

1 Introduction

This document describes the design features of the Bug Labs von Hippel Module. It contains sections describing all the electrical interfaces and specifying electrical requirements on each interface including power requirements and constraints. This document is intended to be the source of information for the von Hippel Module user documentation.

The von Hippel Module is a special-purpose Bug Labs module, intended to be used for module hardware or software development. It exposes all the standardized interfaces in an industry-standard 0.100" (2.54mm) pitch header. This header allows a user to connect the Bug Base module interface to external hardware, thus providing the Bug-specific hardware and allowing experiments with a breadboard or other evaluation hardware. It is called the *User Connector* in this document.

Warning: The von Hippel module, the Base Unit, and externally connected user circuitry may be permanently damaged if the von Hippel module is used incorrectly. Power and direct CPU interfaces are exposed, and therefore care must be taken when connecting to *and programming* devices on the von Hippel Module. This document contains descriptions of what interfaces are sensitive, what protection is built into the module, and what risks are involved. But ultimately the user must take responsibility for heeding all warnings and adhering to all requirements. In a word, a user must understand deep details of the hardware in order to ensure proper operation of the von Hippel module. In general, users must avoid applying high voltages to any of the pins on the user connector; some pins are connected to the Base Unit CPU when the von Hippel Module is installed. While the von Hippel Module is not capable of applying high voltages to external circuitry, it can apply up to 5V, which may be high enough to damage some low-voltage devices in a user's circuit.

This document assumes that the reader is familiar with basic Bug architecture and the Base Unit. The intent of this document is to provide electrical specifications, as well as a software reference, for all software accessible resources on each module. Section 2 contains a high-level description of the von Hippel Module. Section 3 describes the features common among Bug Modules, with details about the functions on this particular module. In Section 4, specific devices are identified, with information about how the device is connected on the board and to the Base Unit, including requirements and limitations where appropriate. I did not attempt to duplicate information contained in device data sheets, and of course the data sheet is the final word on device operation. This is intended to be the information

necessary to support the software interface to the board. Section 5 summarizes the resources on the board, including bus addresses and interrupt behavior.

2 Module Description

The von Hippel Module is intended to be used for module or software development. It exposes all the standardized interfaces in an array of contact points, allowing users to connect new designs, or to connect existing devices with standardized interfaces. Two power supplies and a set of general-purpose Analog Input/Output (I/O) pins are also provided in these contacts. In addition, the von Hippel module provides a standard USB type A connector with power.

Note that the high-speed dedicated serial ports, one for the camera and one for the LCD display, are not exposed on the von Hippel module. These two interfaces are special-purpose and very high-performance, and thus do not lend themselves to breadboarding in this environment. However, many types of camera are available with a standard USB interface; these may be connected directly to the USB port on the von Hippel module.

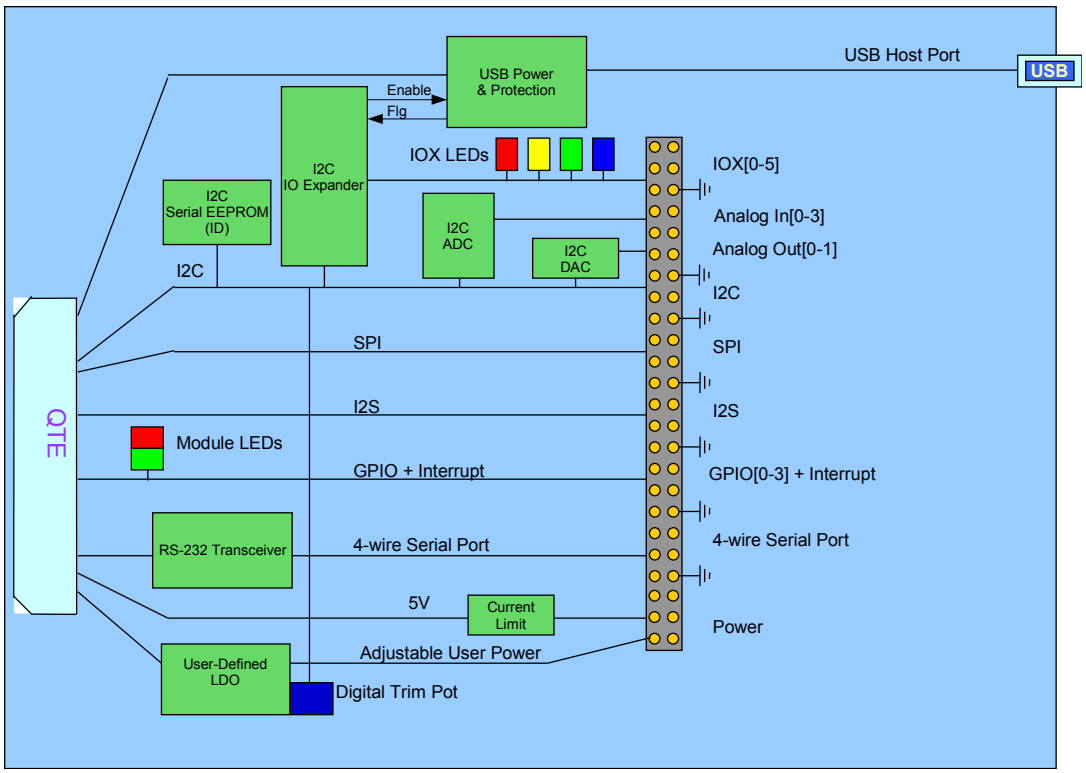


Figure 1. von Hippel Module Block Diagram



3 Common Module Design Features

The following subsections describe all Bug Base interfaces that are used on the von Hippel Module. Interfaces that are not listed here (e.g. LCD) are not used on the module.

3.1 Module Connector and Mechanical Design

The von Hippel Module PCB is 60.23mm square, with beveled corners. The module uses the standard Bug Labs Base Unit connector, the high-density Samtec QTE connector with 40 positions total. The location of the connector is mechanically fixed, and the pinout is defined in Table 1. Unshaded table entries are available to the User; entries with shaded backgrounds are not.

Signal	Netname	Pin	Pin	Netname	Signal
Ext Battery Power to Base	BAT_EXT_P1	01	02	V5_P2	5V Power to Module
Ext Battery Power to Base	BAT_EXT_P3	03	04	SPI_MISO	SPI: Master-In Slave-Out Data
Module Present	MOD_PRSNT_N*	05	06	SPI_MOSI	SPI: Master-Out Slave-In Data
LCD Interface MD0	MPL_MD0	07	08	SPI_SS	SPI: Select
LCD Interface MD1	MPL_MD1	09	10	SPI_CLK	SPI: Clock
LCD Interface Clock	MPL_MC	11	12	SPI_RDY	SPI: Ready
	GND	13	14	GND	
Serial Port: Receive Data	RxD	15	16	I2C_SCL	I^2C: Clock
Serial Port: Transmit Data	TxD	17	18	I2C_SDA	I^2C: Data
Serial Port: Clear to Send	CTS	19	20	GPIO3	GPIO Bit 3 (MSB)
Serial Port: Ready to Send	RTS	21	22	MOD_INT_N*	Module Interrupt, active LOW
GPIO Bit 0 (LSB)	GPIO0	23	24	GPIO2	GPIO Bit 2
I2S Clock	SCK	25	26	LCD_VSYNC	LCD Frame Signal
I2S Frame	SFS	27	28	GPIO1	GPIO Bit 1
I2S Transmit Data	STxD	29	30	GND	
I2S Receive Data	SRxD	31	32	CSI_MCLK	Camera: Clock
	GND	33	34	GND	
USB: Data +	USBDP	35	36	CSI_RI_P	Camera: Data In +
USB: Data -	USBDN*	37	38	CSI_RI_N*	Camera: Data In -
5V Power to Module	V5_P39	39	40	MOD_CHRG_V5	Battery Charge to Module

Table 1: von Hippel Samtec Connector Pinout



The module has the typical Bug Labs bi-color Red/Green LED indicator to represent module status. This LED is controlled entirely by software on the Base Unit, not automatically by any hardware. The LED is powered from the module power, so the LED cannot light either color when the module is powered off.

3.2 Internal Module Power

The module gets 5V input power from the Base Unit. The maximum current draw for a module is 500mA, for a maximum power dissipation of 2.5W for all loads, *including* any user loads connected through the header.

There is a local low-power LDO to develop 3.3V power for a few devices on the von Hippel Module. The remaining devices on the Module are powered directly from the 5V input power. Neither of these internal power buses are available for user applications.

Power available to the User is discussed below.

3.3 I²C Bus

The von Hippel Module contains five devices on the I²C bus, as listed in Table 2: the EEPROM, an I/O expander, a 16-bit ADC, an 8-bit DAC, and a digital potentiometer. As with most modules, the EEPROM is programmed in the Bug Labs factory with module-specific info that is used by the Base Unit driver software to identify the module type (von Hippel Module in this case) and initialize it. The contents of this EEPROM must not be modified by the User.

I²C Device	Device Addr	Notes
AT24C02 EEPROM	1010 000r : 0xA0	size=256B
PCA9538 I/O Expander	1110 001r : 0xE2	8-bit I/O + Interrupt
LTC2487 16-bit ADC	0100 100r : 0x48	8-bit I/O
MAX4820M 8-bit DAC	1011 000r : 0xB0	8-bit I/O
AD5259 Digital Potentiometer	0011 000r : 0x30	8-bit I/O

Table 2. Von Hippel Module – I²C Bus Devices

The NXP I/O expander (abbreviated IOX) provides additional GPIO control/status signals beyond the



four module GPIO signals from the Base Unit. The IOX port assignments are described below. The device also supports an Interrupt output to the Base Unit, derived from programmed inputs to the IOX.

*3.4 **GPIO***

There are four GPIO signals from the Base Unit, numbered GPIO0–GPIO3. All four GPIO signals connect to the User Connector. Two of these GPIO lines are also connected to the module bicolor LED. They may be used externally, but recognize that the state of the GPIO line will be reflected in the module LED.

Note that these GPIO lines connect directly to the Base Unit CPU. Signals on these pins should be 3.3V logic levels, and under no circumstances should voltages above 3.3V be applied to these pins.

GPIO3 defaults to the control lead for the module Red LED, active low=ON, high=OFF. GPIO2 defaults to the control lead for the module Green LED, active low=ON, high=OFF. Each lead is connected directly to its LED cathode, and so the on/off state of the LED will reflect exactly the state of the GPIO line; and the LED loads the GPIO line. Each LED anode is connected to module power, so when the module is not powered the LEDs cannot light.

By convention, when the module is first powered, the software driver on the Base Unit initializes the Red LED to ON, and the Green LED to OFF. As the driver detects and initializes the board, it should turn off the Red LED and blink the Green LED. After the module has been successfully initialized, to preserve power, software should either turn off the Green LED or blink it periodically with a very low duty cycle. Note that if a GPIO net is floating (hi-Z; i.e. set as an input) on the Base Unit, the LED will not light. Note that these leads do *not* have PU or PD resistors on the Module; they will be OFF unless the Base Unit programs GPIO3 and GPIO2 as outputs and sets them to 0.

GPIO	Netname	I/O	Init	Name/Function
GPIO3	GPIO3_REDLED	O or User def.	0	RLED Control & User Connector L=REDLED ON; H=REDLED OFF
GPIO2	GPIO2_GRNLED	O or User def.	1	GRNLED Control & User connector L=GRNLED ON; H=GRNLEN OFF
GPIO1	GPIO1	I/O	Input	User connector
GPIO0	GPIO0	I/O	Input	User connector

Table 3. Von Hippel Module – GPIO Assignments



Once the driver initialization is complete, the GPIO lines may be used as true General Purpose I/O lines to or from the User Connector, including the LED connected lines. If GPIO2 or GPIO3 are used by external circuitry, the respective LED will light when they are at a logic low level, as long as the logic driver can sink at least approximately 10mA of current.

The von Hippel Module does not have a module hardware reset function.

3.5 ***SPI Bus***

The Base Unit CPU has a single SPI interface, which is fanned out and connected to all four module connectors. Each of the module connectors has a unique SPI_SS (SPI Slave Select) signal, but the other signals in the SPI bus are shared. The SPI_MISO (Master In Slave Out) serial data line from the module to the Base Unit therefore must be isolated on each module, so that they don't collide on the shared Base Unit MISO net. That is, there must be a buffer on any module using the SPI interface, and the von Hippel module is no exception. There is a tri-state buffer enabled by SPI_SS.

The problem is, the Base Unit CPU supports either an active high or an active low SPI_SS. However, the tri-state buffer requires an active low SPI_SS (SPI_SS = L = enable; SPI_SS = H = disable). As a result, the von Hippel module *requires* that SPI_SS be programmed as an active low select signal. If a user wishes to connect an SPI device needing an active high SPI_SS, the user *must* include an inverter on the SPI_SS signal from the von Hippel module to the user circuitry in order for this SPI bus to work properly.

Because the SPI bus connects directly to the Base Unit CPU, great care must be taken to avoid applying high voltages to any SPI signal. The CPU is expecting 2.8V-3.3V logic levels on these nets. If the user requires other SPI voltages, level translators *must* be included in the user circuitry. **Under no circumstances should voltages above 3.3V be applied to any SPI nets; doing so could damage the Base Unit.**

3.6 ***Interrupt***

The module interrupt line MOD_INT* to the Base Unit is a wire-OR of two sources. First, the INT output of the I/O expander is connected to MOD_INT* through a series resistor. Any IOX I/O pin that



is programmed as an input may be used as a source for this interrupt. Any such programmed inputs will drive the INT* output low if it changes state from the original value when the IOX is initialized. The user is responsible for initializing these I/O pins, and responding to any resulting interrupts. The general purpose user-defined I/O in the I/O Expanders (IOX[0-5]) will default to inputs unless programmed otherwise, and thus will cause the INT* line to be asserted whenever they change state. The USB_FLG_N* is a dedicated input from the power controller, and will cause an interrupt if properly initialized. All IOX nets are currently pulled down, so they will not cause spurious interrupts if unused and unconnected. The IOX INT output is an open-drain source; thus it is high-impedance when inactive.

The second source of interrupts is the EXT_INT* pin on the User Connector, which is also connected through a series resistor to the MOD_INT* net. Thus, any external device may generate an input by pulling this line low. The user should ensure that any external interrupt source is also open drain or open collector, so that it is high impedance when inactive. If this is not so, then there may be contention on the MOD_INT* line, forcing the line to an indeterminate state and causing intermittent or spurious interrupts to the Base Unit CPU. In this scenario, the series resistors will prevent any circuit damage, but high currents and unpredictable behavior may result. There is a pullup resistor on the Base unit for this line; a pullup should *not* be used on any external interrupt circuit.

3.7 ***I²S Bus***

Digital audio data may be transferred in both directions between the Base Unit CPU and external circuitry via the I²S bus, which is connected directly to the User Connector. The bus is not used by any devices on the von Hippel Module itself.

Note that the Base Unit has a single I²S bus that is multiplexed to the four module slots. Because of this, care must be taken to control the receive data (RxD, from von Hippel Module in to the Base Unit) from the four slots if multiple I²S bus devices are installed. That is, if a second von Hippel Module, or an Audio Module, is plugged in simultaneously, care must be taken in software to prevent input from the two devices from colliding at the Base Unit.

3.8 ***USB***

The von Hippel Module provides a USB type A connector, with protection and power control. The



USB data lines are connected through a transceiver to the Base Unit. The connector is wired as a host. Power to the connector is switched by a Maxim USB power device, enabled (power on) by setting IOX7 high. The active low Fault indicator from the power switch is connected to IOX6.

3.9 Serial Port

Serial data may be transferred in both directions between the Base Unit CPU and external circuitry via the serial port. The serial port from the Base CPU is connected through an RS232 level translator to the User Connector. The serial data is not used by any devices on the von Hippel Module itself.

4 Von Hippel Module Devices

4.1 Connector Array

The von Hippel Module has a footprint for a 2x22 surface-mount 100-mil pitch header or receptacle. Currently boards are being assembled with receptacles. Each pin of the receptacle is labeled with an abbreviated name. Next to each silkscreen label is a hole large enough to accept a 22AWG wire for soldering for a permanent connection if desired.

The connector is arranged in a 2x22 array, with each interface grouped, and each functional group containing at least one ground pin. The pinout is described in Table 4 following. In addition, the (abbreviated) pin names as listed in the table are silkscreened directly onto the PCB for reference.

Function	Net	Pin	Pin	Net	Description
Power	VADJ	1	2	GND	Adjustable Voltage output with GND
	5V	3	4	GND	5V Power bus
DAC	GND	5	6	DACA	Digital-to-Analog converter outputs A & B
	DACB	7	8	GND	
ADC	ADC0	9	10	ADC1	Analog-to-Digital converter inputs 0-3
	ADC2	11	12	ADC3	
I^2C	GND	13	14	SCL	I^2C Bus clock and bi-directional data
	SDA	15	16	GND	
IOX	IOX0	17	18	IOX1	I/O Expander port, input/outputs 0-5 (Note: IOX0-3 are connected to LEDs)
	IOX2	19	20	IOX3	
	IOX4	21	22	IOX5	



Function	Net	Pin	Pin	Net	Description
GPIO	GND	23	24	INT_N*	Interrupt (active low) to Base CPU
	GPIO0	25	26	GPIO1	General-purpose I/O bits 0-3 (Note: Module red & green LEDs are connected to GPIO3 & GPIO2)
	GPIO2 (Grn)	27	28	GPIO3 (Red)	
SPI	GND	29	30	CLK	SPI Interface. Base is master, module is slave.
	RDY	31	32	MOSI	
	SS	33	34	MISO	
I2S	GND	35	36	SCK	Digital audio interface
	STxD	37	38	SRxD	
	SFS	39	40	GND	
Serial Port	RxD	41	42	TxD	Four-wire RS232 serial port interface
	RTS	43	44	CTS	

Table 4: von Hippel Module Pin Field Groups and Pin Assignments

4.2 A/D Converter

The Analog-to-Digital Converter (ADC) device is a Linear Technologies LTC2487. It is a 16-bit ADC, with a maximum conversion speed of 15 samples-per second using its internal oscillator. The device also contains a temperature sensor, and programmable gain control. The analog inputs may be operated as four single-ended inputs or as two pairs of differential inputs. The analog input range is GND to 5V. It is suitable for sensor inputs

The device is controlled and accessed over the I^2C bus. It is assigned an I^2C address of 0x48 by default, but this address may be changed by installing resistor straps if desired.

See the device data sheet for detailed specifications.

4.3 D/A Converter

The Digital-to-Analog Converter (DAC) is a Maxim/Dallas Semiconductor MAX5820M device. This is an 8-bit DAC with two analog outputs with programmable attenuation. The analog output range is GND to 5V. The maximum data rate is determined by the I^2C interface. Normally, this interface runs with a 400kHz bit-clock, with 10 b/transfer, and a minimum of 6 transfers per conversion, which leads to a maximum data rate of roughly 6k samples per second.



The device is controlled and accessed over the I^2C bus. It is assigned an I^2C address of 0xB0 by default, but this address may be changed to 0xB1 by moving a resistor strap if desired.

See the device data sheet for detailed specifications.

4.4 ***USB Interface***

The USB interface is connected to a standard type A USB connector. Power for this connector is controlled by a Maxim MAX1558 device with two control leads connected to the I/O Expander (for IOX pin assignments, see Table 4 below). The 5V power to the USB connector is enabled by the USB_VEN signal. The default state for the power controller is controlled by the driver initialization, which is currently initialized to ON. The USB power controller provides a Fault indicator, USB_FLG_N*, which can be read through the I/O Expander. The controller will pull this lead low if *either* the USB or the user 5V supply has an output fault. The two fault nets are wire-ORed, so software cannot read which caused the fault; however, it should be an easy matter for the user to determine the source.

See the section on user power below for an important discussion of current limits.

See the device data sheet for detailed specifications.

4.5 ***RS232 Serial Port***

The von Hippel Module includes a four-wire serial port, with standard RS232 drivers/receivers. The four serial port signals are present in the pin field, and may be brought out to a standard DB9 for connection to a standard serial port on a PC or any other device. For a standard connection, wire the serial port pins to the DB9 as follows:

Signal Name	DB9 Pin	User Pin	Notes
TxD	2	42	Tx Data out from Bug
RxD	3	41	Rx Data in to Bug
CTS	8	44	Clear to Send in to Bug
RTS	7	43	Request-to-Send out from Bug

Table 5. von Hippel Module – Serial Port Standard DB9 Pinout

The driver/receiver device is a Maxim MAX3223E device, which translates the signals to/from the Base into nominal ±5V driver/receivers. The device is constantly enabled with a pulldown. (This can be changed to a permanent disable by removing the pulldown and installing a pullup; see schematic.) Note that the TxD and RxD directions are defined in Table 5 above; be sure that the data direction is correct to any user endpoint.

4.6 I²C I/O Expander

The NXP PCA9538 I/O Expander device provides additional GPIO control/status signals beyond the four GPIO signals from the Base Unit. Table 6 identifies these expanded IO signals and their functions.

Exp. I/O	Signal	Dir	Def	Function
IOX0	IOX0	I/O	–	General purpose I/O to pin field; user defined; red LED
IOX1	IOX1	I/O	–	General purpose I/O to pin field; user defined; yellow LED
IOX2	IOX2	I/O	–	General purpose I/O to pin field; user defined; green LED
IOX3	IOX3	I/O	–	General purpose I/O to pin field; user defined; blue LED
IOX4	IOX4	I/O	–	General purpose I/O to pin field; user defined
IOX5	IOX5	I/O	–	General purpose I/O to pin field; user defined
IOX6	USB_VEN	Out	L	Enable USB power; H=power ON; L=power OFF
IOX7	USB_FLG_N*	In	H	Fault status from USB power controller; H=normal operation; L=Fault

Table 6. von Hippel Module – I²C I/O Expansion

Each of the 8 I/O pins must be individually programmed in the Configuration Register as an input or output. The Polarity Register can change the polarity of any input pin individually. The I/O Expander device also contains logic to generate an open-emitter interrupt signal. See the INT* section below for details of operation. Following a power-on, all pins on the device default to inputs. IOX0-IOX6 have pulldowns that cause these lines to default to a low level.

The USB power controller must be enabled for the USB port to work. It is disabled following power-on. To enable, bit 6 must be programmed as an output, and then set H. The fault status of the USB power controller is available on bit 7. The conditions that assert the fault signal are described in the data sheet for the Maxim MAX1558 controller.



IOX0 through IOX5 are general-purpose user-defined lines connected to the User Connector. Of these, IOX0-IOX3 are connected through inverters to LEDs equipped on the von Hippel module. Users who wish to reduce power to a minimum may wish to depopulate the LEDs. These LEDs are driven by an inverter to minimize loading, so they may be used as indicators on inputs, outputs, or bidirectional I/O lines.

4.7 ***I²C Digital Potentiometer***

The Analog Devices AD5259 digital potentiometer device provides software programmability for the user LDO. The part contains EEPROM so that the programmed voltage is non-volatile and will be retained even when the Bug is turned off.

The voltage output of the adjustable LDO is set by means of a resistor divider; the digital pot is one of the resistors. The circuit is designed so that any setting of the digital pot will produce a valid output (i.e. no circuitry on the von Hippel Module can be damaged by faulty programming of the digital pot), but of course the user is responsible for setting the correct resistance value to produce the desired voltage output. Setting an incorrect value *may* damage external user circuitry, so care must be taken when programming the digital potentiometer. Users should follow good practice and set the resistance value and test the output ***before*** connecting any external user circuitry.

The device is controlled and accessed over the I²C bus. It is assigned an I²C address of 0x30 by default, but this address may be changed to 0x34, 0x98, or 0x9c by moving one or two resistor straps if desired.

The operation of the digital pot is conceptually simple. The part is connected as in “rheostat mode”, with terminal A connected to the LDO output, and terminals B and W connected together and to the ADJ pin of the LDO. It is "Rupper" in the equation below. There is a 10kΩ resistor connected as "Rlower", from the ADJ pin to ground to complete the voltage divider. The digital pot part has a total resistance of 50kΩ, and is shipped from the vendor set to mid-scale or 25kΩ.

The equation for the LDO is:
Vout = 1.25V * (1 + Rupper / Rlower)



where Rupper is the digital pot, and Rlower is the 10kΩ resistor. Thus, with the factory setting of 25kΩ, the resulting Vout is 1.25V * (1 + 25k/10k) = 4.375V. To set the LDO for 3.3V, Rupper should be set to 16.5kΩ. At the time this is written, the part is being programmed in the Bug Labs assembly factory to 16.5KΩ to produce this 3.3V output value. However, this procedure may change, or your module may have been used previously, so be sure to verify the output voltage before connecting any user equipment. The datasheet for the AD5259 contains complete instructions for programming the digital pot, and they will not be duplicated here. Table 7 lists some common user voltages and their corresponding digital pot setting.

See the section below for an important discussion of power limitations.

Vout	Rupper	Rlower
1.25V (min)	0	10K
1.8V	4.4K	10K
2.5V	10K	10K
3.3V	16.5K	10K
4.375V	25K	10K
4.75V (max)	≥ 30K	10K

Table 7: von Hippel Module: Digital Pot Settings for Common Power Voltages

4.8 ***User (External) Power***

There are two power buses on the user connector available for external use. The MAX1558 controller supplies a current-limited 5V output. It is a dual controller: the user 5V supply is assigned to port A on the controller; the USB connector is assigned to port B on the controller. The user 5V supply is connected across pins 3 and 4 of the user connector. Note the polarity of the supply voltages, as listed in Table 6. The MAX1558 device limits the output to a maximum of 500mA.

There is an adjustable LDO, a National LP38693SD, on the von Hippel module which is connected across pins 1 and 2 of the user connector. As described in the previous section, this LDO output may be programmed by software to any voltage between the LDO minimum of 1.25V to a maximum of the module input minus the maximum dropout voltage (which varies depending with Vi, Vo, and the current load, but roughly speaking max output is 4.75V). The LDO is limited to a maximum of 500mA

output current.

As we have discussed, the von Hippel module has three power outputs (USB, User 5V, User Adjustable). In addition, the module itself contains circuitry that draws power, of course. Each of these power outputs has a current limit of 500mA, meaning that the power output is protected and will automatically shutdown if this limit is exceeded. However, the Bug Base Unit provides only a nominal 500mA to each module connector. Thus, there is an effective limit of approximately 500mA *total* available for the circuitry on the von Hippel module itself (the ADC, DAC, LEDs, etc.) *plus* the USB connector *plus* the User 5V *plus* the User Adjustable power supply. Unfortunately, the von Hippel module does not autonomously create power, and therefore is limited by the input from the Base Unit. These current limits are in place mainly for protection, to prevent damage to the Bug, any connected user circuitry, and the user. If the von Hippel module plus any attached circuitry exceeds this 500mA limit from the Base Unit, the supply on the Base Unit will shut down and remove power.

As mentioned earlier, if the MAX1558 detects a power fault (short to ground) on USB power or User 5V power, it will shut down and assert the FLG signal into the IOX, which can be configured to interrupt the Base CPU if desired. The User-adjustable LDO has no fault indication; if the output is shorted, the LDO will try to supply current (at a lower voltage) until it reaches thermal shutdown, at which point it will shut itself off. The user must be sure to test any external circuitry for shorts before applying power, and for correct voltage levels after applying power.

Design Engineer: R. C. Pritchard
Date: 6/30/2009