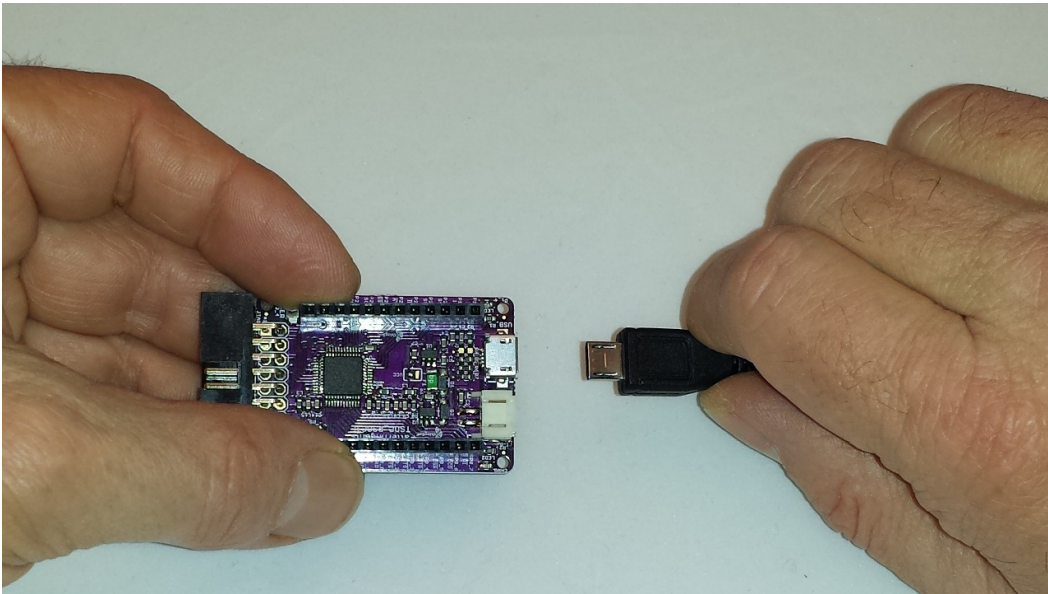


Welcome to the thingSoC® PsoC4L (Large)

The TSOC_PSoC4L is a [PSoC4-Programmable System on Chip](#) based board with integrated USB, Li-Po battery power and charging functions, in an Embedded Module format for the new thingSoC, Internet of Things open source hardware standard.

Getting Started :

- 1) Using a USB-Micro cable, plug the PSoC4L board into your Windows, MAC, or Linux PC.
- 2) The USB drivers should be found automatically on most systems.
However if they are not, they can be downloaded from :
https://github.com/thingSoC/TSOC_PSoC4L/tree/master/TSOC_PSoC4L/drivers

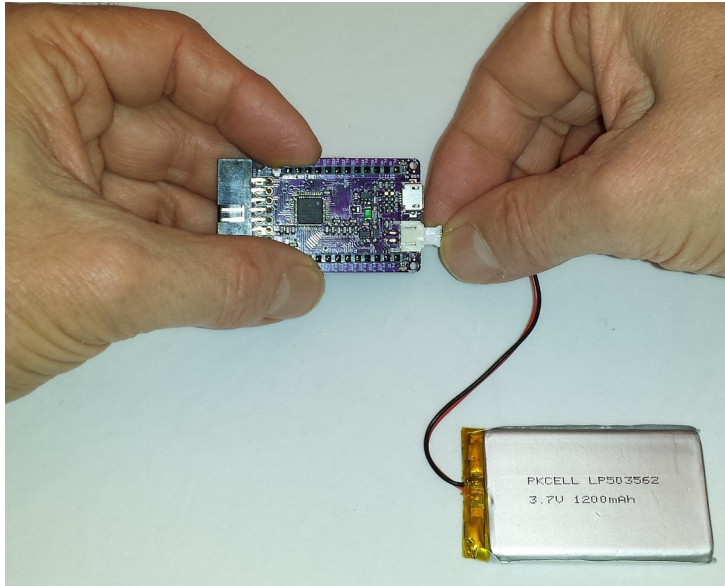


- 3) With the default firmware, when the PSoC4L board is first connected to power, it will run the USB Bootloader program for the first ten (10) seconds of operation. If a bootloader request to update the internal firmware is not received within ten (10) seconds, then the PSoC4L board will enter it's normal operating mode. In bootloader mode the PSoC4L board enumerates as a USB HID (Human Interface Device) class device.
- 4) In normal operating mode (after ten seconds) the USER LED (Green) will blink at a one (1) second blink rate; this indicates that the PSoC4L board is running the RSVP command line interpreter. In normal operating mode, the PSoC4L board will enumerate as a USB CDC (Communications Device Class) and appear to the computer as "COM" port.
- 5) Open a terminal program, like HyperTerm or TeraTerm, and connect to the COM: port that enumerated when the PSoC4L board was plugged in, the baud rate setting doesn't matter as it is a USB device.
- 6) Type "commands" and ENTER (carriage return) to get a listing of the RSVP commands.

See the appendix for a listing RSVP commands.

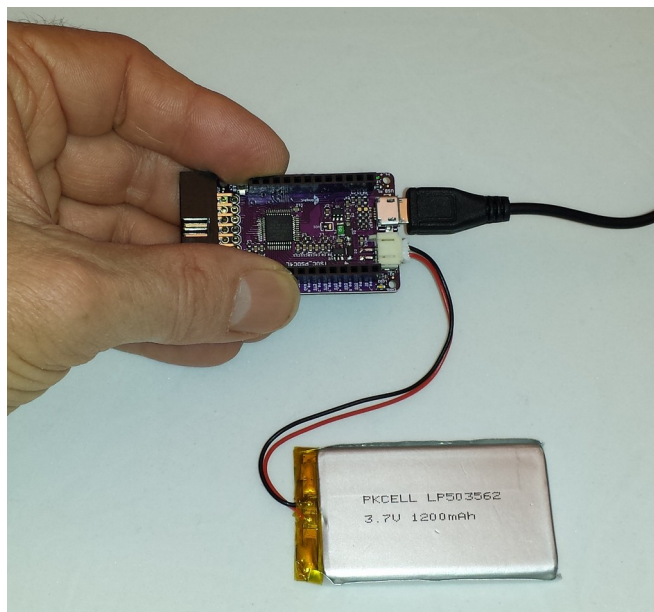
Lithium Battery Operation :

The TSOC_PSoC4L board includes a standard JST two(2) pin connector for commonly available 3.7 volt, single cell, Lithium-Ion (Li-Po) batteries. Simply plug in the battery to power the TSOC_PSoC4L as shown :



Lithium Battery Charging :

The TSOC_PSoC4L board includes a 3.7 volt, single cell, Lithium-Ion (Li-Po) battery charging circuit that is automatically enabled if USB power is connected, as shown :



The Yellow "CHRG" (Charge) LED will illuminate if the battery is charging, and it will shut off automatically once the battery is fully charged.

Safe Handling Precautions :

- 1) Always remove/unplug all power before inserting or removing peripherals.
- 2) Always ground yourself by touching a ground point before handling your boards.
- 3) Use a static safe bag when transporting your TSOC_PSoC4L board.



Adding Peripherals :

The TSOC_PSoC4L board supports both thingSoC and Mikrobus peripherals.

- 1) **Make sure to align the SQUARE/CUT edges** of the boards together to insure that the polarity is correct. Push down **evenly** to seat the boards in the socket.
- 2) Note that the "stacking pins" are somewhat longer by design, and there will be some extra space between the boards when using stacking connectors.



Reprogramming the TSOC_PSoC4L Board :

The TSOC_PSoC4L board comes pre-loaded with a USB-HID bootloader, which executes for the first ten (10) seconds after powering up the board; either using USB or battery power. This means that no additional programming dongles or hardware are needed to reprogram your board. You can also start the USB HID bootloader using the "test boot" command from the CLI.

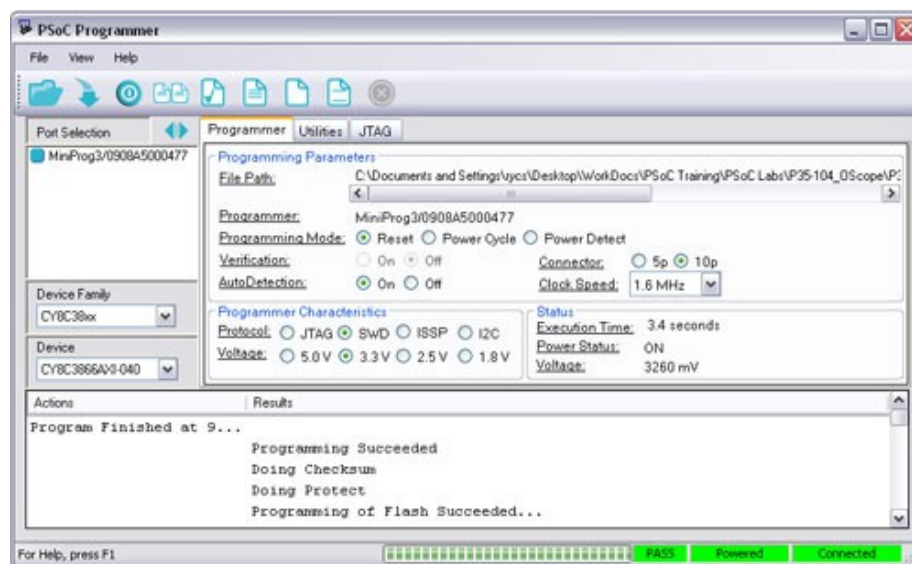
The TSOC_PSoC4L board can be programmed using the Cypress PSoC Creator IDE, which is a free download (registration required) from Cypress Semiconductor, and can be found at :

<http://www.cypress.com/products/psoc-creator-integrated-design-environment-ide>

Cypress has a number of excellent videos and instructions online for using Creator.

The TSOC_PSoC4L board can also be programmed using the Cypress PSoC Programmer tool. The PSoC Programmer tool is a free download (registration required) from Cypress Semiconductor, and can be found at :

<http://www.cypress.com/documentation/software-and-drivers/psoc-programmer-3245>



For more information, consult Cypress Semiconductor Application Note AN73503 :

<http://www.cypress.com/documentation/application-notes/an73503-psoc-usb-hid-bootloader>

"Bare Metal" Programming :

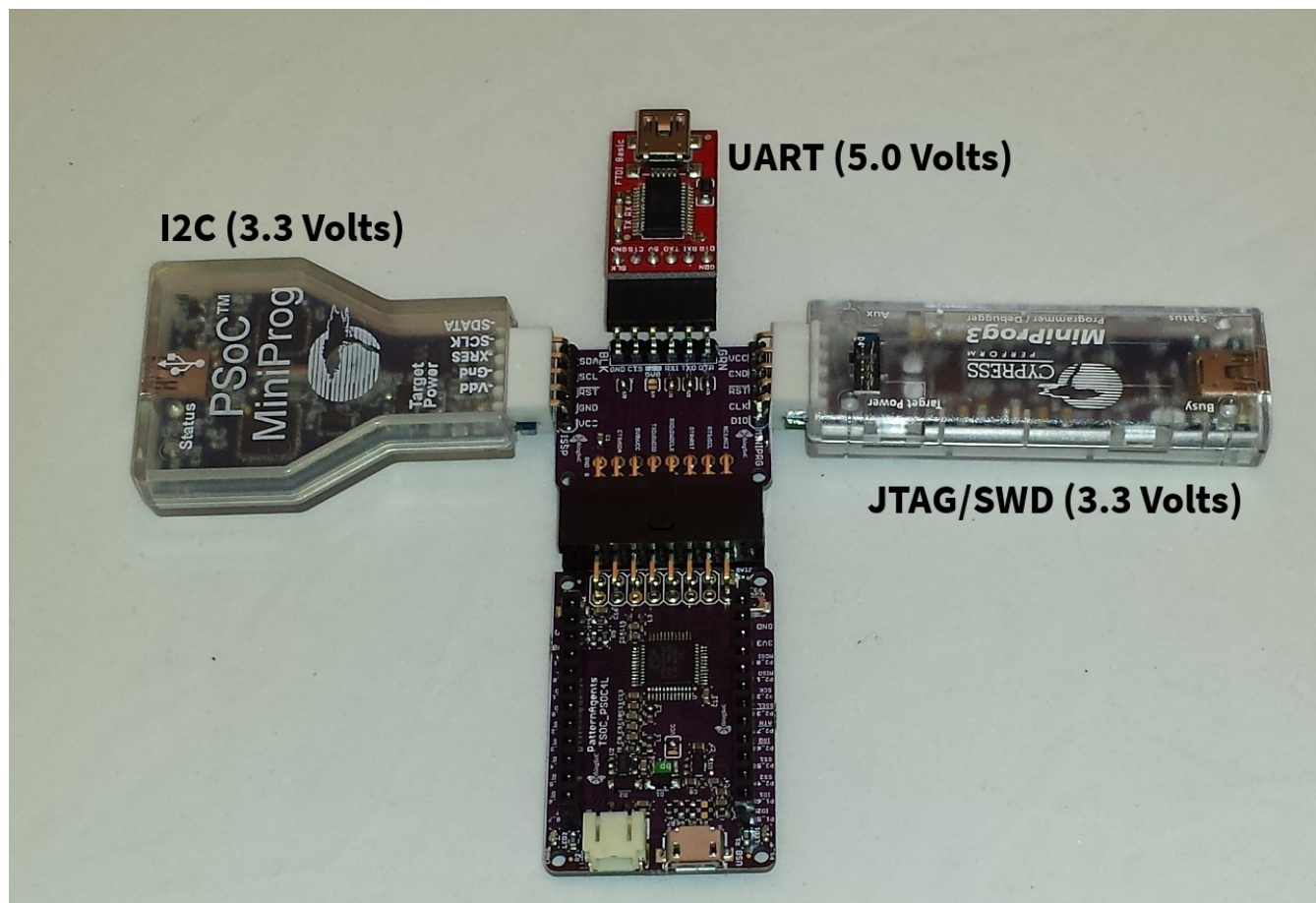
At the factory we use a Cypress MiniProg3 JTAG programmer to install the default firmware on your TSOC_PSoC4L board before it ships to you. If you would like to do bare metal programming or to use the PSoC Creator debugger, you will need a MiniProg3 and a MiniProg adapter.

You can find more information about the MiniProg3 here :

<http://www.cypress.com/documentation/development-kitsboards/cy8ckit-002-psoc-miniproprog3-program-and-debug-kit>

thingSoC MiniProg Adapter :

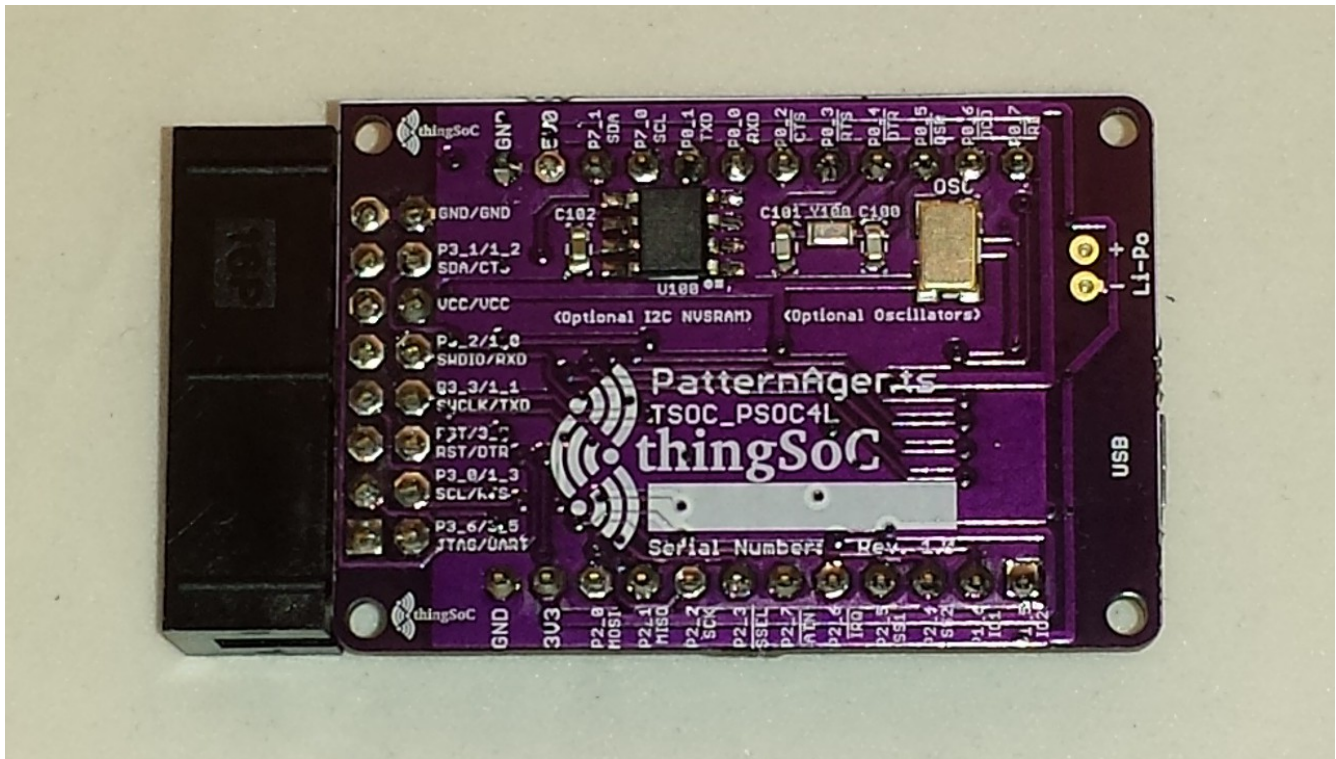
The thingSoC MiniProg adapter is used to provide several interfaces through a single common connector on the TSOC_PSoC4L board, including JTAG, UART, and I2C as shown :



The thingSoC MiniProg adapter implements the thingSoC DPC (Digital Programming Connector) and allows for programming, testing, and debugging using either a UART or I2C bootloader, or bare metal programming using JTAG/SWD. This is generally for advanced users only, and is not normally required for User programming. Most Users will simply use the USB interface for reprogramming. Note that the USB-UART (FTDI) is a 5.0 Volt ONLY connection for backward compatibility.

thingSoC PsoC4L Optional Hardware :

The TSOC_PsoC4L has footprints on the reverse side of the board that are not populated by default. These include options for a Real-Time Clock Watch Crystal (32Khz) , a second Oscillator (5x3mm), and a socket for a Cypress NVSRAM component, as shown below :



Complete documentation, including schematics, layouts, gerbers, and Bill of Materials (BOM) are available on the thingSoC website at :

https://github.com/thingSoC/TSOC_PSoC4L/tree/master/TSOC_PSoC4L

There is also a project Wiki available for asking questions and more information at :

https://github.com/thingSoC/TSOC_PSoC4L/wiki

thingSoC RSVP Command Summary :

The default firmware includes the RSVP command line interpreter, which can be accessed using any terminal emulator program (HyperTerm, TeraTerm, CuteTerm, etc.) on the USB COM: port that enumerates after the USB HID bootloader exits.

commands <CR> Display a list of the currently available commands

info <CR> Display the current firmware revision

echo <CR> Display the current command parameters

console X <CR> Move the CLI console to port X
(change the console port to a Wireless radio, for example)

connect X Y <CR> Connect Port X to Port Y (until USER Switch is pressed)
(useful for testing Radios, and serial peripherals)

test adc <CR> Display the current ADC readings for all ADC channels

test boot <CR> Enter the bootloader program to update the firmware

test led <CR> Blink all onboard LEDs

X = 0 = USB Port

X = 1 = UART1

X = 2 = UART2

X = 3 = I2C1

X = 4 = I2C2

X = 5 = SPI1

X = 6 = SPI2

(for Firmware version 1.3.0 – more commands are being added...)

thingSoC FRU (Field Replaceable Unit) Summary :

The second I2C unit (I2C2) is programmed using EZ-I2C to act as an emulated EEPROM, containing a standard IPMI (Intelligent Peripheral Management Interface) record for a FRU (Field Replaceable Unit). This data structure is used to contain information about the board serial number, date and location of manufacturing, asset number, etc.

For more information, please see the IPMI FRU Specification at :

<http://www.intel.com/content/dam/www/public/us/en/documents/product-briefs/platform-management-fru-document-rev-1-2-feb-2013.pdf>

thingSoC® PSoC4L Features :

The TSOC_PSoC4L is a low cost, embeddable module featuring a PSoC4 "Large" device :

- 32-bit MCU Subsystem (48MHZ ARM Cortex-M0)
- USB Full-Speed Controller - 16 Endpoints
- Li-Po battery connector and integrated USB charger
- Programmable Analog Blocks (4 blocks)
- Programmable Digital Blocks (8 blocks)
- 3.3 Volt or 5.0 Volt Operation
- Serial Communication (3 Ports)
- Counter/Timers/PWMs (8 Ports)
- CAN (controller area network) Controller
- Battery Charge Status LED
- USER LED
- USER Pushbutton
- Optional 32Khz Oscillator for Real-Time Clock
- Optional High Precision Oscillator (High accuracy CAN/Baud rate clock)
- Optional FM24W256-G NVSRAM (32K Bytes Nonvolatile SRAM)
- thingSoC Compliant Module
- [Mikrobus Compatible Module](#)

DEFAULT	PORT	ADC	PWM
USER LED	P1_4		
I02	P1_5		X
I01	P1_6		X
SS2	P2_4	A4	X
SS1	P2_5	A5	X
IRQ	P2_6	A6	X
ATN	P2_7	A7	X
SSEL	P2_3	A3	X
SCK	P2_2	A2	X
MISO	P2_1	A1	X
MOSI	P2_0	A0	X
3.3 VOLTS	3V3		
GROUND	GND		
USER BUTTON	P3_4		

DEFAULT	PORT	PWM
CHARGE LED	LI-PO CTL	
RI	P0_7	X
DCD	P0_6	X
DSR	P0_5	X
DTR	P0_4	X
RTS	P0_3	X
CTS	P0_2	
RXD	P0_0	
TXD	P0_1	
SCL	P7_0	
SDA	P7_1	
5.0 VOLTS	SV0	
GROUND	GND	
RESET BUTTON	XRES	

LEGEND
POWER/GROUND PINS
DIGITAL PINS
ANALOG/ADC PINS
PWM PINS
TOUCH PINS
UART PINS
JTAG/DEBUG PINS
I2C BUS PINS



PatternAgents
 thingSoc
 TSOC_PSoC4L Pin Out
<http://github.com/thingsoc>