF”. Using “startTempMeasure” followed by a 20ms delay and then “startTempMeasureX” will yield the temperature value as a float.

5 Lighting

To demonstrate the potency of the BGX13 module for integration with existing systems, a lighting demo is included on the SLASH1000A. The lighting portion of the shield is a separate circuit capable of being operated independently of the core or sensing subsystems. The system is comprised of a PCA9633 I2C LED driver, 4 0-10 LED control Circuits, and 4 colors of onboard LED for visual feedback. The LED driver controls all elements of this subsystem.

The lighting subsystem is designed to work with voltages between 3.3 and 5 Volts, and as such, operation may require modifications for optimal results. These modifications are detailed in Section 5.1.3. Any functions referenced are found in the PCA9633 Class of the ArrowBGXShield Library.

5.1 Configuration

The lighting section is based on the I2C controlled PCA9633. The PCA9633 is interfaced directly with the host processor and can be used without the BGX13. Likewise, operation of the Sensing and Core systems is possible without this unit. The PWM LED driver is highly configurable and more details are available in the datasheet. In the interest of safe and simple operation, all initialization is handled in the “init” function and additional settings are managed with additional pre-defined functions. The PCA9633 is connected to onboard LEDs and to a Sinking LED driver system. These configurations can be enabled through the “onBoardMode” and “outputMode” functions for onboard LEDs and external LEDs respectively.

5.1.1 Onboard Lights

Five onboard LEDs are provided to show the state of the RGBW (Red, Green, Blue, White) channels. The green channel features two LEDs to ensure equal light output is possible on all channels. The green LEDs will not function at voltages below 4V without modification as per Section 5.1.3. The onboard LEDs can be selected as the preferred output by calling “onBoardMode”. These lights can then be controlled by calling “rgbwControl” with an 8-bit integer parameter for each of the channels. The 255 divided by the 8-bit parameter corresponds to the inverse duty cycle of the PWM output from the driver. This inverse duty cycle is desired as the LED driver is sinking for the onboard LEDs. Please note that each LED has a different luminous intensity vs. forward current curve and as such responses and light outputs may differ for the same value.

5.1.2 0-10V Output

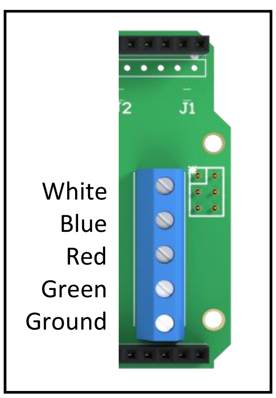


Figure 2 - 0-10V Outputs

To allow the SLASH1000A to interface with existing lighting units, an industry standard 0-10V driver was included on the board. Four channels of this driver are available. The PCA9633 still operates as the core of the system and the 0-10V output can be selected by calling “outputMode”. This function call will properly configure the driver for use with a 5V system. The PCA9633 is operated as a source under this configuration; the resultant PWM waveform is time averaged by an analog amplifier. The resultant signal drives a NPN transistor to create the desired control signal. Operation at voltages other than 5V will require modification as per Section 5.1.3. The output signals are connected to the onboard screw block header to allow wires to be easily connected and removed. The outputs are arranged as shown in Figure 2. Please ensure ground is always the first connection made when connecting any lighting driver. The ERP PSB30W-0700-42 is the recommended driver.

5.1.3 3.3V Operation

The onboard drivers are designed to work at 5V. Operation is possible at 3.3V with a few simple modifications. First, D8 (the second green LED) must be removed and replaced with a jumper. RC0402FR-070RL is the recommend part. Further, R16, R18, R30, and R32 must be removed and replaced with 100K Ohm 0402 resistors. This modification will alter the gain of the amplifier driving the 0-10V output to ensure the full output range is possible. CRCW0402100KJNEDC is the recommended part. The location of all parts can be found in the circuit schematic.

6 Other Circuitry

The SLASH1000A features a host of supporting circuitry to facilitate its operation. The circuits are intended to enable the core systems. As such, these components do not require attention or interaction for standard operation. Details are included here for reference purposes and for advanced users. Modifications to any part of the board may affect functionality.

6.1 Logic Level Translation

In order to allow the Core, Sensing, and Lighting systems to work with host processors running at different voltages, logic level translation is included on the board. There are two logic level translation circuits used; one for general usage and another intended for bi-directional I2C translation.

The primary logic level translation is driven by the ADG3308 8-Channel bi-directional translator. This translator allows for bi-directional communication for lines not tied high or low on both sides. This system is used for GPIOs and UART translation.

In order to facilitate I2C communication, an alternative translator is needed in order to facilitate the pull-up resistors and bidirectional nature. A NPN transistor-based translator is used. More details on the circuit used are available in AN10441 from NXP.

6.2 Power

The SLASH1000A systems are powered in two groups. In order to ensure smooth, reliable 3.3V power is delivered to the SLASH1000A core and sensor systems, a LDO regulator is included. The Lighting system is designed to work off the host processor power rail, at up to 5V, but at as little as 1.8V.

6.3 Future Options

Several features are designed into the SLASH1000A to allow for future additions to the feature set. This document will be updated as these features become available. For questions on features not listed here, or on advanced operation, please contact esc@arrow.com with the subject ‘SLASH1000A’.

7 Additional Resources

The following resources are provided under the MIT License to ease development and provide a starting point.

7.1 Library Links and Demos

All libraries and demos can be found at:

https://github.com/ArrowElectronicsESC/BluetoothXpressShield

7.2 Docs.Silabs

Details on the BGX13 Modules can be found at:

https://docs.silabs.com/gecko-os/1/bgx/latest/

7.3 BOM/Schematic/Assembly

The BOM, Schematic, and assembly documents are available here:

https://github.com/ArrowElectronicsESC/BluetoothXpressShield

8 Acknowledgements

These materials, software, and products were produced by Arrow Electronics. All elements were created by Eric Beppler, Andrew Reisdorph, and Peter Foy. Questions and comments should be directed to esc@arrow.com with the subject line “SLASH1000A”.

A special thanks to Silicon Labs for their support, innovation, and partnership.

9 Change List

Version	Date	Author(s)	Changes
Rev 0.0	12/17/2018	Eric Beppler	Document Creation
Rev 1.0	01/29/2019	Eric Beppler	Initial Release