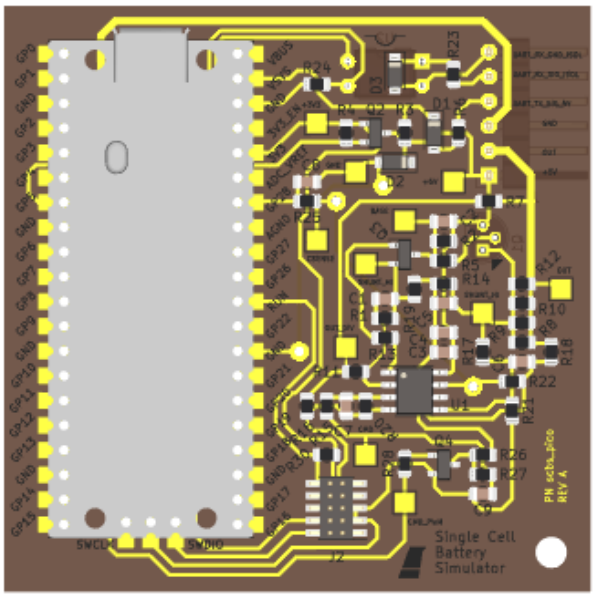
SCBS-Pico Single Cell Battery Simulator

Features

- Settable Output Voltage 2.5-4.5V ±4%
- Output Current 0-200mA (0-600mA peak)
- Output Current Sense 0-200mA ±10%
- Isolated daisy-chainable UART command interface
- Powered by a simple 5V Isolated Power Supply
- SWD programming header
- USB port for power and debug COM port



Applications

- Battery Management System (BMS) testing
- Programmable micro-scale linear power supply
- Strangely shaped coaster

Principle of Operation

The SCBS-Pico utilizes a dual rail-to-rail op-amp with a custom BJT-based output stage to provide a regulated linear power supply output between 0V and V_{supply}-0.5V, where V_{supply} is the input voltage provided to the device's 5V power input. The same op-amp is utilized for current sensing on the output via a high-side shunt resistor.

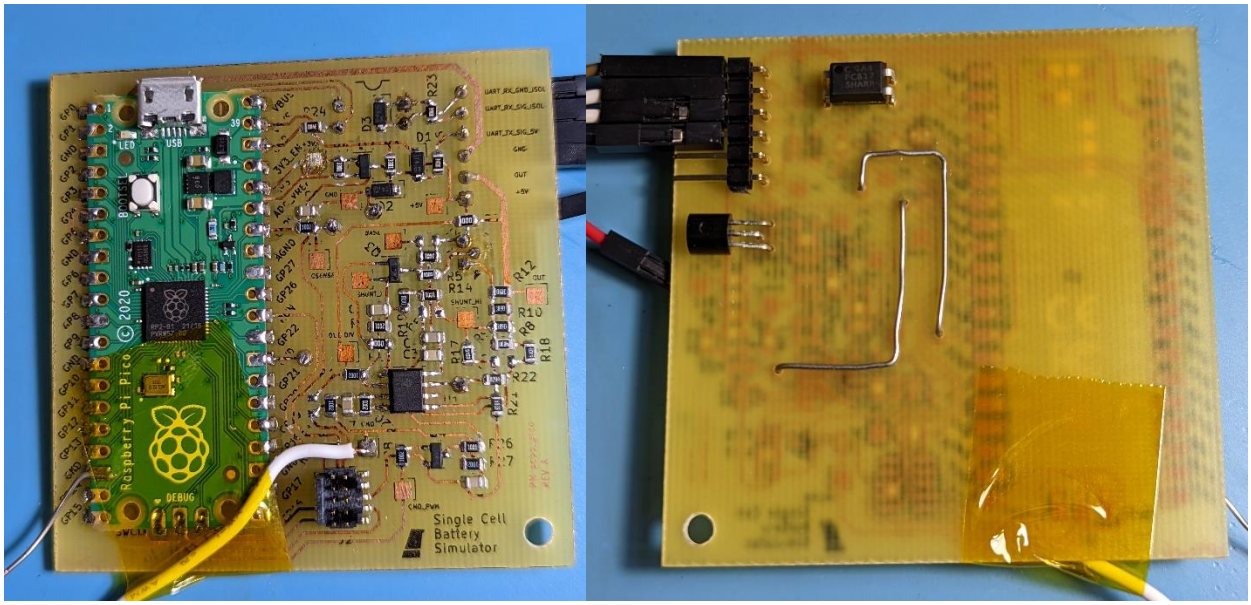
An isolated and level shifted UART interface allows serial communication with 5V voltage levels between multiple SCBS-Pico devices in a daisy-chained configuration. The UART_RX of the first device and the UART_TX of the final device should be connected to the UART master for control of the full chain. NOTE: it is recommended that the first device in the chain (containing the connection from the UART master to isolated UART_RX) have the highest V_{supply} rail, and the last device in the chain (with a level-shifted UART_TX interface) be connected to a GND that is common with the master device. Other topologies can be supported but will require additional isolation on the UART_TX of the last SBCS Pico device.

Pin Map

Pin Number	Pin Name	Function
1	+5V	DC+ power input.
2	OUT	Simulated battery positive terminal.
3	GND	DC- power input. GND reference for UART_TX_SIG_5V. Simulated battery negative terminal.
4	UART_TX_SIG_5V	5V UART output.
5	UART_RX_SIG_ISOL	Isolated 5V UART input.
6	UART_RX_GND_ISOL	Isolated 5V UART input GND reference.

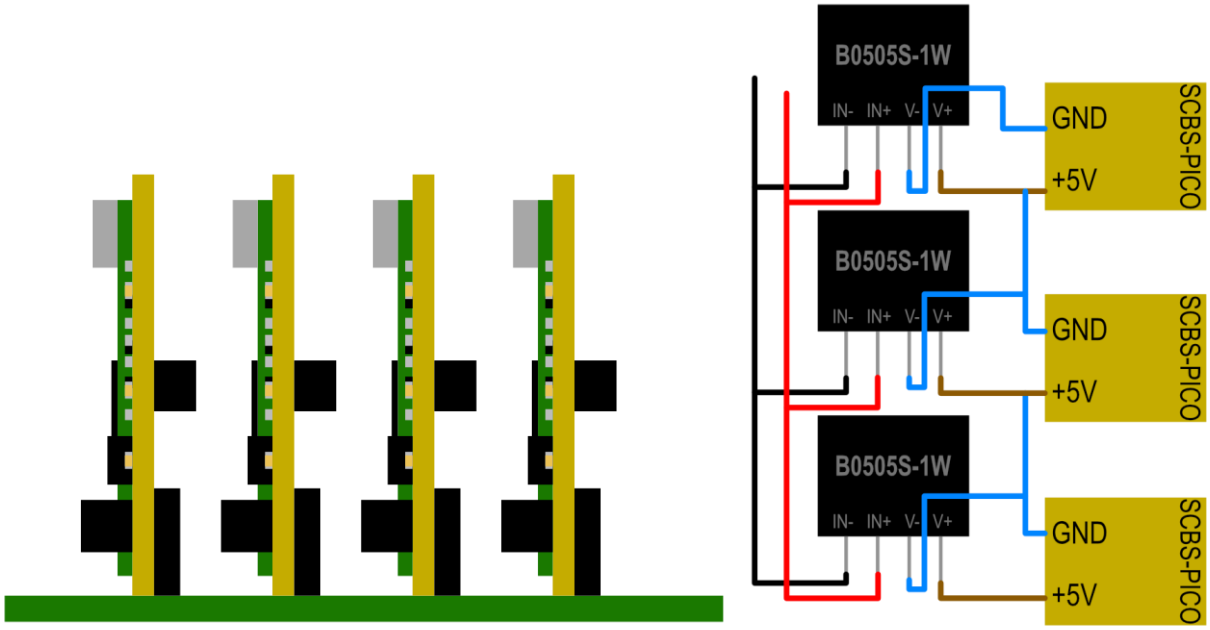
Device Form Factor

Due to time and material constraints for an initial prototype, the SCBS-Pico was constructed with a Raspberry Pi Pico 32-bit microcontroller as the communication and control interface for the analog circuit.



Left: Device front side. Temporary GND wire (silver, bottom left) and CMD wire (white, bottom middle) attached for testing of analog circuit before firmware bringup. Right: Device bottom side. Note PC817 optocoupler (DIP-4 package, top) and output 2N2907 transistor (TO-92 package, left).

The SCBS-Pico's 6-pin 0.1" pin header interface and single M3 mounting hole allow it to be mounted to a motherboard for daisy chaining into an array of devices, in order to allow simulation of a multi-cell battery pack.

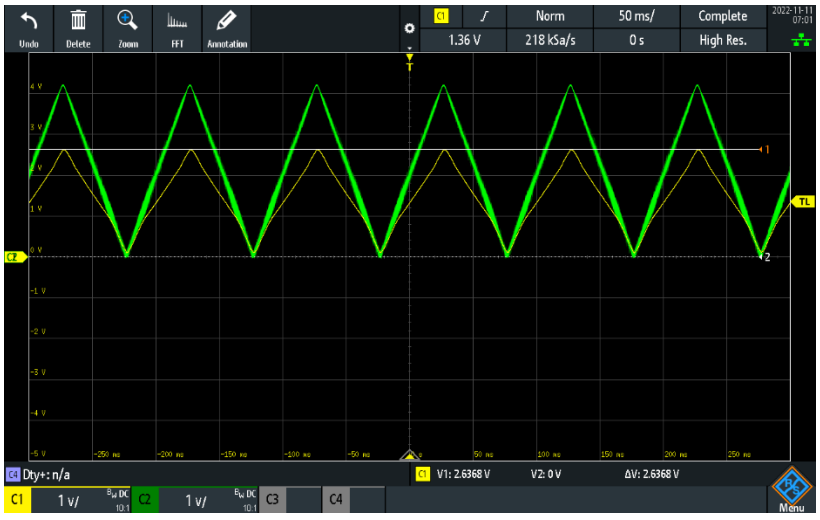


Left: Suggested daisy-chain configuration of SBS-PICO devices on a motherboard providing isolated power to each device and UART connections between each board and to the master. Right: Suggested daisy-chained power topology.

Electrical Characteristics

Parameter	Test Conditions	Min	Typ	Max	Units
V_{supply}		5		5.25	V
$V_{out,max}$	After 5 minutes of operation at 25°C, $I_{out} \leq 25mA$.	4.3	4.5		V
$V_{out,min}$			0		V
$I_{out,max}$	$V_{out} \geq 2.5V$, ambient temp 25°C.		200		mA

Output voltage percent error while tracking a 10Hz 0-4.5V square wave with a load current of 200mA on a DC electronic load. Error waveform shown in orange and yellow (bottom). Yellow waveform (CH1) is commanded output voltage, green waveform (CH2) is actual output voltage.



Current sense waveform (yellow) vs output voltage (green) into a 20Ohm load (physical resistor).

Communication Interface

The scbs_pico board provides a daisy chainable UART interface which can be used to communicate with one or more battery simulators. Each battery simulator ingests packets via its UART_RX port, and modifies / re-transmits the packet via its UART_TX port in order to control devices downstream. The UART interface is isolated on the UART_RX pin in order to enable the cells to be interconnected

Battery Simulator Command Packets

Packet Type	Packet Format
BSDIS	**Battery Simulator Discover** Discovers all battery simulator devices in the chain. Master transmits a BSDIS packet with LAST_CELL_ID=0, and each device increments LAST_CELL_ID and forwards the packet down the chain. The last device replies to the master with LAST_CELL_ID=n, where n is the number of devices in the chain. $BSCDS,<LAST_CELL_ID>*<CHECKSUM>
BSMRD	**Battery Simulator Multi Read** Reads a single register from all battery simulator devices simultaneously. Generates a single BSMRS packet reply upon success. $BSMRD,<REG_ADDR>*<CHECKSUM>
BSMWR	**Battery Simulator Multi Write** Writes a value to a single register on all battery simulator devices simultaneously. Generates a single BSMRS packet reply upon success. $BSMWR,<REG_ADDR>,<VALUE>*<CHECKSUM>
BSSRD	**Battery Simulator Single Read** Reads the value of a register on a single battery simulator, addressed via CELL_ID. Generates a single BSSRS packet reply upon success. $BSRRD,<CELL_ID>,<REG_ADDR>*<CHECKSUM>
BSSWR	**Battery Simulator Single Write** Writes a value to a register on a single battery simulator, addressed via CELL_ID. Generates a single BSSRS packet reply upon success. $BSRWR,<CELL_ID>,<REG_ADDR>,<VALUE>*<CHECKSUM>

Battery Simulator Response Packets

Packet Type	Packet Format
BSSRS	**Battery Simulator Single Response** Response from a single battery simulator (signed with CELL_ID). Includes a value that is a response to a BSSWR or BSSRD command, or an error created by another (invalid) command. Value can be “OK” (no error), “ERR:<CODE>” with an included error code, or “<VALUE>” where an actual value of a register is returned. $BSRSP,<CELL_ID>,<VALUE>*<CHECKSUM>

Revision History

Revision	Date	Author	Note
0.1.0	2022-11-10	J. McNelly	Initial release.
1.0.0	2022-12-08	J. McNelly	First draft with finalized serial protocol used in initial CELLSIM prototype.