Xeno™

MattairTech

Microchip® / Atmel® ARM® Cortex® SAM D21/L21/C21/D51* Development Board

XBee® / XBee-PRO® and XBee® Cellular Radio Support

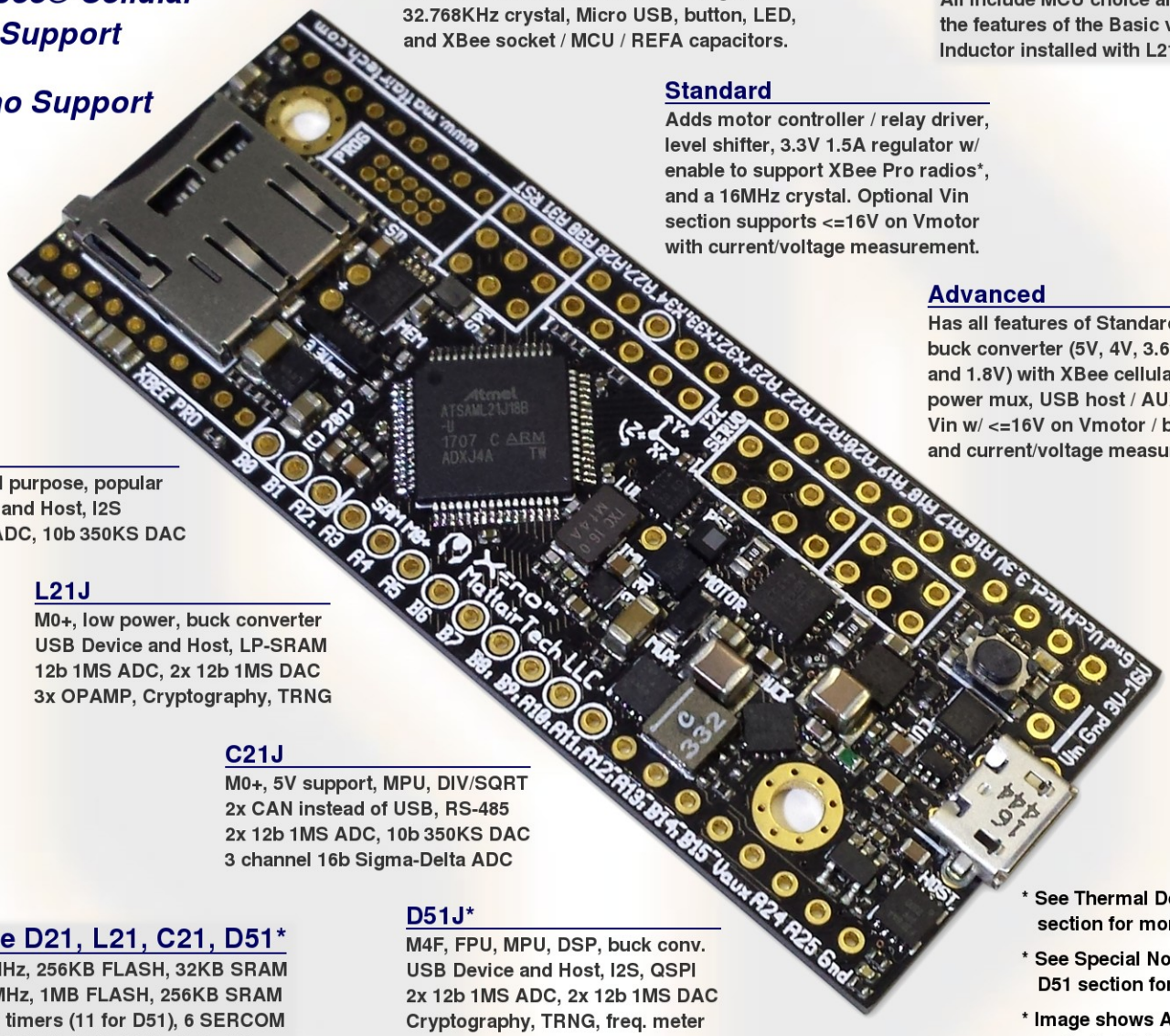
Arduino Support

Basic
Micro SD slot, main 3.3V 250mA regulator, 32.768KHz crystal, Micro USB, button, LED, and XBee socket / MCU / REFA capacitors.

Choose Board Variant
(B)asic, (S)tandard, or (A)dvanced
All include MCU choice and all of the features of the Basic variant.
Inductor installed with L21/D51.

Standard
Adds motor controller / relay driver, level shifter, 3.3V 1.5A regulator w/ enable to support XBee Pro radios*, and a 16MHz crystal. Optional Vin section supports <=16V on Vmotor with current/voltage measurement.

Advanced
Has all features of Standard, adds 2A buck converter (5V, 4V, 3.6V, 2.9V, and 1.8V) with XBee cellular support, power mux, USB host / AUX switch, Vin w/ <=16V on Vmotor / buck input, and current/voltage measurement.



D21J
M0+, general purpose, popular
USB Device and Host, I2S
12b 350KS ADC, 10b 350KS DAC

L21J
M0+, low power, buck converter
USB Device and Host, LP-SRAM
12b 1MS ADC, 2x 12b 1MS DAC
3x OPAMP, Cryptography, TRNG

C21J
M0+, 5V support, MPU, DIV/SQRT
2x CAN instead of USB, RS-485
2x 12b 1MS ADC, 10b 350KS DAC
3 channel 16b Sigma-Delta ADC

Choose D21, L21, C21, D51*
Up to 48MHz, 256KB FLASH, 32KB SRAM
D51: 120MHz, 1MB FLASH, 256KB SRAM
All have 8 timers (11 for D51), 6 SERCOM (SPI, I2C, UART), RTC, DMA, PTC, AC

D51J*
M4F, FPU, MPU, DSP, buck conv.
USB Device and Host, I2S, QSPI
2x 12b 1MS ADC, 2x 12b 1MS DAC
Cryptography, TRNG, freq. meter

* See Thermal Derating section for more info.

* See Special Notes for D51 section for caveats

* Image shows Advanced variant with L21, memory device, and sensors

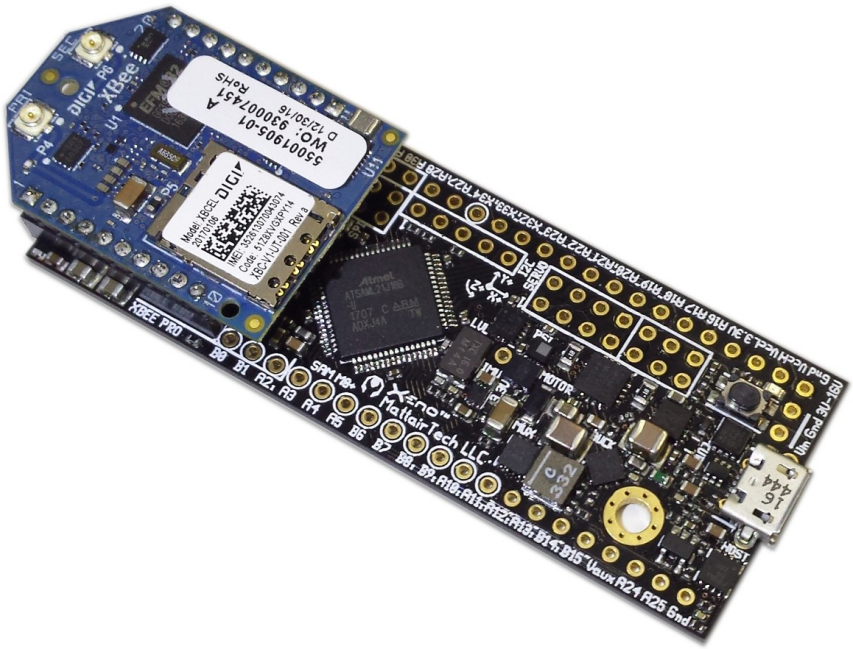
Optional SPI Memory Device
128KB SRAM with battery backup (Vbat pin)
1MB FLASH with protection features, OTP page
64KB EEPROM with OTP page

Optional I2C Sensors
3D Accelerometer / 3D Gyroscope (IMU)
Pressure / Temperature Sensor (PSI)
Both have FIFO, IMU connected to INT1

Table of Contents

Overview

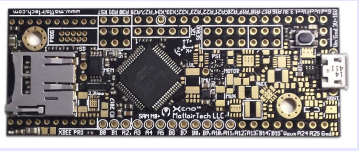
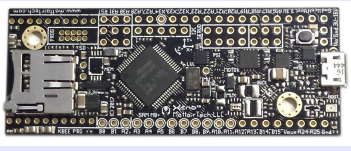
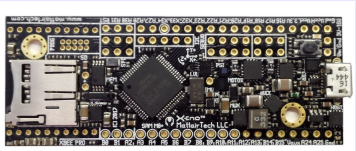
Image shows Advanced variant with Digi XBee® Cellular LTE Cat 1 modem



Introduction

The MattairTech Xeno™ is a development board for the 64-pin Microchip® / Atmel® SAMD21, SAML21, SAMC21, and SAMD51 ARM® Cortex® M0+ (M4F for D51) microcontrollers, with an XBee® radio socket, which makes it suitable for IoT, M2M, and other applications. Choose between the D21 (general purpose M0+ MCU, USB), L21 (low power features, enhanced analog, USB, crypto), C21 (5V support on board, MPU, 2x CAN instead of USB, Sigma-Delta), or D51 (M4F, FPU, MPU, DSP, USB, crypto). The Xeno is available in three variants. The Basic variant has support for standard XBee® radios. The Standard variant adds support for XBee-PRO® radios, by including a high current 3.3V, 1.5A regulator with enable. It also adds a motor controller (up to 2.26A, 5V only, but up to 17V with Vin option), a level shifter with servo support, and a 16MHz crystal. The Advanced variant adds support for Xbee cellular radios (ie: LTE Cat 1, LTE-M, 3G, etc.), as well as a 2A buck converter with selectable output voltage and frequency, a power multiplexer, and a USB Host mode / AUX switch. It also includes the Vin option, which adds a 3V-17V input and a power / current / voltage measurement IC. All variants come with the main 3.3V, 250mA regulator, Micro SD socket, Micro USB with ESD, blue LED, 32.768KHz crystal, button, and coin cell mounting. All variants support an optional SPI memory device (128KB SRAM w/backup, 1MB FLASH, or 64KB EEPROM) and optional sensors (3D accelerometer / 3D gyroscope, pressure / temperature). Arduino compatible core files for all chips (D51 coming soon) is provided. A SAM-BA USB CDC bootloader (UART only for C21) is preinstalled for programming without an external tool (Arduino or standalone use). Optional headers include the main 40-pin dual inline headers which support breadboard or perfboard mounting, Cortex debugger / programmer, two Xbee radio 10-pin sockets, SPI, I2C, Vin, and motor controller / level shifter headers.

Board Variant Features / Comparisons

Feature	Basic (B)	Standard (S)	Advanced (A)
Image			
Main Benefits / Applications	● **Development or part of final project, very flexible configuration, low power features, wide operating range, high quality parts, compact, Xbee / Xbee Pro radio socket, IoT**		
Arduino Support	● **Arduino 1.6.x compatible core (1.6.x, 1.8.x IDE), D51 support coming soon**		
Low Power	● **The design and components were chosen to support low-power operation**		

MCU (SAM M0+/M4F)	● **Microchip / Atmel ARM Cortex M0+/M4F 64-pin microcontroller with choice of:**
	  ● **D21J** (M0+, general purpose, USB, I²S, ADC, DAC, also used in the Arduino Zero)
	  ● **L21J** (M0+, low power, USB, 1MS ADC, 2x 1MS DAC, op-amps, crypto/TRNG)
	  ● **C21J** (M0+, 5V support, MPU, 2x CAN (no USB), DAC, Sigma-Delta, 2x SAR ADCs)
	  ● **D51J*** (M4F, FPU, MPU, DSP, USB, I²S, 2x 1MS ADC, 2X 1MS DAC, crypto/TRNG)
	● CPU operates at up to 48MHz (120MHz for SAMD51)
	● Up to 256KB FLASH and 32KB of SRAM (1MB FLASH and 256KB SRAM for SAMD51)
	● All chips have 8 timers (11 for D51), 6 SERCOM (SPI, I2C, UART), RTC, DMA, PTC, AC		

Main 3.3V Regulator (3.3V)	● **3.3V, 250mA LDO regulator** ● Very low ~8uA quiescent current, ~163mV@250mA dropout (BUCK post-regulation) ● Over-current and over-temperature protection (125C Tj), see Thermal Derating section		
USB Device	● **Micro USB connector with OTG and ESD protection on USB Vbus, D+, and D- lines** ● USB pins (or CAN on the C21) can be routed to header pins for panel-mount connector		
Micro SD Card (SD)	● **Micro SD card using SPI at up to 12MHz, hot insertion, push-in/push-out, card detect**		
Blue LED	● **Blue LED can be disconnected from MCU or controlled by Xbee radio, 0.5mA@3.3V**		
XBee Radio Socket	● **UART or SPI, flow control, assoc. LED, button, DTR/sleep, optional headers**		
Radio Support	● **XBee standard** (3.3V reg.)	● **XBee Pro** (3.3Vsw reg.)	● **LTE Cellular** (BUCK conv.)
High Current 3.3V Regulator, Power Switch (3.3Vsw)	Both the Standard and Basic variants support USB Vbus power only. All variants support the C21 at 5V or 3.3V.	● **3.3V, 1.5A LDO regulator with enable (MCU pin B4)** ● Can power the radio, Micro SD, and/or SPI/I2C headers ● Over-current/temp. protection, see Thermal Derating	
Motor Controller / Relay Driver (MOTOR)		● **Two 1.13A H-Bridges, using VccH or Vin (2.9V to 17V)*** ● 1 stepper motor, 2 DC motors, or 4 relays (open drain) ● 250mOhm RDSon (high/low), current limit, UVLO, sleep	
Level Shifter (LVL)		● **Level shift between VccL (MCU) & VccH (Vbuck / Vbus)** ● Auto direction sensing, ESD, connects to SERVO header	
Current / Voltage Measurement (CUR)		● **Optional on Standard variant*.** Measure Vin power, current, and voltage using I2C and/or analog output.	

Feature	Basic (B)	Standard (S)	Advanced (A)
Adjustable 2A Buck Converter (BUCK)		Both the Standard and Basic variants support USB Vbus power only. The motor controller on the Standard operates at VccH (from USB Vbus, can use USB power adapter), but can optionally be powered by Vin (2.9V to 17V) when the Vin option is ordered (the rest of the board must still be powered by USB). All variants support the C21 at 5V or 3.3V.	• **2A from Vin pin (<=17V)*** • 5V, 4V, 3.6V, 2.9V, or 1.8V • Selectable frequency • Post-regulate or direct
Power Multiplexer (MUX)			• **2A output, auto-switch** • USB Vbus or Vbuck (Vin)
USB Host / AUX Power Switch (HOST)			• **Two switches, 1A each** • 1: USB host mode (Vbus) • 2: AUX header pin output
Vbus Div. (VBDV)			• Measure Vbus on A11
VccH Div. (VHDV)			• Measure VccH on A10
Crystals	• **32.768KHz**	• **32.768KHz and 16MHz**	• **32.768KHz and 16MHz**
Optional 3D Accel-erometer / 3D Gyro (IMU), Pressure / Temp. Sensor (PSI)	• **3D Accelerometer / 3D Gyroscope on I2C, 2/4/8/16g and 125/245/500/1000/2000dps** • Interrupt routed to INT1, FIFO, 6.66KHz (accel.), 3.33KHz (gyro), low power consumption • **Pressure / Temperature Sensor on I2C, 260hPa to 1260hPa absolute range** • 24-bit pressure, 16-bit temp., FIFO, 1Hz to 75Hz ODR, low power (down to 3uA)		
Optional Memory Device (MEM)	• **Optional SPI (up to 12MHz) memory device with choice of:** ◦ 128KB SRAM with battery backup (Vbat pin) ◦ 1MB FLASH with protection features and OTP page ◦ 64KB EEPROM with OTP page		
Coin Cell (Vbat)	• **Vbat can power SRAM backup pin, VBAT pin (L21 and D51 only), and/or VccL rail** • CR2032 mounting holes, optional holder can be mounted top or bottom		
Button	• **Button configurable for reset (default), pin A31, or Xbee radio with debouncing**		
Bootloader	• **SAM-BA USB CDC bootloader (D21, L21, and D51 only, C21 has UART bootloader)** • Use Arduino IDE to upload, or Bossa command line utility for Windows, Linux, and OS X • Bootloader binaries with Micro SD card support available for download		
Microcontroller Support	• **Two 4.7Kohm resistors connected by default to pins A16 and A17 for use with I^2C** • Ferrite bead and 2 capacitors on analog supply, inductor for L21/D51 buck converter • Two capacitors can be enabled for pin A3 for use with external references		
Solder Jumpers	• **54 jumpers on PCB bottom for extreme configuration flexibility** • Power supply config, XBee (UART or SPI, button, LED), low-power config, etc.		
Headers	• Most PORT pins are routed to headers • 2 main headers are on 0.1” spacing (breadboard/perfboard mounting), 0.9” apart • **Optional Cortex Debug Header (10-pin, 50-mil) for programming and debugging**		
PCB	• High-quality PCB with gold-plated finish • Measures 2.87” x 1.10” (72.83mm x 27.94mm) and 0.062” (1.6mm) thick • **Two mounting holes: 126mil (3.2mm) diameter hole, 212mil (5.4mm) diameter pad.**		

* The Vin section (current measurement IC, reverse-polarity protection, Vin bulk capacitor, can power Vmotor) is optional on the Standard, but included on the Advanced. When not installed, only ~5V from USB Vbus is used.
* See Special Notes for D51 section for caveats on using the D51.

Configuration Options

Choose Board Variant:

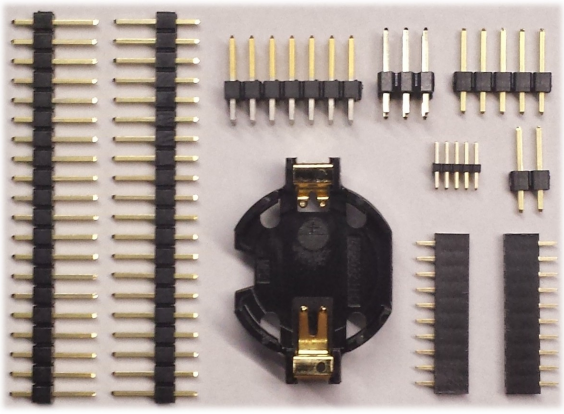
- **Basic** (USB Vbus power, Xbee standard radios)
  - 250mA regulator, Xbee radio socket, Micro SD, USB Micro, 32.768KHz, Button, LED
- **Standard** (USB Vbus power, 5V Vmotor (up to 17V with Vin option), Xbee Pro radios)
  - Adds 1.5A regulator with enable, 2.26A motor controller, level shifter, 16MHz crystal
  - **Optional Vin section** (up to 17V), current/voltage measurement, supports motor controller
- **Advanced** (Buck converter or USB Vbus power, Xbee cellular radios)
  - Adds buck converter, Vin section (up to 17V), power mux, Host/AUX switch, and more

Choose SAMD21, SAML21, SAMC21 (256KB), or SAMD51 (1MB) (see features table more info):

- **D21J** (general purpose SAM M0+ MCU, USB, I^2S, also used in the Arduino Zero)
- **L21J** (low power features, enhanced analog peripherals, USB, crypto/TRNG, custom logic)
  - Inductor installed (in place of 0ohm resistor) to support on-chip buck converter, no pin A28
- **C21J** (5V support on board, MPU, 2x CAN instead of USB, Sigma-Delta and 2x SAR ADCs)
  - Optionally connect VccL to 5V (Vbus/Vbuck), DO NOT install FLASH / sensors if using 5V
- **D51J*** (M4F CPU, FPU, MPU, DSP, USB, I^2S, 2x ADC, crypto/TRNG, custom logic)
  - Inductor installed (in place of 0ohm resistor) to support on-chip buck converter, no pin A28

Choose Optional SPI Memory Device:

- **128KB SRAM** with battery backup (supports optional coin cell holder)
- **1MB FLASH** with protection features and OTP page. Do not use at >3.6V.
- **64KB EEPROM** with OTP page
- No memory device



Optional Sensors:

- Sensors Included
  - **3D Accelerometer / 3D Gyroscope (IMU)**
  - **Pressure / Temperature Sensor (PSI)**
  - Do not use either sensor at >3.6V
- Sensors not included

Choose Optional Headers (includes coin cell holder):

- Headers Included / Xbee Headers not installed
- Headers Included / Xbee Headers installed
- Headers not included
- See 'Installing Headers' before header installation, especially with the Xbee headers.

All come with SAM-BA bootloader preinstalled:

- USB CDC Serial supported with D21, L21, and D51
- TTL Serial supported with C21

Custom Variant

All of the options can be selected and ordered online at the Xeno Custom product page:
https://www.mattairtech.com/index.php/development-boards/xeno-custom.html

Application Requirements

Xbee Radio Requirements:

- **All radios:** By default, the radio is connected to the MCU via UART with optional flow control. Solder jumpers can be used to connect to the shared SPI bus instead (including ATTN interrupt). If the Button and Blue LED option is installed (BL), they can be connected via solder jumper to the Xbee radio for commissioning button and ASSOC LED support.
- **Standard Radios:** Standard Xbee radios (smaller footprint), consume less than 100mA, so the main 3.3V regulator (250mA, see 'thermal derating') is all that is needed.
- **Pro Radios:** Xbee Pro radios can consume a few 100's of mA continuously (especially Wi-Fi). Thus, the high current regulator (HREG) must be installed in most cases. Radio power will then come from the HREG 3.3Vsw rail (solder jumper). See 'thermal derating'.
- **Cellular Radios:** Cellular radios can consume several 100's of mA continuously, and up to 1.5A transiently (2A in some cases). While the high current regulator (HREG) can supply this, it cannot do so for long because of thermal dissipation, unless the voltage drop is sufficiently low. Thus, the buck converter is required, which in turn requires the VIN option. Both linear regulators should be installed as well. Set Vbuck to 3.6V (solder jumpers) and post-regulate to 3.3V through both regulators (radio power must be set to 3.3Vsw, which comes from HREG). Alternatively, the buck converter output can be connected directly to VccL at 3.6V (3.58V), which is at the edge of the operating ranges of the microcontroller, Micro SD card, USB, sensors, and serial FLASH (radio power must be set to VccL using solder jumper). In this case, around 100uF of capacitance will be present on the radio power rail.

Other Requirements:

- **High Current:** MREG supplies 250mA, HREG supplies 1.5A, and BUCK supplies 2A. All three regulators require thermal derating (see 'Thermal Derating'). Note that BUCK can be set to 3.6V and post-regulated to 3.3V by the linear regulators, with no derating.
- **Low Power:** All components were chosen and circuitry was designed with low power consumption applications in mind. All components either have a low current sleep state or can be disconnected (except VDIV). See 'Power Supply' section for a current consumption table.
- **USB:** The C21 does not support USB. USB requires 3V to 3.6V VccL (it is 3.3V by default).
- **5V operation:** Only the C21 supports 5V, by connecting VccH to VccL (VccH from Vbus or Vbuck @5V, optionally through MUX). Do not install FLASH, sensors, or Micro SD in this case.

Graphical Configurator coming soon:

- Application requirements section (Xbee radio type, current requirements, low-power, etc.)
- Show option dependencies, ant lines to depict power flow with different configurations
- Quiescent current shown above components, output current / thermal derating information

Configuration Options

Core Components:

- Includes PCB, 32.768KHz crystal, coin cell components (diode, capacitor), pull resistors, decoupling / bypass capacitors (MCU, Xbee), AREF / VDDANA filters, and I2C pullups.

Choose SAMD21, SAML21, SAMC21 (256KB), or SAMD51 (1MB) (see features table more info):

- **D21J-256** (general purpose SAM M0+ MCU, USB, I^2S, also used in the Arduino Zero)
- **L21J-256** (low power features, enhanced analog, USB, crypto/TRNG, custom logic)
  - Inductor installed (in place of 0ohm resistor) to support on-chip buck converter, no pin A28
- **C21J-256** (5V support, MPU, 2x CAN instead of USB, Sigma-Delta and 2x SAR ADCs)
  - Optionally connect VccL to 5V (Vbus/Vbuck), DO NOT install FLASH / sensors if using 5V
- **D51J-1M*** (M4F CPU, FPU, MPU, DSP, USB, I^2S, 2x ADC, crypto/TRNG, custom logic)
  - Inductor installed (in place of 0ohm resistor) to support on-chip buck converter, no pin A28

Choose Power Inputs (Both are optional. If none chosen, provide external regulated voltage):

- **USB:** Provides both a USB data connection and ~5V from Vbus. Install in most cases. It consists of the Micro USB connector, ESD protection, and Vbus capacitor (when applicable).
- **VIN:** Supports 3V to 17V. It is used by the buck converter and/or the motor controller only. It consists of the reverse-polarity protection P-FET, the bulk capacitor, and other components. When both USB and BUCK are installed, MUX can be used to auto-select between the two.

Choose Voltage Regulators (All optional. If none chosen, provide external regulated voltage):

- **Main 3.3V Regulator (MREG):** Powers 3.3V (and VccL) rail from VccH rail. Up to 250mA (see 'Thermal Derating'). Install in most cases. Consumes 8uA quiescent current. It can optionally supply (solder jumpers) the Xbee radio, Micro SD card, and/or SPI/I2C header power pins.
- **High Current 3.3V Regulator (HREG):** Powers 3.3Vsw rail from VccH rail. Up to 1.5A (see 'Thermal Derating'). Consumes 2mA when enabled (no load), 0.1uA quiescent current when disabled. This rail can be powered down. It can optionally supply (solder jumpers) the Xbee radio, Micro SD card, and/or SPI/I2C header power pins.
- **Buck Converter (BUCK):** Powers Vbuck rail (or VccH directly when bypassing MUX) from Vin rail (3V to 17V) at up to 2A (see 'Thermal Derating'). Requires VIN option. Consumes 17uA quiescent current (8.1uA @ 5V/3.6V voltage divider current). Configurable output voltage (5V, 4V, 3.6V, 2.9V, or 1.8V) and frequency (2.5MHz and 1.25MHz). Use 3.6V for post-regulation by MREG and HREG. Can also directly power VccL.

Choose Other Hardware:

- **Button and Blue LED (BL):** The button and debounce filter are connected to reset by default, which can also be used to enter the SAM-BA bootloader (double-tap). It can connect to the MCU instead, or to the Xbee radio (commissioning button). The LED is connected to the MCU by default, but can also be controlled by the Xbee radio (ASSOC LED). It consumes ~0.5mA.
- **16MHz Crystal (HC):** Use for improved high-frequency accuracy, or lower-power applications.
- **Current Measurement (CUR):** Powered by VccL. VIN option required. Measures current /

voltage through Vin (up to 17V), which supplies both the buck converter (BUCK) and the motor controller (MOTOR), so one or both of those options is also required. I2C and/or analog MCU interface (solder jumpers). Compatible with 3V to 5V VccL. Consumes 3.5uA during sleep.

- **Power Mux / Comparator (MUX):** BUCK, VIN, and USB required. Vbuck or USB Vbus input, VccH output. Consumes 56uA quiescent current (0.2uA for the comparator and 6.1uA@5V for the divider). When both supplies are connected, Vbuck has priority, regardless of Vbuck voltage. Priority can be reversed, but only when Vbuck is 4V or less. Both the mux and comparator (and divider) can be disconnected to save power and/or bypass the mux. In this case, VccH connects directly to either USB Vbus (~5V) or Vbuck (1.8V to 5V), thus MUX is optional. Supplies 2A (1.5A min.). See 'Thermal Derating'. Do not use with 1.8V Vbuck.
- **USB Host / AUX Switch (HOST):** Powered by VccH. 1A output per switch (can combine for 2A). Consumes 3uA quiescent current (53uA typical when switch(s) on, 3.5uA@5V divider current). Requires BUCK (set to 5V) when supplying Vbus (USB host mode).
- **Vbus Voltage Divider (VDIV):** Cannot disconnect via jumper, don't install to save 7.7uA@5V.
- **Motor Controller (MOTOR):** Vmotor can come from either Vin (3V to 17V, requires VIN option) or VccH (usually ~5V through schottky diode). Consumes 1uA during sleep. Up to 1.13A per H-bridge (see 'thermal derating'). Can use for up to 4 relays (low-side drive). When MOTOR not installed, the control input MCU pins are shorted to the output header pins.
- **Level Shifter (LVL):** Powered by VccL on low voltage side, VccH on high voltage side. Consumes 3.5uA quiescent current. Bi-directional, auto direction sensing. Low current output drive (one-shot high current drive to change states, then bus hold low current drive). When LVL not installed, the low-side MCU pins are shorted to the high-side header pins.
- **Micro SD Card (SD):** Jumper selectable power, VccL or 3.3Vsw (requires HREG option). Micro SD cards can consume several 10's of uA when not selected, so connect to 3.3Vsw (which can powered down) to turn off. Do not use with >3.6V. Connected to shared SPI bus.

Choose Optional SPI Memory Device (All powered by VccL. All connect to shared SPI bus):

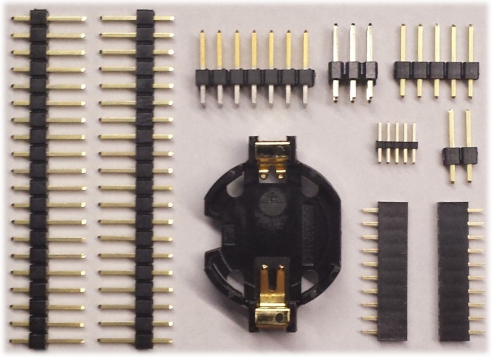
- **128KB SRAM** with battery backup (coin cell option). Consumes 4uA quiescent current.
- **1MB FLASH** with protection and OTP page. 2uA quiescent current. Do not use with >3.6V.
- **64KB EEPROM** with OTP page. Consumes 2uA quiescent current.

Optional Sensors (Both powered by VccL, do not use at >3.6V, Both connect to I2C bus):

- **3D Accelerometer / 3D Gyroscope (IMU).** Consumes 6uA quiescent current.
- **Pressure / Temperature Sensor (PSI).** Consumes 1uA quiescent current.

Choose Optional Headers / Coin Cell Holder:

- **Standard / Cortex Headers (HI):** Not installed
- **Xbee Headers:** Installed **(XI)** or Not Installed **(XIN)**
- **Battery Holder (BH):** for CR2032 Lithium 3V coin cell.
- See 'Installing Headers' first, especially with Xbee.



All come with SAM-BA bootloader preinstalled:

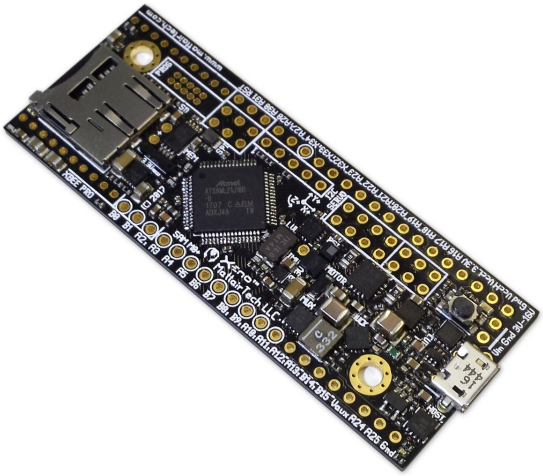
- USB CDC Serial supported with D21, L21, and D51
- TTL Serial supported with C21

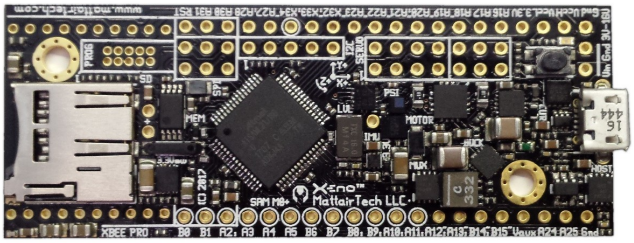
Custom Variant Options Table

Option	SKU	Notes / Dependencies	Price
PCB / Core Components		Required (32.768KHz crystal, pullups, capacitors, etc.)	$14.00
Microcontroller (required, choose one)	D21J-256	General purpose M0+ MCU	$4.00
	D51J-1M	See 'Special Notes for D51' in Xeno Manual first	$6.00
	L21J-256	Internal buck converter supported	$7.00
	C21J-256	5V support. No USB support.	$5.00
	CUSTOM	Send email with choice of compatible MCU	$10.00
16MHz Crystal	HC	Improved high-freq. accuracy or low-power applications	$1.00
Button and Blue LED	BL	Use with MCU or Xbee radio	$2.00
Main 3.3V Regulator	MREG	Supplies 250mA to 3.3V rail and usually VccL from VccH	$3.00
High Current Regulator	HREG	Supplies 1.5A to 3.3Vsw rail from VccH	$7.00
Buck Converter	BUCK	Supplies 2A to Vbuck rail from Vin. Requires VIN.	$12.00
Vin Power Input	VIN	Used for BUCK or MOTOR only. 3V to 17V supported.	$4.00
USB	USB	Can supply one MUX input or VccH directly	$3.00
Current Measurement	CUR	Vin power input (measure motor and system)	$6.00
Power Mux / Comparator	MUX	Auto-selects between USB and BUCK (both required).	$8.00
USB Host / AUX Switch	HOST	Requires BUCK for USB host mode (includes shield filter)	$5.00
Vbus Voltage Divider	VDIV	Cannot disconnect via jumper, consumes 7.7uA@5V	$1.00
Motor Controller	MOTOR	Requires VIN (3V to 17V) or VccH (USB or BUCK)	$9.00
Level Shifter	LVL	Bidirectional level translation between VccL and VccH.	$3.00
Micro SD Card Slot	SD	Powered by VccL (usually MREG) or 3.3Vsw (HREG)	$3.00
Memory Device (choose none or one)	SRAM	Powered by VccL	$6.00
	FLASH	Powered by VccL. Do not use >3.6V.	$4.00
	EEPROM	Powered by VccL	$4.00
Accelerometer / Gyroscope	IMU	Powered by VccL. Do not use >3.6V.	$8.00
Pressure / Temperature	PSI	Powered by VccL. Do not use >3.6V.	$7.00
Standard / Cortex Headers	HI	See 'Installing Headers' before installation	$2.00
Xbee Headers (choose none or one)	XI	Xbee headers included. See 'Installing Headers' first.	$2.00
	XIN	Xbee Headers Installed	$3.00
Battery Holder	BH	See 'Installing Headers' before installation	$1.00

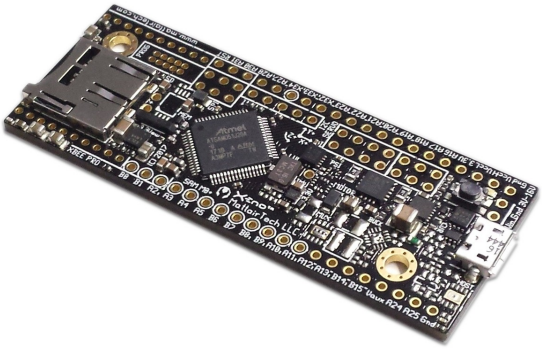
Photos

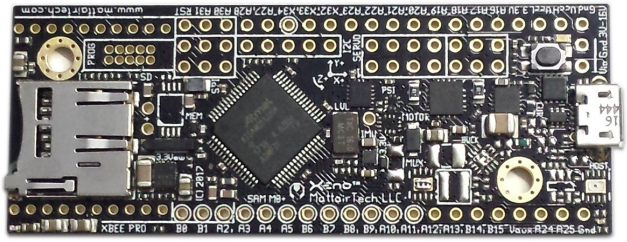
Advanced



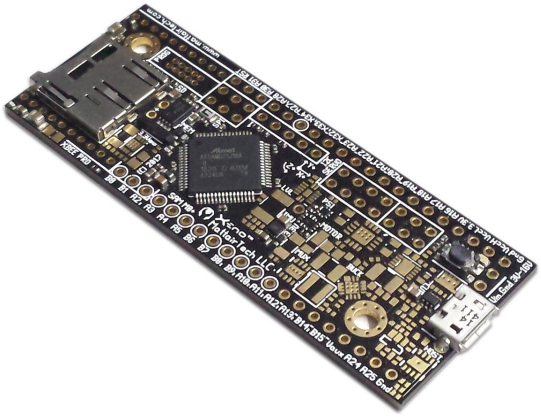


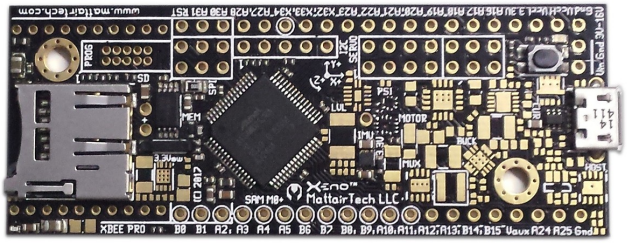
Standard





Basic





Special Notes for D51

The Xeno was designed for the SAMD21, SAML21, and SAMC21. However, the SAMD51 is pin compatible, and shares some of the features of the L21 (ie: buck converter, Vbat pin). Thus, support for it was added, but only after the pcb was designed. As a result, there are a few caveats when using the Xeno with the D51. These differences are primarily due to the fact that the D51 has additional limits on which pins can be used with a peripheral instance. Often a peripheral pad can be mapped to more than one pin, and many peripherals have a group of pads (ie: SERCOM). With the D21, L21, or C21, any pin shown in the datasheet 'I/O Multiplexing and Considerations' can be used. With the D51, IOSET pin grouping was introduced, and only certain pins may be grouped together (usually only pins that are physically close). The Xeno pcb layout restrictions meant using SERCOM pins that were not physically adjacent. A future revision of the pcb (Rev C) will eliminate the need for the jumper wires.

In order to use the SPI bus (Micro SD, optional memory device, SPI mode of Xbee radio):

- Needed because SERCOM5 on the D51 must use IOSET2.
- Solder a jumper wire between A21 and B30 (S43, pin 1 of the SPI header, MISO).
- Solder a jumper wire between A23 and B22 (S45, pin 4 of the SPI header, MOSI).
- Solder a jumper wire between A22 and B23 (S44, pin 3 of the SPI header, SCK).
- B22, B23, and B30 cannot be used for other purposes (leave floating)

In order to use the Xbee radio UART interface (use SPI mode to avoid this fix):

- Needed because D51 UART transmit data pinout options (TXPO) do not include TX on pad 2.
- Solder a jumper wire between B8 (SERCOM4, TX pad 0) and A18 (XBDI). Solder a jumper wire between B9 (SERCOM4, RX pad 1) and A31 (XBDO).
- A18 and A31 cannot be used for other purposes (leave floating)

Other considerations:

- Motor controller: While there are timers on all pins, IOSETs may impact usage. See datasheet.
- Level shifter: While there are timers on all pins, IOSETs may impact usage. See datasheet.
- Use the high gain setting with the 32.768KHz crystal.
- The 4-bit SD peripheral (only on D51) is not connected to the Micro SD card. SPI mode only.
- The QSPI peripheral (only on D51) is not connected to the optional Memory. SPI mode only.
- The I2C bus should use SERCOM1 rather than SERCOM3 (both available on the same pins).
- Check the errata for rev A chips (current rev. as of writing). Vbat does not work. Do not alter BOD33 Disable fuse bit (use register), use the INTREF reference only above 0C, and more.
- Due to errata, the DAC cannot use the VDDANA reference. As a workaround, connect a jumper wire between VccL and pin A3 (REFA) and enable the capacitors by setting J33. Then use the external unbuffered reference.

Microcontroller Features / Comparisons

	ATSAMD21J	ATSAML21J	ATSAMC21J	ATSAMD51J
Product Slides	http://atmel.force.com/support/servlet/fileField?id=0BEG000000003Jc	http://atmel.force.com/support/servlet/fileField?id=0BEG000000002Wx	http://atmel.force.com/support/servlet/fileField?id=0BEG000000003Hq	
Datasheet	http://ww1.microchip.com/downloads/en/DeviceDoc/40001882A.pdf	http://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-42385-SAM-L21-Datasheet.pdf	http://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-42365-SAM-C21_Datasheet.pdf	http://www.microchip.com/mymicrochip/filehandler.aspx?ddocname=en599585
Voltage	● 1.62V – 3.63V	● 1.62V – 3.63V	● **2.7V – 5.5V**	● **1.71V – 3.63V**
Processor	● ARM Cortex-M0+ CPU ● Up to 48MHz ● Von-Neumann ● Single-cycle hardware multiplier ● Micro Trace Buffer	● ARM Cortex-M0+ CPU ● Up to 48MHz ● Von-Neumann ● Single-cycle hardware multiplier ● Micro Trace Buffer	● ARM Cortex-M0+ CPU ● Up to 48MHz ● Von-Neumann ● Single-cycle hardware multiplier ● Micro Trace Buffer ● **Memory Protection Unit (MPU)**	● **ARM Cortex-M4F CPU** ● **Up to 120MHz** ● **Harvard Architecture** ● **Floating Point Unit** ● **Hardware Division** ● **DSP, Saturated Math** ● **8-Zone Memory Protection Unit** ● **4KB Combined Cache** ● Embedded Trace Module
Memories	● 32/64/128/256KB in-system self-programmable Flash ● 4/8/16/32KB SRAM Memory	● 32/64/128/256KB in-system self-programmable Flash ● **1/2/4/8KB Flash Read-While-Write section** ● 4/8/16/32KB SRAM Memory ● **2/4/8/8KB SRAM Low power Memory**	● 32/64/128/256KB in-system self-programmable Flash ● **1/2/4/8KB independent self-programmable Flash for EEPROM emulation** ● 4/8/16/32KB SRAM Memory	● **1MB/512KB/256KB in-system self-programmable Flash** ● **Dual Bank with Read-While-Write support** ● **256/192/128KB SRAM** ● **Up to 4KB TCM** ● **Up to 8KB backup mode SRAM** ● **ECC available for Flash and SRAM**
System	● Power-on reset (POR) and brown-out detection (BOD) ● Internal and external clock options with **48MHz Digital Frequency Locked Loop (DFLL48M)** and 48MHz to 96MHz Fractional Digital Phase Locked Loop (FDPLL96M) ● External Interrupt Controller (EIC) ● 16 external interrupts ● One non-maskable interrupt ● Two-pin Serial Wire Debug (SWD) programming, test and debugging interface	● Power-on reset (POR) and brown-out detection (BOD) ● Internal and external clock options with **48MHz Digital Frequency Locked Loop (DFLL48M)** and 48MHz to 96MHz Fractional Digital Phase Locked Loop (FDPLL96M) ● External Interrupt Controller (EIC) ● 16 external interrupts ● One non-maskable interrupt ● Two-pin Serial Wire Debug (SWD) programming, test and debugging interface	● Power-on reset (POR) and brown-out detection (BOD) ● Internal and external clock options with 48MHz to 96MHz Fractional Digital Phase Locked Loop (FDPLL96M) ● External Interrupt Controller (EIC) ● 16 external interrupts ● One non-maskable interrupt ● Two-pin Serial Wire Debug (SWD) programming, test and debugging interface	● Power-on reset (POR) and brown-out detection (BOD) ● Internal and external clock options with **48MHz Digital Frequency Locked Loop (DFLL48M) and two 96MHz to 200MHz Fractional Digital Phase Locked Loops (FDPLL200M)** ● External Interrupt Controller (EIC) ● 16 external interrupts ● One non-maskable interrupt ● Two-pin Serial Wire Debug (SWD) programming, test and debugging interface
Low Power	● Idle and standby sleep	● Idle, Standby, **Backup,**	● Idle, standby, **and off**	● Idle, Standby, **Backup,**

	ATSAMD21J	ATSAML21J	ATSAMC21J	ATSAMD51J
	modes ● SleepWalking peripherals	**and Off** sleep modes ● SleepWalking peripherals ● **Static and Dynamic Power Gating Architecture** ● **Battery backup support** ● **Two Performance Levels** ● **Embedded Buck/LDO regulator supporting on-the-fly selection**	sleep modes ● SleepWalking peripherals	**Hibernate, and Off** sleep modes ● SleepWalking peripherals ● **Battery backup support** ● **Embedded Buck/LDO regulator supporting on-the-fly selection**
DMA	● 12-channel Direct Memory Access Controller (DMAC)	● **16-channel** Direct Memory Access Controller (DMAC)	● 12-channel Direct Memory Access Controller (DMAC)	● **32-channel** Direct Memory Access Controller (DMAC)
Event System	● 12-channel Event System	● 12-channel Event System	● 12-channel Event System	● **32-channel** Event System
16-bit Timers (TC)	● Five 16-bit Timer/Counters (TC), configurable as either: ● One 16-bit TC with compare/capture channels ● One 8-bit TC with compare/capture channels ● One 32-bit TC with compare/capture channels, by using two TCs	● Five 16-bit Timer/Counters (TC) **including one low-power TC**, each configurable as: ● One 16-bit TC with two compare/capture channels ● One 8-bit TC with two compare/capture channels ● One 32-bit TC with two compare/capture channels, by using two TCs	● Five 16-bit Timer/Counters (TC), configurable as either: ● One 16-bit TC with compare/capture channels ● One 8-bit TC with compare/capture channels ● One 32-bit TC with compare/capture channels, by using two TCs	● **Six 16-bit** Timer/Counters (TC), configurable as either: ● One 16-bit TC with compare/capture channels ● One 8-bit TC with compare/capture channels ● One 32-bit TC with compare/capture channels, by using two TCs
24/16-bit Timers (TCC)	● Two 24-bit Timer/Counters and one 16-bit Timer/Counter for Control (TCC), with extended functions: ● Up to four compare channels with optional complementary output ● Generation of synchronized pulse width modulation (PWM) pattern across port pins ● Deterministic fault protection, fast decay and configurable dead-time between complementary outputs ● Dithering that increase resolution with up to 5 bit and reduce quantization error	● Two 24-bit Timer/Counters and one 16-bit Timer/Counter for Control (TCC), with extended functions: ● Up to four compare channels with optional complementary output ● Generation of synchronized pulse width modulation (PWM) pattern across port pins ● Deterministic fault protection, fast decay and configurable dead-time between complementary outputs ● Dithering that increase resolution with up to 5 bit and reduce quantization error	● Two 24-bit Timer/Counters and one 16-bit Timer/Counter for Control (TCC), with extended functions: ● Up to four compare channels with optional complementary output ● Generation of synchronized pulse width modulation (PWM) pattern across port pins ● Deterministic fault protection, fast decay and configurable dead-time between complementary outputs ● Dithering that increase resolution with up to 5 bit and reduce quantization error	● Two 24-bit Timer/ Counters and **three** 16-bit Timer/Counter for Control (TCC), with extended functions: ● **Up to six** compare channels with optional complementary output ● Generation of synchronized pulse width modulation (PWM) pattern across port pins ● Deterministic fault protection, fast decay and configurable dead-time between complementary outputs ● Dithering that increase resolution with up to 5 bit and reduce quantization error
RTC	● 32-bit Real Time Counter (RTC) with	● 32-bit Real Time Counter (RTC) with	● 32-bit Real Time Counter (RTC) with	● 32-bit Real Time Counter (RTC) with

	ATSAMD21J	ATSAML21J	ATSAMC21J	ATSAMD51J
	clock/calendar function	clock/calendar function	clock/calendar function	clock/calendar function
WDT	● Watchdog Timer (WDT)	● Watchdog Timer (WDT)	● Watchdog Timer (WDT)	● Watchdog Timer (WDT) **with windows mode**
USB / CAN	● **One full-speed (12Mbps) Universal Serial Bus (USB) 2.0 interface** ● Embedded host and device function ● Eight endpoints	● **One full-speed (12Mbps) Universal Serial Bus (USB) 2.0 interface** ● Embedded host and device function ● Eight endpoints	● **Up to two Controller Area Network (CAN) interfaces:** ● CAN 2.0A/B ● CAN-FD 1.0 ● Each CAN interface have two selectable pin locations to switch between two ● external CAN transceivers (without the need for an external switch)	● **One full-speed (12Mbps) Universal Serial Bus (USB) 2.0 interface** ● Embedded host and device function ● Eight endpoints
SERCOM	● Up to six Serial Communication Interfaces (SERCOM), each configurable to operate as either: ● USART with full-duplex and single-wire half-duplex configuration ● I^2C up to 3.4MHz ● SPI ● LIN slave	● Up to six Serial Communication Interfaces (SERCOM) **including one low-power SERCOM**, each configurable to operate as either: ● USART with full-duplex and single-wire half-duplex configuration ● I^2C up to 3.4MHz ● SPI ● LIN slave	● Up to six Serial Communication Interfaces (SERCOM), each configurable to operate as either: ● USART with full-duplex and single-wire half-duplex configuration ● I^2C up to 3.4MHz ● SPI ● LIN **master**/slave ● **RS-485**	● Up to six Serial Communication Interfaces (SERCOM), each configurable to operate as either: ● USART with full-duplex and single-wire half-duplex configuration ● I^2C up to 3.4MHz ● SPI (**with inter-byte space**) ● LIN **master**/slave ● **ISO7816** ● **RS-485**
I^2S	● **One two-channel Inter-IC Sound (I^2S) interface**			● **One two-channel Inter-IC Sound (I^2S) interface**
SD(HC)				● **One SD(HC) Memory Card Interface**
ADC	● One 12-bit, 350ksps Analog-to-Digital Converter (ADC) with up to 20 channels ● Differential and single-ended input ● **1/2x to 16x programmable gain stage** ● Automatic offset and gain error compensation ● Oversampling and decimation in hardware to support 13-, 14-, 15-, or 16-bit resolution	● One 12-bit**, 1MSPS Analog-to-Digital Converter (ADC) with up to 20 channels** ● Differential and single-ended input ● Automatic offset and gain error compensation ● Oversampling and decimation in hardware to support 13-, 14-, 15-, or 16-bit resolution ● **Can use with 3 OPAMPs**	● **Two 12-bit, 1Msps Analog-to-Digital Converter (ADC) with up to 12 channels each (20 unique channels in total)** ● Differential and single-ended input ● Automatic offset and gain error compensation ● Oversampling and decimation in hardware to support 13-, 14-, 15- or 16-bit resolution ● **One 16-bit Sigma-Delta Analog-to-Digital Converter** (SDADC) with up to 3 differential channels	● **Two 12-bit, 1Msps Analog-to-Digital Converter (ADC) with up to 16 channels each (20 unique channels in total)** ● Differential and single-ended input ● Automatic offset and gain error compensation ● Oversampling and decimation in hardware to support 13-, 14-, 15- or 16-bit resolution
DAC	● 10-bit, 350ksps Digital-	● **Two 12-bit, 1MSPS**	● 10-bit, 350ksps Digital-	● **Two 12-bit, 1MSPS**

	ATSAMD21J	ATSAML21J	ATSAMC21J	ATSAMD51J
	to-Analog Converter (DAC)	**Dual Output** Digital-to-Analog Converter (DAC)	to-Analog Converter (DAC)	**Dual Output** Digital-to-Analog Converter (DAC)
AC	● Two Analog Comparators (AC) with window compare function	● Two Analog Comparators (AC) with window compare function	● **Four** Analog Comparators (AC) with window compare function ● **<50ns**	● Two Analog Comparators (AC) with window compare function
Custom Logic		● **One Configurable Custom Logic (CCL)**	● **One Configurable Custom Logic (CCL)**	● **One Configurable Custom Logic (CCL)**
PTC	● Peripheral Touch Controller (PTC) ● 256-Channel capacitive touch and proximity sensing	● Peripheral Touch Controller (PTC) ● **169-Channel** capacitive touch and proximity sensing ● **Wake-up on touch in standby mode**	● Peripheral Touch Controller (PTC) ● 256-Channel capacitive touch and proximity sensing ● **DMA**	● Peripheral Touch Controller (PTC) ● 256-Channel capacitive touch and proximity sensing (32 self-capacitive) ● **Wake-up on touch**
Crypto		● **One AES encryption engine** ● **One True Random Generator (TRNG)**		● **One AES encryption engine with 256-bit key and 2MB/s rate** ● **One True Random Generator (TRNG)** ● **Public Key Crypto with RSA, DSA, and Elliptic Curves** ● **Integrity Check Module (SHA1, SHA224, SHA256)**
I/O	● Up to **52** programmable I/O pins	● Up to 51 programmable I/O pins	● Up to **52** programmable I/O pins	● Up to 51 programmable I/O pins
Other	● CRC-32 generator	● CRC-32 generator ● **Three Operational Amplifiers (OPAMP)**	● CRC-32 generator ● **Frequency Meter** ● **Integrated Temperature Sensor** ● **Hardware Divide and Square Root Accelerator (DIVAS)**	● **One Quad I/O Serial Peripheral Interface** ● CRC-32 generator ● **Up to 3 wake-up pins** with tamper detection ● **Position Decoder** ● **Frequency Meter** ● **Temperature Sensor** ● **Parallel Capture Controller (PCC)**

SAMx21 Development Software

Environment	Windows	Linux	OS X
Standalone	SAM-BA bootloader, SAM-BA bootloader utility, Test program coming soon	SAM-BA bootloader, SAM-BA bootloader utility, Test program coming soon	SAM-BA bootloader, SAM-BA bootloader utility
Arduino	Arduino core files includes SAM-BA bootloader and bootloader utility, Test sketch coming soon	Arduino core files includes SAM-BA bootloader and bootloader utility, Test sketch coming soon	Arduino core files includes SAM-BA bootloader and bootloader utility, Test sketch coming soon
Atmel Studio (coming soon)	Test program project files	NA	NA
PlatformIO	November 2017?	November 2017?	November 2017?

SAMx51 Development Software

Environment	Windows	Linux	OS X
Standalone	SAM-BA bootloader, SAM-BA bootloader utility, Test program coming soon	SAM-BA bootloader, SAM-BA bootloader utility, Test program coming soon	SAM-BA bootloader, SAM-BA bootloader utility
Arduino (coming soon)	Arduino core files soon! includes SAM-BA bootloader and bootloader utility, Test sketch coming soon	Arduino core files soon! includes SAM-BA bootloader and bootloader utility, Test sketch coming soon	Arduino core files soon! includes SAM-BA bootloader and bootloader utility, Test sketch coming soon
Atmel Studio (coming soon)	Test program project files	NA	NA
PlatformIO	Winter 2017?	Winter 2017?	Winter 2017?

Pinout

Arduino Pinout

```
============================= MattairTech Xeno (ATSAMx21Jxxx) ============================
Other  COM   PWM  Analog  INT  Arduino*           Arduino*  INT  Analog  PWM   COM  Other
==========================================================================================
                                  ------------------
         Board Variant:           | (no external pin) |
 B=Basic, S=Standard, A=Advanced  |            |-- B3 | 49   I   O       VBAT(L)/SDCD(+B)
    M=Memory device installed     |            |-- B5 | 48   I                   INT1(+B)
                                  |            |-- B4 | 47                       3SEN(+B)
                                  |                   |
XBDS(B)                O        0 | B0            RST |                          BOOT(+B)
MECS(+M)               O        1 | B1            A31 | 31            RX1  SWDIO/XBDO(+B)
DAC0                   O   I    2 | A2            A30 | 30                 SWDCLK(+B)
REFA(B)                O        3 | A3        A28(D/C)| 28                   SHCS(D/C)
3SVO(S)/REFB           O        4 | A4            A27 | 27   I                    INT2
DAC1(L)                O        5 | A5        X34 (B2)| 34       O   TC60~   LED(+B)/XBRT
CMVO(A)                O        6 | B6        X33(B16)| 33   I              INT0(+B)/BTN
ASEN(+A)               O        7 | B7        X32(B17)| 32   I       TC61~       MOPS(+S)
          TX3          O   I    8 | B8            A23 | 23           TC41~
          RX3          O   I    9 | B9            A22 | 22           TC40~
VHDV(A)   MOSI1        O   I   10 | A10           A21 | 21           TC71~
VBDV(+A)  SCK1         O   I   11 | A11           A20 | 20   I       TC70~
XBCT(B) SDA1/MISO1 TCC20   I   12 | A12           A19 | 19           TC31~        CMRI(S)
        SCL1/SS1  TCC21    I   13 | A13           A18 | 18           TC30~  TX1  XBDI(+B)
HSEN(A)         TC50~      I   14 | B14           A17 | 17                  SCL   SCL(+B)
BKFS(+A)        TC51~          15 | B15           A16 | 16                  SDA   SDA(+B)
                                  | Vaux         3.3V |
USB D- (D/L)+B, CAN TX (C)        | A24   _____  VccL |   ! VccL is 3.3V by default.
USB D+ (D/L)+B, CAN RX (C)        | A25  |     | VccH |     DO NOT exceed 3.6V on VccL or
                                  | Gnd  | USB |  Gnd |     on any IO pin with the D21 or
       Chip Variant:               ------------------      L21 installed. 5V is allowed
    D=D21, L=L21, C=C21                                   ONLY with the C21 installed.
                                 1------------------      By default, VccH is 5V.
MISO(+B)                    43 | B30          Vcon |
SCK(+B)                     44 | B23    SPI    B22 | 45                       MOSI(+B)
SHCS(D/C) or (SDCS(+B))  28(46)| A28(B31)      Gnd |
                                  ------------------

                                 1------------------
LVL_SHIFT_0(+S) TX2    O   I   35 | A6             A7 | 36   I   O    RX2 LVL_SHIFT_1(+S)
LVL_SHIFT_2(+S)  TCC12 O  NMI  37 | A8    LEVEL    A9 | 38       O TCC13  LVL_SHIFT_3(+S)
                                  | VccH  SHIFT  VccH |
                                  | Gnd           Gnd |
MOTOR_BOUT1(+S)    TCC04       39 | B10           B11 | 40       TCC05    MOTOR_BOUT2(+S)
MOTOR_AOUT1(+S)    TCC06       41 | B12   MOTOR   B13 | 42       TCC07    MOTOR_AOUT2(+S)
                                  | Vmotor        Gnd |
                                  ------------------
```

* Most pins can be used for more than one function. The same port pin number printed on the board is also used in Arduino (without the 'A') for all of the Arduino functions.

* Different variants have different hardware installed onboard. The alternate functions column shows for which board variant(s) the associated hardware is installed: B=Basic, S=Standard, A=Advanced, and M=Memory device installed. The Advanced variant has all of the hardware from the Standard, and the Standard has all of hardware from the Basic.

+ This function is enabled by default depending on the variant indicated by the letter. Thus, the associated header pin cannot be used. In most cases (except most +A pins), solder jumpers can be used to enable or disable the alternate onboard function.

~ TC3, TC4, TC5, TC6, TC7 on the D21 are instead TC4, TC0, TC1, TC2, TC3 on the L21/C21.

Silkscreen Legend:
  Top: A circle around pin is analog function, '~' is timer, small 'I' is interrupt
  Bottom: A box around pin means 'Other' function enabled by default depending on variant

Installing Headers

Please read this entire section before installing headers.

Installing Xbee Headers

Most 2mm female headers (especially low-profile headers), including the optional headers shipped with the Xeno, require extra care when soldering, because the sockets can very easily fill with solder, rendering the headers useless. Unlike more common 100-mil headers, the 2mm headers do not have a mechanical barrier preventing solder from flowing into the socket. To prevent this:

- Install the Xbee radio into the socket to act as a heat sink. You may want to start soldering on the ends, then periodically wiggle the radio to make sure pins are not being soldered.
- Use a relatively low soldering iron temperature.
- Pay attention to how much solder is flowing into the hole. Unlike 100-mil headers, solder may keep flowing into the hole rather than build up around the pin (the heat sinking should help).

Installing Cortex Header

Because the Cortex header is located under the Xbee radio, and because each radio has a different module thickness on the bottom, if and how the header is installed depends on the radio. In most cases, the header can be installed normally. With thicker modules, it may be possible to install the header with the shorter pins facing up (check to make sure your programming cable pins make contact). With the thickest modules (ie: Digi Wi-Fi radio), the header cannot be installed. It is possible to program the MCU by installing the header directly into the cable connector, then inserting the other side into the pcb and using sideways pressure to keep contact. It is also possible to simply install higher-profile headers.

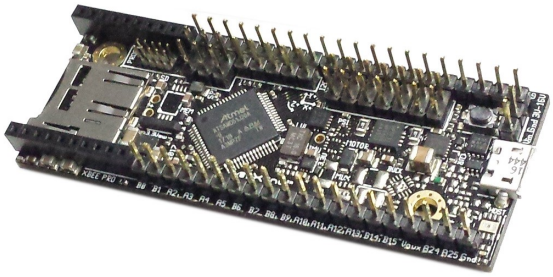
Installing Main / SPI Headers

When using an Xbee Pro radio, the first three pins on each header closest to the radio socket cannot be installed, nor can the SPI header. The full main headers (20x2) can be installed on the bottom. With Xbee standard, there is no header interference, so all headers can be installed on top.

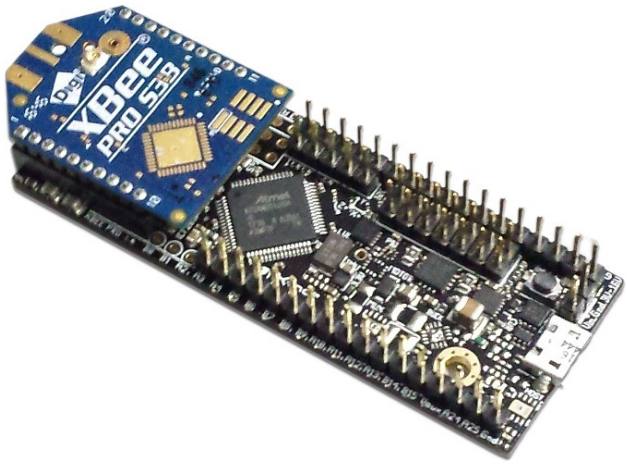
Installing Coin Cell Holder

When installing the optional coin cell holder, both the SPI and I2C headers cannot be installed. Additionally, the holder cannot be installed on top if using an Xbee radio (standard or pro).

Xbee Radio Headers

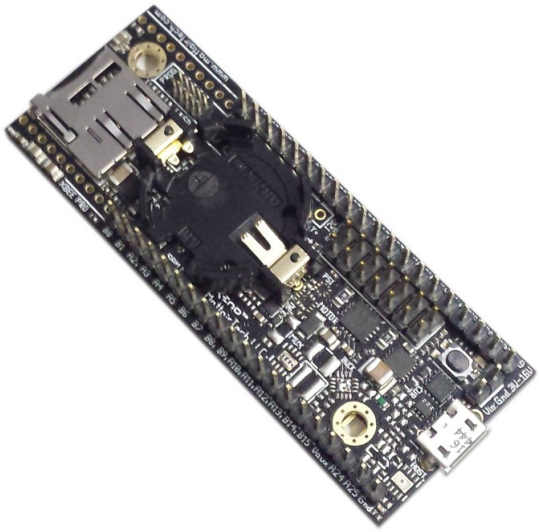


All headers installed on top. An Xbee standard (smaller form factor) radio can be installed, but not a coin cell holder.

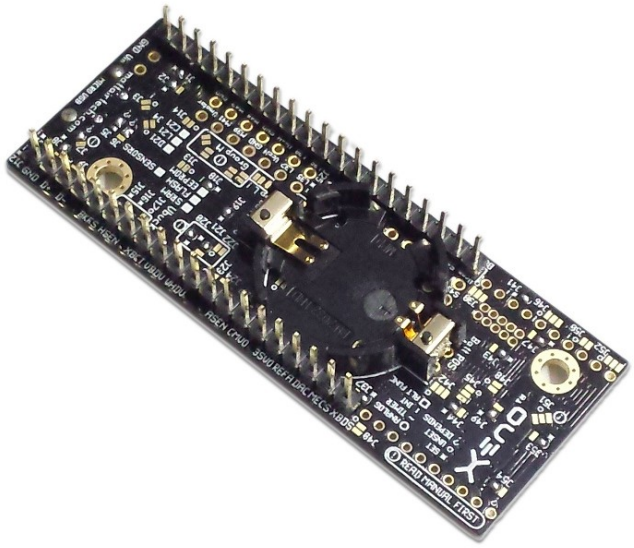


Xbee Pro radio installed, with main headers on top. In this case, the three pins on each header closest to the radio socket cannot be installed, nor can the SPI header. The full main headers (20x2) can be installed on the bottom.

Coin Cell Holder



Coin cell holder installed on top. An Xbee radio cannot be installed.



Coin cell holder installed on bottom, allowing Xbee radio to be installed. Main headers also shown on the bottom (which allows Xbee Pro without losing 6 pins), but note that both cannot be installed at the same time on the bottom.

Power Supply

Introduction

VccL power rail (main lower voltage rail, 3.3V by default)

- Powers MCU, memory device, sensors, level shifter (low-side), & current measurement IC
- By default uses main 3.3V regulator (input from VccH). Instead can connect to VccH (use buck converter or USB Vbus for voltage other than 3.3V). See 'Power Supply Configurations'.
- Can supply externally regulated voltage on VccL pin. 1.71V-3.63V (C21: 2.7V-5.5V). See J19.

VccH power rail (main higher voltage rail, 5V by default), Buck Converter / Power Multiplexer

- VccH powers both 3.3V regulators, level shifter (high-side), Host switch, and Vmotor (default).
- **Basic and Standard:** VccH is connected to USB Vbus (~5V). There is no user configuration.
- **Advanced:** Includes buck converter with configurable voltage (5V default) and Vin input (up to 17V). VccH is connected to power mux which auto-selects between Vbuck (5V, 4V, or 3.6V) which has priority, or Vbus (~5V). Mux can be bypassed, connecting VccH directly to Vbuck (5V, 4V, 3.6V, 2.9V, or 1.8V) or to Vbus (~5V). Post regulate 3.6V for best combination of power efficiency, low ripple, and thermal derating. See 'Power Supply Configurations'.

High Current 3.3V Regulator with Enable (3.3Vsw) / Motor Controller (Standard and Advanced)

- Allows higher current consumption (1.5A) and ability to power down connected devices. Three power rails can connect to either 3.3Vsw or VccL: Vxbee, Vsd, and Vcon. See 'XBee Radio', 'SD Card', 'High Current 3.3V Regulator', and 'Power Rails Table'. See J54, J53, and J43.
- Motor controller by default connected to VccH (5.5V max) through schottky diode. Can instead connect to Vin rail at up to 17V (Vin is optional with Standard). See 'Motor Controller' chapter.
- Level shifter translates bi-directionally between VccL and VccH. See 'Level Shifter' chapter.

Coin Cell (Vbat)

- **L21/D51 Vbat pin:** Provides backup power to MCU (ie: RTC). See 'Coin Cell (Vbat)' section.
- **SRAM Vbat pin:** Can provide backup power to optional SRAM. See 'Memory Device (MEM)'.
- **Vbat to VccL:** Vbat can connect to VccL through a schottky diode for use as a backup supply or for low power operation. See 'Coin Cell (Vbat)' section in 'Other Hardware' chapter.

Chip Specific Support

- **5V operation (C21 only!):** VccL can connect to either USB Vbus (~5V), or the buck converter output (5V, 4V, 3.6V, or 2.9V; 3.3V from regulator only). See 'Power Supply Configurations'.
- **On-chip buck converter (L21 / D51 only):** An inductor replaces the 0ohm resistor, and along with 1uF (4.7uF for D51) & 100nF capacitors, supports the buck converter for higher efficiency. Note that the L21 and D51 lack pin A28 (uses for VSW instead). No user configuration needed.

CAUTION
Do not operate the D21, L21, D51, SD card, optional sensors, or optional FLASH memory at >3.6V. See 'Thermal Derating' for information on current limits for a given ambient temperature. All power supply ICs will enter thermal shutdown if they get too hot.

Power Supply Configurations

Basic

See 'Main 3.3V Regulator (3.3V)' section. Vxbee, Vsd, and Vcon source select jumpers all set to VccL.

VccL @ 3.3V from main regulator (default)

- Connect Vbus to VccH by setting J7 (Vaux-VccH-Vbus) toward the question mark. See J7.
- Disconnect VccL from VccH by clearing solder from all three pads of J32 (VccL-NC-VccH).

VccL powered by VccH directly @ ~5V from USB Vbus (C21 only!)

- **Do not operate D21, L21, D51, SD card, optional sensors, or optional FLASH at >3.6V.**
- Connect Vbus to VccH by setting J7 (Vaux-VccH-Vbus) toward the question mark. See J7.
- Connect VccL to VccH by soldering all three pads of J32 (VccL-NC-VccH). See J32.

Standard

See 'Main 3.3V Regulator' and 'High Current 3.3V Regulator' sections.
Vsd and Vcon source select jumpers set to VccL. Vxbee set to 3.3Vsw.

The Standard has the same VccL/VccH configuration options as the Basic, see above.

Advanced

Please read this entire chapter to understand the more complicated power supply of the Advanced.
The solder jumper table contains additional information, so consult this table when referenced below.
Vsd and Vcon source select jumpers are set to VccL. Vxbee is set to 3.3Vsw.
Always drive BKFS (otherwise, it is floating). Use low-frequency for higher efficiency but larger ripple.

VccL @ 3.3V from main regulator,
VccH @ 5V, 4V, or 3.6V (Vbuck) or ~5V (Vbus) auto-selected by the multiplexer (default)

- Vbuck is 5V by default. Multiplexer enabled (Vbuck has priority by default).
- Disconnect VccL from VccH by clearing solder from all three pads of J32 (VccL-NC-VccH).
- Disconnect Vbus from VccH by clearing solder from all three pads of J7 (Vaux-VccH-Vbus).
- Connect the mux by setting J11, J12, J20, and setting J21 (toward asterisk).
- Multiplexer priority: With J8 set (comparator enabled), when Vbuck power is good (PG high), Vbuck is selected, regardless of the voltage of either Vbuck or Vbus. When PG is low, the mux auto-selects the source with the higher voltage (used for Vbuck or Vbus alone). Alternatively, the comparator can be disabled by clearing J8, so the mux will always be in auto-select mode. In this case, 5V Vbuck cannot not be used (ambiguous with Vbus 5V, could oscillate between sources), so use only 4V or 3.6V. Priority will then be to Vbus when both sources connected.
- Set buck converter voltage using 'Buck Converter Output Voltage Table'. See 'Buck Converter'. Post regulate 3.6V for best combination of power efficiency, low ripple, and thermal derating. If using 3.6V, set J22 to short the main regulator input diode and do not use coin cell for VccL.
- If using Vaux-Vbus set J10, otherwise clear it. See 'USB Host / Aux Power Switch'.
- Set 3.3V-VccL (J19). Clear all three pads of VccL-NC-3.3Vsw (see J51 for details).

VccL @ 3.3V from main regulator, VccH @ 5V, 4V, or 3.6V from buck converter (Vbuck)

- Vbuck is 5V by default. Multiplexer bypassed.
- Disconnect VccL from VccH by clearing solder from all three pads of J32 (VccL-NC-VccH).
- Disconnect Vbus from VccH by clearing solder from all three pads of J7 (Vaux-VccH-Vbus).
- Disconnect the mux by clearing J8, J11, J12, J20, and setting J21 (away from asterisk, VccH-Vbuck). Host switch disconnected (AUX connected). See 'USB Host / Aux Power Switch'.
- Set buck converter voltage using 'Buck Converter Output Voltage Table'. See 'Buck Converter'. Post regulate 3.6V for best combination of power efficiency, low ripple, and thermal derating. If using 3.6V, set J22 to short the main regulator input diode and do not use coin cell for VccL.
- Set 3.3V-VccL (J19). Clear Vaux-Vbus (J10) and all 3 pads of VccL-NC-3.3Vsw (see J51).

VccL @ 3.3V from main regulator, VccH @ ~5V from USB Vbus

- Multiplexer bypassed. Vbus capacitance will exceed USB spec. Use USB power adapter.
- Disconnect VccL from VccH by clearing solder from all three pads of J32 (VccL-NC-VccH).
- Connect Vbus to VccH by setting J7 (Vaux-VccH-Vbus) toward the question mark. See J7.
- Disconnect the mux and Vbuck by clearing J8, J11, J12, J20, and clearing solder from all three pads of J21. Host switch disconnected (AUX connected). See 'USB Host / Aux Power Switch'.
- If using Vaux-Vbus set J10, otherwise clear it. See 'USB Host / Aux Power Switch'.
- Set 3.3V-VccL (J19). Clear all three pads of VccL-NC-3.3Vsw (see J51 for details).

VccL powered by VccH directly @ 5V, 4V, 3.6V, 2.9V, or 1.8V from buck converter

- **Do not operate D21, L21, D51, SD card, optional sensors, or optional FLASH at >3.6V.**
- Vbuck is 5V by default. Multiplexer bypassed. Do not use coin cell for VccL.
- Connect VccL to VccH by soldering all three pads of J32 (VccL-NC-VccH). See J32.
- Disconnect Vbus from VccH by clearing solder from all three pads of J7 (Vaux-VccH-Vbus).
- Disconnect the mux by clearing J8, J11, J12, J20, and setting J21 (away from asterisk, VccH-Vbuck). Host switch disconnected (AUX connected). See 'USB Host / Aux Power Switch'.
- Set buck converter voltage using 'Buck Converter Output Voltage Table'. See 'Buck Converter'. When using 1.8V or 2.9V, be sure to check BOD setting in MCU. **Use 4V & 5V with C21 only!**
- Set all 3 pads of VccL-NC-3.3Vsw (J51) to add 32uF from regulator output (ie: **cell modem**).
- VccL Capacitance: 64uF + 32uF (if J51 set) + 11uF (Vxbee) + 10.1uF (Vsd) = 117.1uF
- Clear 3.3V-VccL (J19) and Vaux-Vbus (J10).

VccL powered by VccH directly @ ~5V from USB Vbus (C21 only!)

- **Do not operate D21, L21, D51, SD card, optional sensors, or optional FLASH at >3.6V.**
- Multiplexer bypassed. **C21 only!** Do not use coin cell for VccL.
- Connect VccL to VccH by soldering all three pads of J32 (VccL-NC-VccH). See J32.
- Connect Vbus to VccH by setting J7 (Vaux-VccH-Vbus) toward the question mark. See J7.
- Disconnect the mux and Vbuck by clearing J8, J11, J12, J20, and clearing solder from all three pads of J21. Host switch disconnected (AUX connected). See 'USB Host / Aux Power Switch'.
- If using Vaux-Vbus set J10, otherwise clear it. See 'USB Host / Aux Power Switch'.
- Set all 3 pads of VccL-NC-3.3Vsw (J51) to add 32uF from regulator output (ie: **cell modem**).
- VccL Capacitance: 42uF + 32uF (if J51 set) + 11uF (Vxbee) + 10.1uF (Vsd) = 95.1uF
- Vbus capacitance will exceed USB spec. Use USB power adapter.
- Clear 3.3V-VccL (J19).

VccL powered by VccH directly,
VccH @ 5V, 4V, or 3.6V (Vbuck) or ~5V (Vbus) auto-selected by the multiplexer (C21 only!)

- **Do not operate D21, L21, D51, SD card, optional sensors, or optional FLASH at >3.6V.**
- Vbuck 5V by default. Mux enabled (Vbuck has priority by default). Do not use coin cell for VccL
- Connect VccL to VccH by soldering all three pads of J32 (VccL-NC-VccH). See J32.
- Disconnect Vbus from VccH by clearing solder from all three pads of J7 (Vaux-VccH-Vbus).
- Connect the mux by setting J11, J12, J20, and setting J21 (toward asterisk).
- Multiplexer priority: With J8 set (comparator enabled), when Vbuck power is good (PG high), Vbuck is selected, regardless of the voltage of either Vbuck or Vbus. When PG is low, the mux auto-selects the source with the higher voltage (used for Vbuck or Vbus alone). Alternatively, the comparator can be disabled by clearing J8, so the mux will always be in auto-select mode. In this case, 5V Vbuck cannot not be used (ambiguous with Vbus 5V, could oscillate between sources), so use only 4V or 3.6V. Priority will then be to Vbus when both sources connected.
- Set buck converter voltage using 'Buck Converter Output Voltage Table'. See 'Buck Converter'.
- If using Vaux-Vbus set J10, otherwise clear it. See 'USB Host / Aux Power Switch'.
- Set all 3 pads of VccL-NC-3.3Vsw (J51) to add 32uF from regulator output (ie: **cell modem**).
- VccL Capacitance: 42uF + 32uF (if J51 set) + 11uF (Vxbee) + 10.1uF (Vsd) = 95.1uF
- Clear 3.3V-VccL (J19).

Advanced Power Configurations Table*

VccL-VccH	3.3V-VccL	3.3Vsw-VccL	Power Mux			USB Host / AUX			
J32	J19	J51	J8	J20	J21 (3-pad)	J7 (3-pad)	J10	J11	J12
OFF	ON	OFF	CHOOSE	ON	MUX_OUT - VccH	ALL OFF	CHOOSE	ON	ON
OFF	ON	OFF	OFF	OFF	VccH - Vbuck	ALL OFF	OFF	OFF	OFF
OFF	ON	OFF	OFF	OFF	ALL OFF	VccH - Vbus	CHOOSE	OFF	OFF
ON	OFF	CHOOSE	OFF	OFF	VccH - Vbuck	ALL OFF	OFF	OFF	OFF
ON	OFF	CHOOSE	OFF	OFF	ALL OFF	VccH - Vbus	CHOOSE	OFF	OFF
ON	OFF	CHOOSE	CHOOSE	ON	MUX_OUT - VccH	ALL OFF	CHOOSE	ON	ON

** Please read the corresponding section under 'Power Supply Configurations' → 'Advanced'.*

Power Rails Table

Rail	Voltage and Source	Capacitance*	Description and Notes
VccL Main Rail	**3.3V (default):** 3.3V from main regulator **VccH:** When using buck converter (or USB Vbus) to set VccL to any voltage other than 3.3V (VccL-VccH jumper set, Use caution!, see J32). **VccL pin:** A regulated voltage can be applied. 1.71V to 3.63V (2.7V to 5.5V for C21). See J19. **Vbat:** When Vbat is connected to VccL through the schottky diode (J42) and VccL is below 2.7V-2.8V (3V cell), then Vbat powers VccL.	4*100nF + 4.7uF from 3.3V rail (or VccH or Vbat rails) + 4.7uF VddAna Optional: 100nF level shifter 100nF sensors 100nF memory Add capacitance from Vxbee and Vsd if connected. If the Vin option is installed, there is 1.1uF behind a 100ohm resistor (current monitor).	This is the main lower voltage rail used to power the microcontroller, memory device, sensors, the low-voltage side of the level shifter, and the current measurement IC. It can optionally power the Vxbee, Vsd, and Vcon rails. **When connecting to VccH or using the VccL pin, do not operate the D21, L21, D51, SD card, sensors, or FLASH memory at >3.6V.** Vbat: Use as a backup supply to VccL or for very low power operations. Do not use when VccL-VccH jumper set. If VccL-3.3Vsw jumper set, adds 32uF.
VccH Main Rail	**Basic and Standard:** **5V** from USB Vbus (VccH-Vbus jumper) **Advanced:** **5V (default)** from power mux (Vbuck or USB Vbus). Mux can also supply 4V or 3.6V from buck converter. Mux can be bypassed, connecting VccH directly to either Vbuck (5V, 4V, 3.6V, 2.9V, or 1.8V) or to Vbus (~5V).	Basic: 4.7uF Vbus Standard: 10.1uF 10uF 3.3Vsw input 100nF level shifter Advanced: 36.8uF 22uF mux output 10uF 3.3Vsw input 4.7uF host switch 100nF level shifter All variants have 1uF on main 3.3V regulator input, after the diode.	This is the main higher voltage rail used to power both 3.3V regulator inputs, the high-voltage side of the level shifter, the Host/AUX switch, and by default the motor controller. If VccL to VccH jumper set, then add capacitance from VccL. If Vbuck to VccH jumper set, add 22uF from Vbuck. Both jumpers may be set. If the motor controller is installed, there is additional 10uF after the VccH-Vmotor schottky diode.
3.3V Regulator	**3.3V** output from VccH input See 'Thermal Derating'	4.7uF output	This is the main 3.3V 250mA regulator output rail. By default, it is routed to VccL through J19. It is also routed to the main header 3.3V pin.
3.3Vsw Regulator Standard / Advanced Only	**3.3V** output from VccH input See 'Thermal Derating'	22uF output + 10uF (near Vxbee) Add capacitance from Vxbee and Vsd if connected. By default (except with Basic), Vxbee	This is the ouput rail for the high-current 3.3V 1.5A regulator with enable (configurable power-up state). Three power rails can connect to either 3.3Vsw or VccL (Vxbee, Vsd, and Vcon). Devices connected to 3.3Vsw can consume more current than VccL, as well as be powered off.

Rail	Voltage and Source	Capacitance*	Description and Notes
		is connected.	When turning off 3.3Vsw, do not drive or pull any lines high that are routed to devices powered by 3.3Vsw. See 'High Current 3.3V Regulator'.
Vbuck Regulator Advanced Only	**5V, 4V, 3.6V, 2.9V, or 1.8V** from Vin See 'Thermal Derating'	22uF output	This is the ouput rail for the 2A buck converter with configurable output voltage and frequency. It connects to one of the power mux inputs (the other is Vbus). See 'Buck Converter'.
Vbus External	**5V** (4.5V to 5.5V) from USB Vbus Basic and Standard: Vbus is connected to VccH Advanced: Vbus is connected to power multiplexer input	Basic: 4.7uF Vbus + 1uF VccH Standard: 0uF Vbus + 11.1uF VccH Advanced: 4.7uF Vbus	This is connected to the USB Micro connector Vbus pin. Vbus can be connected to Vaux so that it is available on the main header and/or to double the Host/AUX switch output current to 2A. The Standard lacks the 4.7uF capacitor near the USB connector, because VccH is tied to Vbus, and VccH has 10uF from the high current regulator input. See 'USB Host/AUX Switch'.
Vin External Standard (option) / Advanced Only	**3V to 17V** Note that the silkscreen is printed with 16V	22uF (23uF with buck converter installed)	Current from the Vin header pin flows through a PFET for reverse-polarity protection and through a current sense resistor to a 22uF bulk capacitor for use by the buck converter and/or motor controller. See 'Vin Section / Current Measurement IC'.
Vaux External	**VccH** Supports 1.8V to 5.5V Output (default) or input depending on config.	4.7uF + Vbus capacitance if connected.	This connects to the AUX output of the Host/AUX switch. It can also be tied to Vbus to double output current or use as input. See 'USB Host/AUX Switch'.
Vcon Peripheral	**VccL (default)** or 3.3Vsw	Capacitance from VccL or 3.3Vsw only	Vcon is available on two external pins (both SPI and I2C headers). Connect this rail to 3.3Vsw to supply more current and/or to turn off completely.
Vxbee Peripheral	**Basic: VccL only** **Standard & Advanced: 3.3Vsw (default)** or VccL	10uF + 1uF + 13pF + capacitance from VccL or 3.3Vsw Up to 117uF total	Connected to Vxbee pin of the Xbee radio socket. Connect this rail to 3.3Vsw to supply more current and/or to turn off completely. For more power and/or capacitance, see 'Xbee Radio'.
Vsd Peripheral	**VccL (default)** or 3.3Vsw	10uF + 100nF (after ferrite bead)	Connected to the Micro SD card slot. Can be hot-inserted without causing reset. Connect to 3.3Vsw to turn off.
Vmotor Peripheral	**VccH (default)** or Vin **VccH:** supports ~5V from	10uF with VccH or 22uF with Vin (23uF with buck	Connected to the motor controller supply input (Vin), which uses it for the H-bridges as well as for the logic

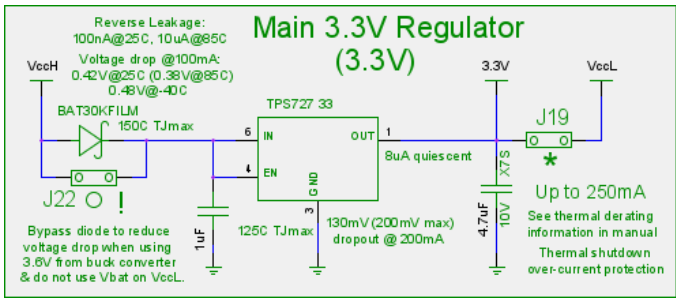
Rail	Voltage and Source	Capacitance*	Description and Notes
Standard / Advanced Only	USB Vbus or 5V, 4V, 3.6V, or 2.9V (light loads) from the buck converter. **Vin:** supports 3V to 17V.	converter installed)	supply (via an internal 4.85V linear regulator). Vmotor also connects to the Vmotor header pin through a 2.5A standard fuse, for use with four independent open-drain outputs.
Vbat Battery	**2.7V to 2.8V** typical from a 3V coin cell (diode drop)	4.7uF Note that both POS pins must be connected (the holder does this).	Connected to the POS terminal of the optional coin cell holder. This rail can be directly connected to the optional SRAM Vbat pin and/or the Vbat pin of the L21 or D51. Vbat can also connect to VccL through a schottky diode.
VddCore Internal	**1.2V** from on-chip linear regulator (all chips) or on-chip buck converter (L21 and D51). VccL input. **Do not use externally!**	1uF (4.7uF for D51) + 100nF	Connected to the MCU VDDCORE pin, which is the output from the on-chip linear regulator. The L21 and D51 also have an on-chip buck converter that can be enabled on the fly. This rail powers the cpu and most peripherals.
VddAna Internal	**VccL**	4.7uF + 100nF (after ferrite bead)	This is connected to the MCU VDDANA pin, which powers analog peripherals as well as some PORT pins.
RefA Internal	**1V (2V for C21 ADC) to VddAna-0.6V**	4.7uF + 100nF when J33 set	This is connected to the MCU REFA pin (A3), which can be used as a voltage reference for the ADC and/or DAC.

** Capacitance may need to be taken into account for the Xbee radio or other external hardware. The high current 3.3V regulator can handle additional external capacitance. Avoid adding external capacitance to the main 3.3V regulator output (VccL by default) when Vxbee connected to VccL.*

Main 3.3V Regulator (MREG, 3.3V rail)

All variants include the main 3.3V, 250mA linear regulator, which supplies the 3.3V rail and (by default) the VccL rail (MCU). VccL by default supplies the Vsd and Vcon rails and optionally Vxbee.

Features (from Texas Instruments datasheet)



- Texas Instruments TPS72733
- 3.3V output from up to 5.5V input
- Up to 250mA (see thermal derating)
- +/- 2.5mV accuracy @ 25C
- 2% accuracy over line, load, temperature
- Fast transient response
- 50mV load transient (200mA step)
- 1.9V UVLO, 2KV HBM ESD
- 130mV (200mV max.) dropout @ 200mA
- Low dropout works for post-regulation of 3.6V from buck converter. 50dB PSRR @ 10mA and 2.5MHz (27dB @ 200mA). Subtract 6dB when using 1.25MHz. No thermal derating with 3.6V.
- Schottky diode on input prevents backflow current from Vbat coin cell. Also dissipates a small amount of power. Can be bypassed (ie: post-regulation of 3.6V from buck converter).
- 7.9uA (12uA max.) ground pin current @ 0mA, 130uA @ 250mA.
- Thermal protection, short-circuit protection
- -40C to 125C junction temperature

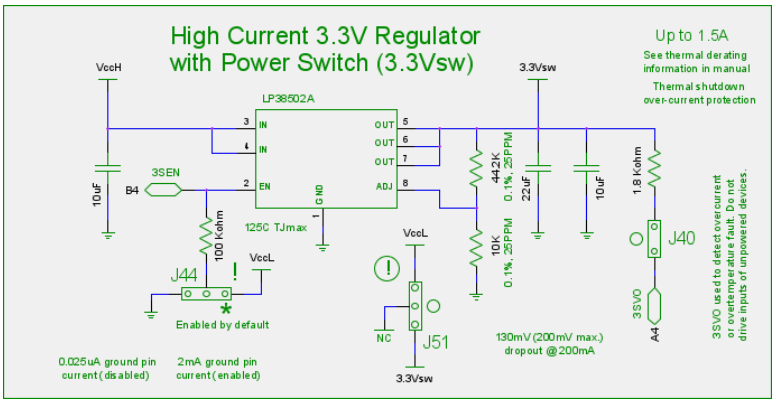
Hardware Settings

- **See 'Thermal Derating' section for more information.**
- J19 is set by default, connecting the main 3.3V regulator output to VccL. Clear J19 when connecting VccL to VccH, or when using an externally regulated voltage on the VccL pin.
  - J19 can also be used to measure VccL current by clearing the jumper and measuring across the 3.3V and VccL main header pins.
  - VccL is the main lower voltage rail used to power the microcontroller, optional memory device, optional sensors, the low-voltage side of the level shifter, and the current measurement IC. VccL by default supplies the Vsd and Vcon rails and optionally Vxbee.
- Set J22 to short the input diode of the main 3.3V regulator to eliminate the voltage drop when using the buck converter at 3.6V. See 'Coin Cell' in 'Other Hardware'.
  - Do not use the coin cell with VccL (can still use for MEM and MCU Vbat) when J22 is set.
- **With all variants (especially the Advanced), see 'Power Supply Configurations' above.**

High Current 3.3V Regulator with Enable (HREG, 3.3Vsw rail)

With the Standard and Advanced variants, a 1.5A, 3.3V regulator is installed which supports Xbee Pro radios. The output rail (3.3Vsw) can optionally power the Xbee radio (default), Micro SD card, and/or two Vcon header pins. The rail can be powered down for low-power operations.

Features (from Texas Instruments datasheet)



- Texas Instruments LP38502A
- 3.3V output from up to 5.5V input
- Up to 1.5A (see thermal derating)
- 1.5% accuracy at 25C (A grade)
- 3% accuracy over line, load, temp.
- Supports large output capacitance
- 40mV load transient (1.4A step)
- 2KV HBM ESD
- 220mV (275mV max.) dropout @ 1.5A,25C. 375mV max. across temp.
- Low dropout works for post-regulation of 3.6V from buck converter. At least 33dB PSRR @ 1A and 1.25MHz. Higher at 2.5MHz. This figure may be lower when using 3.6V input. Thermal derating not required with 3.6V.
- Enable input with a configurable default powerup state (on or off). On by default. 25us delay.
- 3SVO can be used to detect over-current or over-temperature faults.
- VccL to 3.3Vsw can be used to add 32uF when VccL is tied to VccH (ie: cell modem).
- 0.025uA (0.125uA max.) ground pin current when disabled. 15uA max. across temperature.
- 2mA (3.5mA max.) ground pin current when enabled. 4.5mA max. across temperature.
- -40C to 125C junction temperature, Thermal protection, short-circuit protection

Hardware Settings

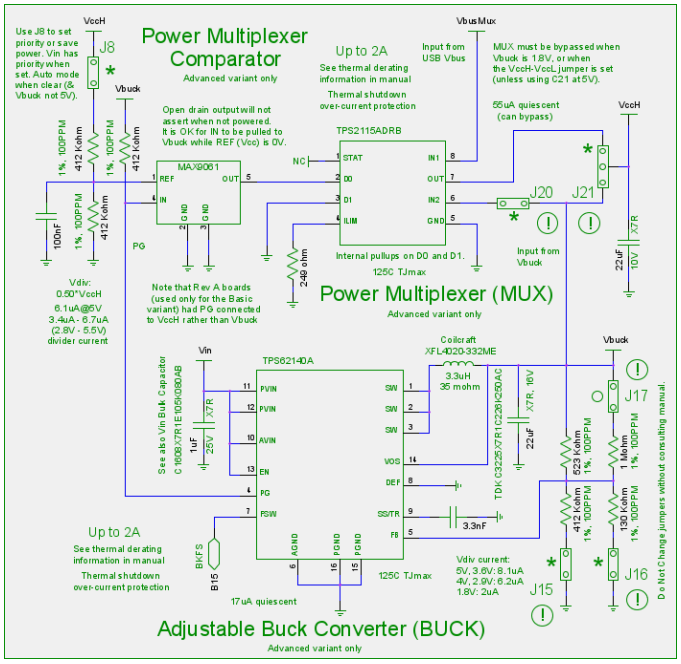
- **See 'Thermal Derating' section for more information.**
- Enable input with a configurable default powerup state (on or off). On by default (J44=VccL).
  - Drive high to enable, low to disable. Pull resistor current is 25uA @3.3V when opposing.
  - Default powerup state: Set J44 to Gnd (away from asterisk) to disable at powerup.
  - When turning off 3.3Vsw, be sure not to drive or pull any lines high that are routed to devices powered by 3.3Vsw. There is a current limiting resistor installed in series with the TX line (XBDI) to the Xbee radio, so this is not a concern with the default configuration.
- J40 can be set to connect the regulator output (3.3Vsw) to 3SVO (A4) through a 1.8Kohm resistor for comparator (AIN0) threshold detection or for ADC measurement. This allows detection of an over-temperature or over-current fault. If a fault is detected, be sure not to drive or pull any lines high that are routed to devices powered by 3.3Vsw.
- 3.3Vsw-VccL jumper: Set J51 to add 32uF to VccL when VccL=VccH (**see 'Power Supply'**).

Buck Converter (BUCK)

With the Advanced variant, a 2A buck converter with selectable output voltage (5V by default, 4V, 3.6V, 2.9V, or 1.8V) is installed, with support for Xbee cellular radios. The input can be up to 17V from Vin and the output is either to the power multiplexer or directly to VccH. Frequency is selectable.

Features (from Texas Instruments datasheet)

- Texas Instruments TPS62140A
- Up to 2A output, DCS-Control topology
- Input voltage range: 3V to 17V
- Adjustable output voltage:
  5V (default), 4V, 3.6V, 2.9V, or 1.8V
- Post-regulate (3.6V, 4V, 5V) or direct to VccL
- Power good (PG) controls power multiplexer
- RDSon: 90mohm (H), 40mohm (L) typ.
  (170mohm and 70mohm max); Vin>=6V
- Selectable frequency, either 2.5MHz (default) or 1.25MHz (higher efficiency, larger ripple).
- 100% Duty-Cycle Mode (low dropout)
- Ripple: 10-15mV, up to 2x @ low/high current
- Thermal protection, short-circuit protection
- 2.8V UVLO, 1.65ms soft-start time
- 17uA (25uA max.) quiescent current
  (2uA to 8.2uA divider current).
- -40C to 125C junction temp., 2KV HBM ESD



Hardware Settings / Notes

- **See 'Thermal Derating' section for more information.**
- Select operating frequency by driving BKFS (B15) low (2.5MHz) or high (1.25MHz, larger ripple but higher efficiency). BKFS is always connected and floating, so it should **always be driven**.
- Post-regulate (3.6V, 4V, 5V) or direct to VccL (5V, 4V, 3.6V, 2.9V, or 1.8V).
  - Use 3.6V for best combination of power efficiency, low ripple, and thermal derating.
- Power good (PG) switches mux to Vbuck when high (by default). See 'Power Multiplexer'.
- Use J15, J16, and J17 to set the output voltage (5V by default, 4V, 3.6V, 2.9V, or 1.8V). These jumpers are among several jumpers that must be changed as a group. **See 'Power Supply Configurations',** 'Buck Converter Output Voltage Table' and 'IC Voltage Compatibility Table'.
- 458mA power save mode (PSM) threshold @ 1.25MHz, 229mA @ 2.5MHz.
- At low currents (below 10-60mA), output can be few 10's of mV above set point, depending on input voltage (up to 35mV @ 3.3V out and 17V in), otherwise 10mV above set point normally.
- Switching frequency falls as voltage falls when in power save mode (PSM) or nearing dropout.
- See 'Vin Section / Current Measurement IC' chapter for details on the power input.

Buck Converter Output Voltage Table

Voltage	J17 (1M-Vbuck)	J15 (412K-Gnd)	J16 (130K-Gnd)	Divider Current
5V (5.03V), default	off	on	on	8.1uA
4V (4.02V)	off	off	on	6.2uA
3.6V (3.58V)	on	on	on	8.1uA
2.9V (2.91V)	on	off	on	6.2uA
1.8V (1.82V)	off	on	off	2uA
1.47V*	on	on	off	2uA
0.80V*	on	off	off	0uA
0.80V*	off	off	off	0uA

** = Invalid setting. Note: 523Kohm to Vbuck is always present. Parallel combo of 412Kohm and 130Kohm is 98819 ohm. Parallel combo of 523Kohm and 1Mohm is 343401 ohm.*

IC Voltage Compatibility Table

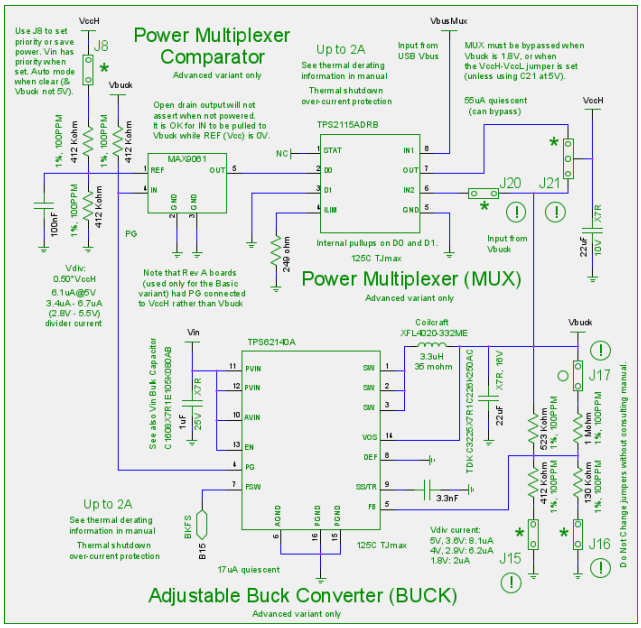
IC	1.8V	2.9V	3.3V	3.6V	4.0V	5.0V
D21 / L21 / D51	*	*	*	*	!!	!!
C21		*	*	*	*	*
USB (D21/L21/D51), out of spec at 2.9V		?	*	*	!!	!!
Blue LED, dim at 2.9V		?	*	*	*	*
Level shifter A (VccL)	*	*	*	*		
Level shifter B (VccH)	*	*	*	*	*	*
Motor Controller		?	*	*	*	*
Current Sense IC Vdd			*	*	*	*
USB Host / Aux Switch	*	*	*	*	*	*
Power Multiplexer		*	*	*	*	*
Multiplexer Comparator	*	*	*	*	*	*
3D Accelerometer / 3D Magnetometer	*	*	*	*	!!	!!
Pressure / Temperature Sensor	*	*	*	*	!!	!!
SRAM		*	*	*	*	*
EEPROM	*	*	*	*	*	*
FLASH		*	*	*	!!	!!
Micro SD Card		*	*	*	!!	!!
Main 3.3V regulator		~	~	*	*	*
High Current 3.3Vsw regulator		~	~	?	*	*

? = near low voltage limit, special handling may be needed, ~ = regulator output will fall out of regulation
! ! = DO NOT USE device at 4.0V or 5.0V. It is OK for level shifter A side to be up to 5.5V (not operational).

Power Multiplexer (MUX) / Comparator

With the Advanced variant, a power multiplexer with one output and two input sources (USB Vbus and buck converter output) is installed. If only one input exists, the mux will select it. If both inputs exist, the buck converter is selected (default, see J8), regardless of Vbuck voltage. Mux can be bypassed.

Power Multiplexer Features (from Texas Instruments datasheet)



- Texas Instruments TPS2115A
- Switches between Vbuck and Vbus (USB)
- Current limit set at ~2A (1.5A? min.)
- 2.8V to 5.5V supply range (no 1.8V Vbuck)
- auto-switch or bypass
- Controlled output voltage transition
- Reverse and cross-conduction blocking
- RDSon = 84mOhm typ. (150mOhm max.)
- ~2.55V UVLO, 2KV HBM ESD
- 55uA (91uA max.) quiescent current
- Can be bypassed to reduce current to 0uA
- -40C to 125C junction temp., thermal protection

Comparator Features (Maxim datasheet)

- Maxim MAX9061
- Inverting input, open-drain output
- Used for both inversion and level translation
- Buck converter power good (PG) controls IN input. When PG high, MUX switches to Vbuck.
- 0.9V to 5.5V supply voltage range (REF, which is both the reference input and Vcc)
- IN and OUT pins can be up to 5.5V even if Vcc = 0V (no clamp diodes)
- 0.17uA (0.35uA max.) operating current plus 3.4uA to 6.7uA from divider (2.8V to 5.5V)
- Can be disconnected to save power (both IC and voltage divider) or change source priority
- -40C to 85C operating temperature, 3ms. powerup time, 25us propagation delay

Hardware Settings

- **See 'Thermal Derating' section for more information.**
- Actual current limit range is about 1.5A to 2.5A across temperature, so **1.5A minimum**.
- Both MUX and comparator support all buck converter output voltages except 1.8V.
- J20, J21, and J8 are used to bypass the mux. These jumpers are among several jumpers that must be changed as a group. **See 'Power Supply Configurations'.**
- J8 used to supply power and a reference voltage to comparator REF (Vcc) pin using a VccH/2 resistor divider. Comparator will switch mux to Vbuck when present (2.9V, 3.6V, 4V, and 5V). Clear J8 to force mux to stay in auto mode (switch to higher of Vbus or Vbuck, to force Vbus to be selected when present). Only do this if Vbuck is 4V or lower. It is OK for comparator IN and OUT pins to have a voltage present when un-powered, and OUT pin will be high impedance.

IC Voltages / UVLO / Current Draw / Input Levels

IC	Operating Range*	Abs Max*	UVLO	Current (sleep / typ / max)	Input Levels / Notes
D21	1.62V-3.63V	3.8V	Configurable	See datasheet	See datasheet
L21	1.62V-3.63V	3.8V	Configurable	See datasheet	See datasheet
C21	2.7V-5.5V	5.5V	Configurable	See datasheet	See datasheet
D51	1.71V-3.63V	3.8V	Configurable	See datasheet	See datasheet
Level Shifter A (VccL)	1.2V-3.6V	6.5V	--	3.5uA / 3.5uA / 10uA	Can be <= 5.5V, Input levels: 0.35Vcc, 0.65Vcc
Level Shifter B (VccH)	1.65V-5.5V	6.5V	--	included in above	B voltage must be greater than or equal to A
Motor Controller	2.7V-18V	20V	2.5V max rising	1uA / 1.6mA / 1.8mA	Input levels: 0.6V, 2.0V
Current Sense IC (Vdd)	3.0V-5.5V	6V	--	3.5uA / 300uA read, 450uA int / 900uA	Up to 32V input, Input levels: 0.8V, 2.0V
Host / Aux Switch (both switches)	1.8V-5.5V	6V	1.65V (1.55V-1.75V)	2uA / 53uA / 95uA Divider: 3.5uA@5V	Input levels: 0.5V-0.9V, 0.8V-1.4V (@1.8V-@5.5V)
Power Mux	2.8V(1.5V)-5.5V	6V	2.58V (1.3V) rising	1uA / 56uA / 200uA	One supply must be >= 2.8V, Levels: 0.7V, 2.0V
Mux Comparator	1.0V-5.5V	6V	--	0.2uA / 0.2uA / 0.5uA Divider: 6.1uA@5V	Divider: 3.4uA - 6.7uA (2.8V - 5.5V)
Accelerometer / Magnetometer	1.71V-3.6V	4.8V	--	6uA / 400uA@12Hz, 1.1mA@1.6KHz / ???	Input levels: 0.3Vcc, 0.7Vcc
Pressure / Temp-erature Sensor	1.7V-3.6V	4.8V	--	1uA / 3uA,12uA / ???	Input levels: 0.2Vcc, 0.8Vcc
SRAM	2.5V-5.5V	6.5V	2V max. bat switch	4uA (1uA VBat) / 3mA / 10mA	Input levels: 0.1Vcc, 0.7Vcc, 1V RAM retention
EEPROM	1.8V-5.5V	6.5V	--	2uA / 2mA? / 3mA	Levels: 0.25Vcc, 0.75Vcc
FLASH	2.3V-3.6V	4.1V	2.3V	2uA (10uA stby.) / 3mA (read) / 16mA	>= 20MHz, Input levels: 0.3Vcc, 0.7Vcc
Buck Converter	3V-17V	20V	2.8V	1.5uA / 17uA / 25uA Div.: 8.1uA@5V/3.6V	Div.: 6.2uA@2.9V/4V, 2uA @1.8V, Levels: 0.3V, 0.9V
Main 3.3V Reg.	2.0V-5.5V	6V	1.9V	8uA / 60uA / 130uA	8uA@0mA, 60uA@20mA
3.3Vsw Regulator	2.7V-5.5V	6V	--	0.1uA / 2mA / 4.5mA	Divider: 7.3uA (enabled)
Micro SD Card* (see datasheet)	2.7V-3.6V	3.6V?	?	50uA-200uA / 10mA-100mA / 150mA	Input levels: 0.25Vcc, 0.625Vcc, 250ms startup
Vbus Divider	2.8V-5.5V	6V	--	7.7uA@5V	4.3uA-8.4uA (2.8V-5.5V)
VccH Divider	1.8V-5.5V	6V	--	7.7uA@5V	2.8uA-8.4uA (1.8V-5.5V)
Blue LED	2.9V-5.5V	6V	dim at 2.9V	0.6mA@3.3V	2.2mA@5V

** To reduce sleep current, MicroSD can be powered down when connected to 3.3Vsw rail.*

** Do not exceed operating range high value (except level shifter A). Absolute maximum for transients only.*

** For reference only. See relevant datasheets for electrical characteristics.*

RDSon / Dropout Table

Device	*RDSon (mOhms) or dropout (mV), (typ / max)*
Reverse polarity PFET (TJmax=150C)	4.4mOhm @Vgs=-4.5V / 8.3mOhm max @Vgs=-2.5V
Current Sense resistor	10mOhm
Buck converter FET (TJmax=125C)	High-side: 90mOhm / 170mOhm Low-side: 40mOhm / 70mOhm
Inductor DC resistance	35mOhm
Power Multiplexer (TJmax=125C)	84mOhm / 150mOhm
Host / Aux Power Switch (TJmax=125C)	70mOhm / 140mOhm
Motor controller H-Bridge (8 N-FETs, TJmax=125C)	High-side: 250mOhm, **400mOhm hot** / 500mOhm Low-side: 235mOhm, **400mOhm hot** / 500mOhm
Motor Controller Sense Resistors	180mOhm
Vmotor fuse (2.5A)	<42mOhm
3.3V Regulator (TJmax=125C)	65mV@100mA, 130mV@200mA, 163mV@250mA
3.3Vsw Regulator (TJmax=125C)	220mV@1.5A / 275mV, 375mV across temperature
VccH to Vmotor Schottky Diode (TJmax=175C)	300mV@250mA(25C) / 450mV@3.0A(125C)
Main 3.3V regulator input / Vbat Schottky Diodes (TJmax=150C)	240mV@1mA(25C), 365mV@1mA(-40C) / 480mV@250mA(150C)

** For reference only. See relevant datasheets for electrical characteristics.*

Thermal Derating

The circuit design and components were selected to support an ambient operating temperature range of -40C to 70C. All components are rated to at least 85C (higher for power components). This information is based on thermal simulation and limited testing. Further derating may be needed if combining these elements and/or radiative/convective cooling is restricted (ie: enclosure). All devices have thermal protection. They will power down if they get too hot, then power up after cooling down a small amount. Do not use this as a substitute for proper thermal control measures, as **excessive use of thermal protection degrades the device**. This section will be updated.

Main 3.3V Regulator (3.3V)

- **No derating up to 25C, derate down to 135mA at 70C.** Estimated J/A: 235C/W. Tjmax: 125C.
- 0.15A: 59C rise (0.25W) @5V. 0.25A 98C rise (0.42W) @5V. Did not include diode.
- Input schottky diode (TJmax=150C): No further derating required. Helps dissipate some power.
- Thermal derating not required at any temperature when using 3.6V or 4V from the buck converter.

High Current 3.3V Regulator (3.3Vsw)

- **Derate to 585mA at 25C, derate down to 324mA at 70C.** Estimated J/A: 100C/W. Tjmax: 125C.
- 0.5A: 85C rise (0.86W) @5V. 0.3A: 55C rise (0.52W) @5V. 0.6A: 105C rise (1.03W) @5V.
- Thermal derating not required at any temperature when using 3.6V from the buck converter.

Motor Controller (MOTOR)

- **Using one bridge: No derating up to 50C, derate down to 970mA at 70C.**
- **Using both bridges: Derate to 920mA each at 25C, derate down to 685mA at 70C.**
- Estimated J/A: 130C/W. Tjmax: 125C. Worst RDSon@5V: 450mOhm. 1.13A: 75C rise (0.575W).
- 180mOhm current sense resistors: 0.23W, 1.13A(full). **No derating required.**
- VccH-Vmotor diode: TJmax=175C. Limited by controller derating. **No further derating required**.

Buck Converter (BUCK)

- **Derating TBD:** Derating is more complicated with buck converters.
- **No derating up to 43C (100% PWM mode), derate down to TBD at 70C.**
- Estimated J/A: 120C/W. Tjmax: 125C. Worst RDSon (high-side): 170mOhm @6V input.

Power Multiplexer (MUX)

- **No derating up to 38C, derate down to 1.6A at 70C.** Estimated J/A: 145C/W. Tjmax: 125C.
- Worst RDSon@5V: 150mOhm. 2A: 87C rise (0.6W)

USB Host / AUX Switch (HOST)

- **No derating required @5V.** Estimated J/A: 165C/W. Worst RDSon@5V: 165mOhm. Tjmax: 125C.
- Single switch at 1A: 27C rise (0.165W); Both switches at 1A each: 55C rise (0.52W)
- At lower voltages the RDSon can be up to double (@1.8V), so derate accordingly (see datasheet).

Low Power Considerations

VccL current can be measured by clearing J19 (3.3V-VccL) and measuring across the 3.3V and VccL main header pins. Below are tips for saving power. This section will be updated.

Basic

Quiescent Current: TBD

- With the L21 or D51, use the internal Vddcore buck converter to significantly reduce power.
- When sensors are installed, the IMU will be driving INT1 low against 10Kohm (330uA @ 3.3V). Configure the IMU interrupt for open-drain, active-low operation to let the pin float (normally) and to match the other devices on INT1.

Standard

See also Basic tips. Quiescent Current: TBD

- The high current 3.3V regulator consumes 2mA when enabled. Disable to reduce it to <0.1uA.
- Micro SD cards can consume 10's of uA when not selected. To reduce sleep current further, connect the Micro SD (Vsd) to the 3.3Vsw rail so that it can be powered down.
- Motor Controller: Drive the MOPS (B17) pin low to enter sleep.
- Current Measurement IC: Set J14 to control the _READ_/INT pin using CMRI (A19). There is no pull resistor, so this pin should not be left floating. Set to read mode (drive low) so that it can enter sleep after Tsleep (1.14s). A register can be written via I2C to enter sleep immediately.

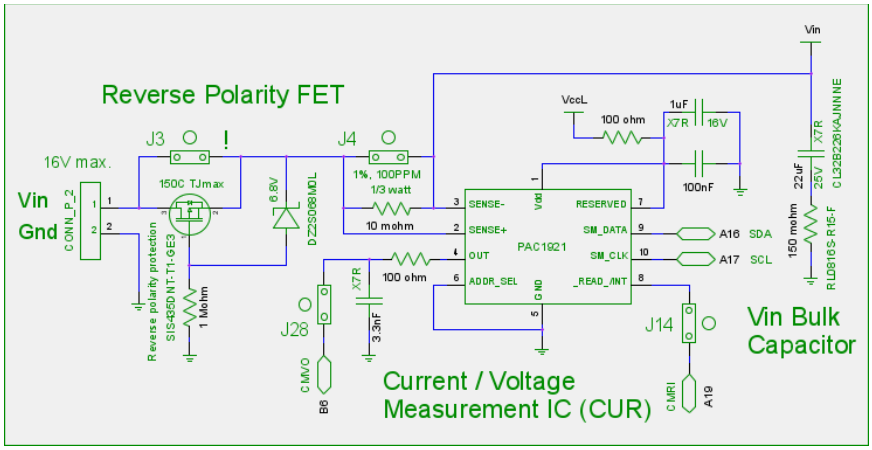
Advanced

See also Standard and Basic tips. Quiescent Current: TBD

- Buck Converter: Always drive BKFS (otherwise, it is floating). Use low-frequency (drive high) to save power. If VccH does not need to be 5V, lower Vbuck to 4V or 3.6V (best) so that the regulators have a lower voltage drop and thus dissipate less power. When VccL is connected directly to VccH, consider using an even lower Vbuck voltage.
- Power Multiplexer: Bypass the mux to save 55uA. Disable the comparator to save 6uA (@5V).
- USB Host / AUX Switch: Disable both switches to save ~47uA, plus whatever the downstream device was consuming. Note that if the USB cable is grounding the ID pin, then overriding the low will cost ~330uA.
- Do not connect the VccH voltage divider, which consumes 7.7uA @5V VccH.

Vin Section / Current Measurement IC

The Advanced variant, and optionally the Standard variant (with the Vin option installed), include a high voltage input (up to 17V, note that the silkscreen is printed with 16V), a reverse-polarity protection P-FET, a power / current / voltage measurement IC, and a bulk capacitor. Vin connects to the buck converter input (which then ultimately supplies the VccH rail). The motor controller can be connected to Vin as well (otherwise, Vmotor is limited to 5.5V from VccH). Maximum current through Vin is 5.6A.



Power / Current / Voltage Measurement IC (CUR)

Features (from Microchip datasheet)

- Microchip PAC1921 for power, current, and voltage measurement of Vin
- I2C for configuration and reading result/accumulator registers. I2C Address: 1001100.
- DAC analog output (10-bit) for low latency and low power operations
- With power selected for DAC analog output (default), all 3 can be read from I2C
- DAC output 0-3V by default (can be 2V, 1.5V, or 1V),
- 3.0V to 5.5V supply voltage range (VccL)
- 10mOhm, 1/3watt current sense resistor
- 14-bit or 11-bit sigma-delta converter, Auto-Zero offset, 1% power accuracy
- 0-32V Vin voltage measure range (1.95mV resolution @ 14-bits), set to 0-16V range.
- 10A(+/-) Vin current measure range (610uA resolution @ 14-bits) using 1X gain. Set to 5A (2X gain, 305uA resolution @ 14-bits) or 2.5A (4X gain, 153uA resolution @ 14-bits).
- Average measurements, 1X-128X selectable binary gain
- Read/Integrate pin connected to CMRI (A19). Can override pin with I2C register bits.
- 450uA (900uA max) integrate, 300uA (450uA max) read current
- 35uA Vin bias current (1uA in shutdown)
- Auto-sleep with 3.5uA (15uA max) sleep current (86us wake time)
- 40ms powerup delay
- -40C to 85C operating temperature

Hardware Settings / Usage

- Arduino library coming soon.
- **Maximum Current:** 5.6A maximum through Vin section current sense resistor (70C ambient), which is 10mOhm, 100ppm, 1/3W
- **Power States:** Controlled via _READ_/INT pin, I2C register bit, or automatically with free-run.
  - **Integrate State:** Takes measurements and adds values to accumulator.
  - **Read State:** This will stop integration, calculate results, place them in registers, and output the chosen value on the DAC pin. Automatically sleeps (DAC off) after Tsleep (1.14s typ.).
  - **Sleep State:** Tsleep (1.14s typical) after entering read state, device will enter sleep (3.5uA). The DAC is turned off. Device wakes when entering integrate state.
- **Measurement Modes:** Device starts in pin-controlled, power mode by default
  - **Power:** All measurement and result registers populated. Power output on DAC.
  - **Current (Vsense):** Free-run mode only. Only Vsense registers populated, output on DAC.
  - **Voltage (Vbus):** Free-run mode only. Only Vbus registers populated, output on DAC.
- **Pin Controlled** / **Free-Run:** Power can be either, but current and voltage are free-run only
  - **Pin Controlled:** Set J14 to control the _READ_/INT pin using CMRI (A19).
    - There is no pull resistor, so this pin should not be left floating.
    - Drive CMRI low to enter the Read state. Drive CMRI high to enter the Integrate state.
    - Integration time depends on how long pin is held high (2.7ms to 2.9s @ 14bit).
  - **Free-Run:** Integration time depends on number of samples (1 to 2048), filtering, measurement type, and resolution (1.41ms to 2.9s @ 14bit). Keep in integrate state.
- **Resolution:** 10bit from DAC or result registers. 14bit/11bit from accumulators (CPU average).
- **Gain:** Gain is digital, however, because the result registers are 10bit, gain (combined with a minimum number of samples) can be used up to a point to increase resolution.
- **I2C Interface:** Shared I2C bus on pins A16 (SDA) and A17 (SCL), both pins pulled to VccL by 4.7Kohm resistors. I2C also used by accelerometer/gyroscope and pressure/temp. sensor.
- **Analog Output:** Set J28 to connect the OUT pin to CMVO (B6) for ADC measurement. The DAC by default outputs from 0V to 3V (can be 2V, 1.5V, or 1V).

Reverse-polarity FET / Vin Bulk Capacitor

Vishay SiS435DNT P-Channel MOSFET (see datasheet)

- Used for reverse-polarity protection. Also helps control inductive voltage spiking.
- External gate protection using zener diode and resistor.
- RDSon: 4.4mOhm (5.4mOhm max.) @ Vgs=-4.5V, 6mOhm max. @ Vgs=-3.7V
- Can be bypassed by setting solder jumper J3, disabling reverse-polarity protection (see J3).
- Tjmax = 150C. No thermal derating required, even at 6A.

Vin Bulk Capacitor

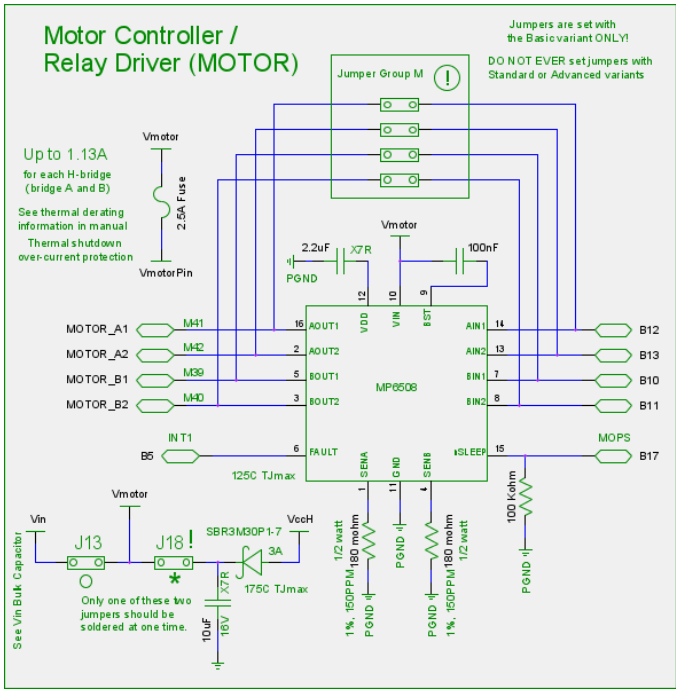
- Samsung CL32B226KAJNNNE 22uF, 25V, X7R ceramic, 1210 package
- 150mOhm series resistor adds ESR to help control inductive input voltage spiking
- Advanced variant: A 1uF X7R (without resistor) is installed on the buck converter input. It handles high-frequency components of the converter demand (bulk cap. handles the rest).

Motor Controller / Relay Driver (MOTOR)

The Standard and Advanced variants include a motor controller that can be used for two bi-directional DC motors (or four single direction), four relays (solenoids), or one bi-polar stepper motor. With the Vin option installed (always present on the Advanced), up to 17V is supported, with power, current, and voltage measurement. Otherwise, only ~5V from USB Vbus is used. With the Basic variant, the MCU pins are shorted with the header pins.

Features (from MPS datasheet)

- MPS MP6508
- 2.7V - 17V, 5V from VccH by default
- Optional Vin section allows up to 17V and includes current/voltage measurement IC. Otherwise, only ~5V from Vbus is used.
- Two internal full H-Bridge drivers
- All pins can be independently driven low
- Four relays, two DC motors (forward-reverse), or one bipolar stepper motor
- 1.2A per driver (two pins per driver)
- Internal current limit set to 1.13A each
- Two 180mOhm current sense resistors
- Low RDSon (~250mOhm HS and LS)
- Quiescent Current: 1.6mA
- Sleep Current: 1uA, pin controlled (B17)
- Over-current, UVLO, & thermal shutdown
- FAULT pin connected to board INT1 line
- 2.5A standard fuse on Vmotor header pin
- -40C to 85C ambient, 125C TJmax

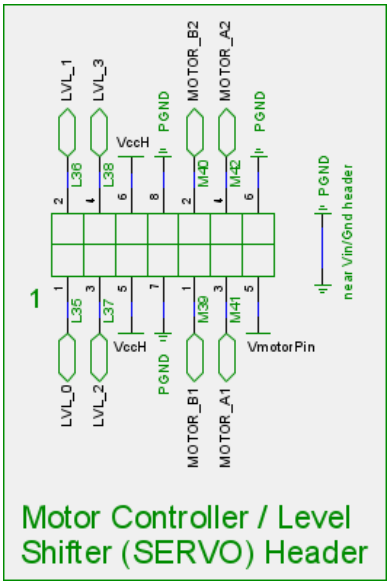


Hardware Settings

- With the motor controller option installed, all jumpers in Jumper Group M are clear. **Do not ever set these jumpers**, as doing so will short the inputs with the outputs (Basic variant only).
- By default, jumper J18 (VccH-Vmotor) is set and J13 (Vin-Vmotor) is clear, thus Vmotor operates at VccH minus the diode drop (300mV@250mA(25C) / 450mV@3.0A(125C)). If the Vin option is installed (always present with Advanced variant), then clear J18 and set J13 to operate at Vin (up to 17V). Do not set both jumpers. While this would facilitate auto-switching between VccH and Vin, reverse leakage current in the VccH-Vmotor diode can be excessive.
- A 2.5A fuse (NOT resettable) is installed on the Vmotor header pin. This pin should only be used to provide a common high voltage when independently driving outputs low (ie: relay use). If using externally, do not allow Vmotor to exceed the current limits for the diode or Vin section.
- **The nSLEEP pin is always connected** to MOPS (B17) of the MCU. It is pulled to ground by a 100K resistor, thus **sleep is enabled by default**. Drive MOPS high to enable the controller.

- The FAULT pin is also connected to the board INT1 line. This pin is open-drain active-low. It will drive low when there is an over-temperature fault. INT1 is also connected to the optional Accelerometer / Gyroscope (IMU, configurable events), and the Host / AUX switch (over-current, UVLO, over-temperature). **If the IMU is installed, then it must be configured first** before using INT1, otherwise contention may occur (see IMU chapter).
- See 'Vin Section / Current Measurement IC' chapter if option installed (included w/ Advanced).
- The motor controller requires thermal derating. The diode and current sense resistors do not require additional derating. **See 'Thermal Derating' section for more information.**
- Fuse blow time: 4 hours min. @ 2.5A, 100ms typical (5s max.) @ 5A, 200ms max. @ 7.5A

Firmware Settings



Motor Controller / Level Shifter (SERVO) Header

- Arduino Timer library coming soon (Oct. or Nov. 2017?)
- The motor controller logic supply is from an internal regulator (from VccH or Vin) and the inputs are TTL compatible, so any voltage supported by the MCU (VccL) can be used.
- PWM signals can be applied to control current. Additionally, current is also limited internally to 1.13A per bridge (based on the two 180mOhm current sense resistors). See motor controller datasheet.
- **Four Relays/Solenoids:** Each pin can be driven independently. Drive high to connect the corresponding output to ground (turn on). The other side of the relay connects to the Vmotor pin. Drive low to make the output to high-impedance.
- **Two or Four DC Motors:** With two motors, each motor connects to an H-bridge, for forward, reverse, brake, and coast support. Control speed with PWM. Use the table below. With four motors, connect each motor the same way as a relay (see above). Drive high to turn on. A single direction with coast is supported. Control speed with PWM.
- **One Stepper Motor:** A single bi-polar stepper motor is supported (full or half step). See motor controller datasheet.

Input / Output Table

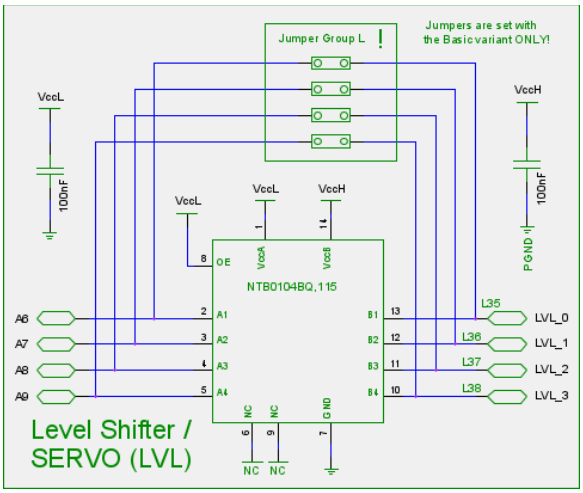
AIN1(B12) / BIN1(B10)	AIN2(B13) / BIN2(B11)	AOUT1 / BOUT1	AOUT2 / BOUT2
Low	Low	High-Impedance	High-Impedance
Low	High	GND	Vmotor
High	Low	Vmotor	GND
High	High	GND	GND

Level Shifter / Servo (LVL)

The Standard and Advanced variants include a bidirectional (with auto-direction sensing) level shifter that translates between four MCU pins on VccL (3.3V by default) and four header pins on VccH (5V by default), for timer, SERCOM, and digital functions. The header can accommodate two hobby servo connectors. With the Basic variant, the MCU pins (A side) are shorted with the header pins (B side).

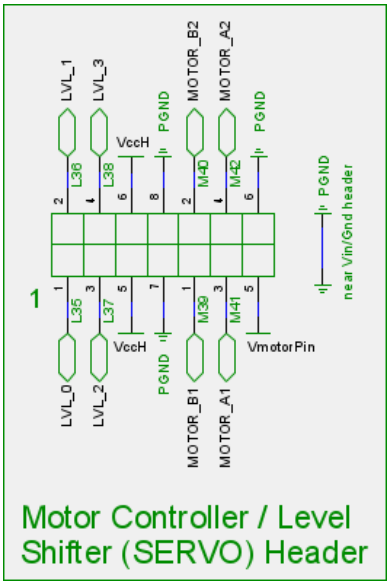
Features (from NXP datasheet)

- NXP NTB0104
- 1.2V – 3.6V (VccL), 1.65V – 5.5V (VccH)
- Translation between VccL (MCU) and VccH
- Bidirectional, auto direction sensing
- Input levels: 0.35Vcc, 0.65Vcc
- Default 3.3V to 5V translation useful for servos
- No special power sequencing required
- Static Current: 3.5uA typ.
- 15KV ESD (HBM) on B (VccH) pins
- -40C to 85C ambient, 125C TJmax



Hardware Settings

- With the level shifter option installed, all jumpers in Jumper Group L are clear. Do not set these jumpers, as doing so will short the inputs with the outputs (Basic variant only).
- B voltage must be greater than or equal to A (can be <= 5.5V)
  - When VccL and VccH are different, VccL is always 3.3V, but VccH can be 5V (default), 4V, or 3.6V.
  - When VccL = VccH, it can be 5V, 4V, 3.6V, 2.9V, or 1.8V.
- Device driving input must be at least 2mA. Check the drive strength register of the MCU if VccL < 3V (4.5V for C21).
- B side capacitive loads of up to 70 pF.
- B side pull resistors must be kept higher than 50KOhm.
- 40Ohm to 50Ohm output impedance when switching outputs, then held at high or low using 4KOhm pull resistor.
- Not for use in open drain driver applications such as I2C.
- It is OK for A side to be up to 5.5V (but not operational).



Firmware Settings

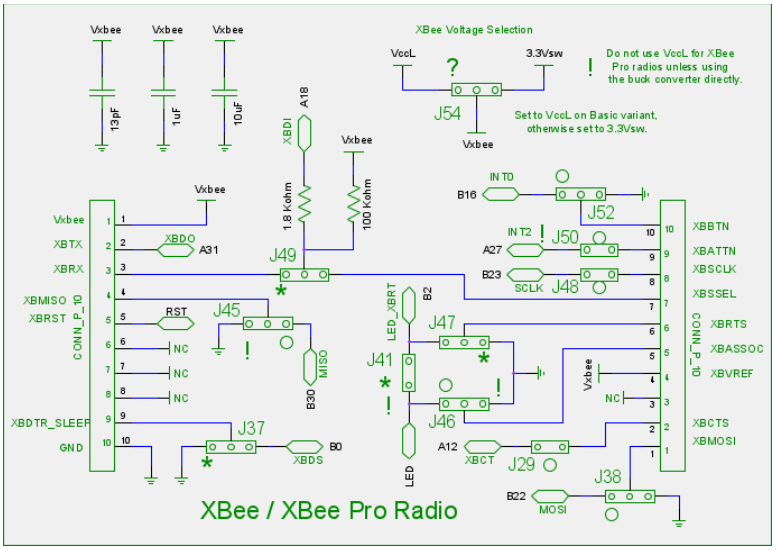
- Uses MCU pins A6 (L35), A7 (L36), A8 (L37), and A9 (L38). Arduino timer library coming soon.
- Two 5V hobby servos may be connected to pins 3 (PWM), 5 (VccH), and 7 (PGND), and pins 4 (PWM), 6 (VccH), and 8 (PGND). VccH must be set to 5V. Two additional servos can be connected to the other 2 level-shifted pins, with 5V and GND coming from the main header.

XBee / XBee Pro Radio Socket

All variants include an Xbee radio socket that supports Xbee (original form factor) radios. The Standard adds support for Xbee Pro radios (larger size, higher current), while the Advanced supports cellular radios (Pro size, highest current). UART (with optional CTS and RTS) or SPI are supported.

Features

- Supports Xbee, Xbee Pro form factors
- Many radios to choose from, including Wi-Fi, Cellular (LTE Cat 1, LTE-M, NB-IOT, LTE Cat 3, 3G), Bluetooth, GPS, 900MHz, Digi ZigBee and DigiMesh
- UART or SPI (with ATTN interrupt)
- UART supports RTS and CTS
- ASSOC LED and button support
- SLEEP / DTR pin support
- Vxbee can connect to VccL or 3.3Vsw
- Optional low-profile headers



Hardware Settings

- **Read 'Installing Headers' first.**
- **Radio Type:** Standard, Pro, or Cellular (Pro form factor with high current demands)
  - **Xbee Standard:** Supported on all board variants, connected to either VccL or 3.3Vsw.
  - **Xbee Pro:** Supported on the Standard and Advanced, connected to 3.3Vsw or, with the Advanced, to VccL when the buck converter is used directly.
  - **Xbee Cellular:** Supported on the Advanced only, connected to VccL when the buck converter is used directly (in some cases 3.3Vsw may be used with buck converter 3.6V).
- **Power Source:**
  - **3.3Vsw:** Default setting with Standard and Advanced. Higher current, can be powered off.
    - When turning off 3.3Vsw, do not drive or pull any lines high that are routed to devices powered by 3.3Vsw (though this is not required when using UART, see below).
  - **VccL:** Default setting with Basic. Source is from main 3.3V regulator or buck converter (Advanced). To use the buck converter, see 'Power Supply Configurations'.
- **Communication Mode:** UART (default) or SPI
  - XBDI used for XBRX (UART) or XBSSEL (SPI). It has a 100K pull resistor to Vxbee.
  - **UART (default):** Transparent or API mode.
    - Set J49 towards the asterisk to connect XBDI to XBRX. XBTX always connected.
    - CTS and RTS: Connect one or both for flow control. See J29 and J47.
    - 1.8K series resistor limits current when 3.3Vsw is off and XBDI (MCU TX) driving high.
    - The D51 requires two jumper wires (1 wire if using SPI), see 'Special Notes for D51'.
  - **SPI:** Uses API mode only. Higher clock speeds than UART. SPI Mode 0.
    - Connect the shared SPI bus MOSI, MISO, and SCLK to the radio (see J38, J45, and

J48). Set J49 away from the asterisk to connect XBDI to XBSSEL. Set J50 to connect XBATTN to shared INT2 line so the radio can notify the MCU it has data. Because XBATTN is push-pull, **do not use other devices on INT2**.
    - D51 requires 3 wires from main header to SPI header. See 'Special Notes for D51'.
    - CONFIG pin (XBRX, DIN) w/o pullup OK when in SPI mode (during powerup). All pullups should be enabled by default in radio (but check to be sure).
- **Button and LED:** The Xeno button and LED can connect to the Xbee radio instead of the MCU. See J52 for the commissioning button and J46 & J41 for the associate LED.
- **Sleep/DTR:** For pin sleep (or use commands), or for DTS (programmable radios). See J37.
- **Skywire Modem Support:** These modems require several additional pins to be grounded. See 'Pin Use' and 'Skywire Cellular' columns of 'XBee Radio Header Pins' table.
- **Arduino Libraries:** There is an API mode library available (but I have not tested yet).

XBee Radio Header Pins

Pin	Pin Use	Most Digi Radios*	Digi Cellular	Skywire Cellular	GPS Bee	BLEBee v2.0.0	Bluetooth Bee
1	+	VCC	VCC	VCC	Vcc	3.3V	Vcc
2	D	DOUT	DOUT	DOUT	TX	TX	TX
3	D	DIN/CONFIG	DIN/CONFIG	DIN	RX	RX	RX
4	S, K	SPI_MISO	DIO12	GND	NC	NC	NC (P2_2)
5	+	RESET	RESET	RESET	NC (EX)	RESET	RESET
6	--	Unused	Unused	Unused	Unused	Unused	Unused
7	--	Unused	Unused	Unused	Unused	Unused	Unused
8	--	Unused	Unused	Unused	Unused	Unused	Unused
9	O	DTR/SLEEP	DTR/SLEEP	DTR	NC	NC	NC (P0_1)
10	+	GND	GND	GND	GND	Gnd	GND
11	S, K	SPI_MOSI	DIO4	GND	NC	NC	NC (P1_5)
12	O	CTS	CTS	CTS	NC	NC	CT
13	--	Unused	Unused	Unused	Unused	Unused	Unused
14	+	VREF (NC on some)	VREF	VREF	NC	NC	NC (P1_2)
15	O, K	ASSOCIATE	ASSOCIATE	GND	TP (1Hz-4Hz)	ASSOC	NC (P1_3)
16	O	RTS	RTS	RTS	NC	NC	RT
17	S	SPI_SSEL	DIO3	DIOx	NC	CH3	NC (P1_1)
18	S	SPI_CLK	DIO2	DIOx	NC	CH2	NC (P1_0)
19	O	SPI_ATTN	DIO1	ADC1	SCL	CH1	USB_N
20	O, K	CB	DIO0	PWR_ON	SDA	CH0	USB_P

*Pin Use Legend: + = Always connected, -- = Never connected, D = By default, this is connected, O = Can optionally be connected, S = Alternate SPI configuration, K = Must be GND for Skywire modems (see J52, J46, J45, J38). * = no SPI on DigiMesh*

XBee Radio Voltages / Current consumption

Radio	Form / Variant*	Voltage Range	Current* (sleep / typ / max)	Input Levels / Notes
Digi Pro 900HP XBP9B-Dxxx-xxx	Xbee Pro Standard	2.1V-3.6V	2.5uA / 29mA RX, 215mA TX / 290mA	Input levels: 0.3Vcc, 0.7Vcc, -40C to 85C, <50mVp-p on Vcc
Digi WiFi XB2B-WFxx-xxx	Xbee Pro Standard	3.14V-3.46V	6uA (2mA) / 100mA RX, 309mA TX / ???	Input levels: 0.3Vcc, 0.7Vcc, -30C to 85C, <50mVp-p on Vcc
Digi Pro DigiMesh XBP24-DMxxx-xxx	Xbee Pro Standard*	2.8V-3.4V	<1uA / 31mA(45mA) RX, 120mA TX / ???	Input levels: ??, -40C to 85C, <50mVp-p on Vcc
Digi Pro 802.15.4 XBP24-Axx-xxxx	Xbee Pro Standard*	2.8V-3.4V	<1uA / 31mA(45mA) RX, 120mA TX / ???	Input levels: ??, -40C to 85C, <50mVp-p on Vcc
Digi Pro ZigBee XBP24CZ7xxx-xxx	Xbee Pro Standard*	2.7V(2.2V)-3.6V	<1uA / 31mA(45mA) RX, 120mA TX / ???	Input levels: ??, -40C to 85C, <50mVp-p on Vcc
Digi LTE Cat 1 XBC-V1-UT-001	Xbee Pro Advanced*	3.0V-5.5V (2.7V-5.5V)	??? / 143mA idle (860mA RX+TX) / 1A	Input levels: 0.3Vcc, 0.7Vcc, -40C to 85C, lower current when using 5V
Digi LTE-M XBC-V2-UT-001 XBC-A2-UT-001	Xbee Pro Advanced*	3.0V-4.3V	TBD / TBD / TBD	Input levels: ??, -40C to 85C, Verizon (V2) and ATT (A2)
Digi 3G Global XBC-M5-UT-001	Xbee Pro Advanced	2.7V-5.5V	10uA / 87mA Idle, 224mA RX, 702mA TX	Input levels: ??, -40C to 85C, lower current when using 5V
Skywire LTE M1 NL-SW-LTE-SVZM20	Xbee Pro Advanced*	3.14V-5.5V	TBD / TBD / TBD	Input levels: ??, -40C to 85C
Skywire LTE Cat 1 NL-SW-LTE-WM14 NL-SW-LTE-GELS3	Xbee Pro Advanced	3.0V-5.5V	1uA / 60mA idle, 150mA avg. 600mA (active cell) / 1.5A	Input levels: 0.15Vcc, Vref-0.4V, -40C to 85C (-25C to 75C ATT), Verizon (GELS) & ATT (WM)
Skywire LTE Cat 3 NL-SW-LTE-Txxx	Xbee Pro Advanced	3.5V-4.3V	1uA / 750mA / 2A	Input levels: 0.15Vcc, Vref-0.4V, -40C to 85C, GPS / GLONASS
Skywire 3G HSPA NL-SW-HSPAxx	Xbee Pro Advanced	3.5V-4.3V	1uA / 580mA / 2A	Input levels: 0.15Vcc, Vref-0.4V, -40C to 85C, GPS option
Digi DigiMesh XB24-DMxxx-xxx	XBee Basic	2.1V-3.6V	<1uA / 28mA RX, 33mA TX / ???	Input levels: ??, <50mVp-p on Vcc, -40C to 85C, up to 45mA w/ boost
Digi 802.15.4 XB24CAxxx-xxx	XBee Basic	2.1V-3.6V	<1uA / 28mA RX, 33mA TX / ???	Input levels: ??, <50mVp-p on Vcc, -40C to 85C
Digi ZigBee XB24CZ7xxx-xxx	XBee Basic	2.1V-3.6V	<1uA / 28mA RX, 33mA TX / ???	Input levels: ??, <50mVp-p on Vcc, -40C to 85C, programmable variants
GPS Bee (SIM28)	XBee Basic	2.8V-3.6V	200uA / 19mA / 100mA	Input levels: 0.8V,2.0V, 24mA Acquisition, -40C to 85C
BLEBee v2.0.0 (BLE113)	XBee Basic	2.0V-3.6V	<1uA / 26mA / ???	Input levels: 0.5V, 2.5V, <10mVp-p on Vcc, -40C to 85C
Bluetooth Bee (CC2540F256)	XBee Basic	2.0V-3.6V	<1uA / 26mA / ???	Input levels: 0.5V, 2.5V, <10mVp-p on Vcc, -40C to 85C

** Use PR command to set pullups on unused pins to minimize sleep current. Use pin sleep for even lower current in some cases. To reduce further, Xbee radio can be powered down when connected to 3.3Vsw.*

** These radios can be used with a lower board variant (one below) if current/temperature is limited.*

Memory Device (MEM)

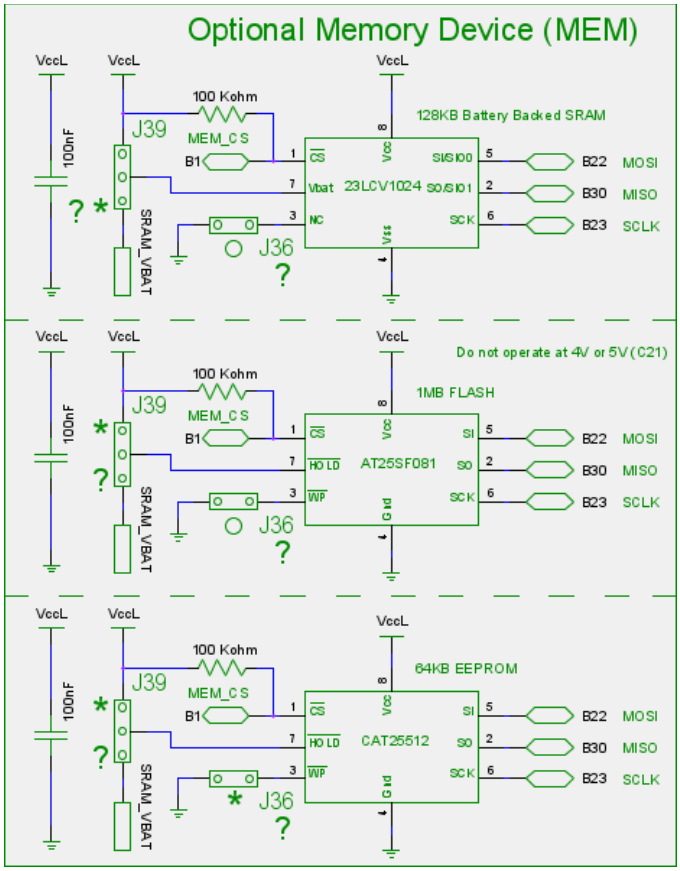
A serial memory device can optionally be installed, with the choice of 128KB SRAM, 1MB FLASH, or 64KB EEPROM. The device is connected to the microcontroller via the shared SPI bus on pins B22, B23, B30, and B1 (MEM_CS, which is pulled to VccL with a 100Kohm resistor). The SPI bus is also routed to the Micro SD card slot, the SPI header, and optionally to the XBee radio. Note that due to limitations of the SPI peripheral, **12MHz is the maximum clock speed**. All devices are compatible with Mode 0. The SRAM Vbat pin can be connected to the coin cell.

The D51 requires a jumper wire to use SPI, see 'Special Notes for D51'.

128KB SRAM with Battery Backup

Features (from Microchip datasheet)

- Microchip Technology Inc. 23LCV1024
- 2.5V – 5.5V Supply Voltage Range
- External Battery Backup Support from Vin Pin
- Read Current: 3 mA at 5.5V, 20 MHz
- Standby Current: 4 uA at +85°C
- Unlimited Read and Write Cycles, Zero Write Time
- 128K x 8-bit Organization: 32-byte page
- Byte, Page and Sequential mode for Reads and Writes
- Arduino library available now
  - See 'SRAM_23LC' in Arduino Libraries Manager.



Hardware Settings

- Solder Jumper J39 (3-pad) must be set to SRAM_VBAT (pad near ?)
  - A coin cell up to 3.5V may be installed in the optional CR2032 holder
  - When Vcc is less than about 1.8V, switchover to the battery will occur (drawing ~1uA)
  - SRAM contents will be kept with Vbat as low as about 1V
- Solder Jumper J36 must be cleared (the associated 23LCV1024 pin is NC)

1MB FLASH

Features (from Adesto datasheet)

- Adesto® AT25SF081
- 2.3V – 3.6V, **Do not operate at 4V or 5V** (the C21 supports 2.7V to 5.5V)
- Uniform 4-Kbyte, 32-Kbyte, or 64-Kbyte Block Erase; Full Chip Erase
- Three protected, programmable security register pages
- Byte/Page Program (1 to 256 Bytes)
- 0.7ms Typical Page Program (256 Bytes) Time
- 70ms Typical 4-Kbyte Block Erase Time (300ms for 32KB and 600ms for 64KB)
- 2µA Deep Power-Down current (Typical), 10uA Standby
- 4mA Active Read Current (Typical)
- Endurance: 100,000 Program/Erase Cycles, Data Retention: 20 Years
- Arduino library coming soon

Hardware Settings

- Solder Jumper J39 (3-pad) must be set to VccL (pad away from ?) which will disable _HOLD_
- Solder Jumper J36 should be set, which will enable _WP_
  - Firmware must also enable write protection, thus writes can still be made with _WP_ low
  - If J36 is not soldered, _WP_ will be pulled high by an internal resistor

64KB EEPROM

Features (from ON Semiconductor datasheet)

- ON Semiconductor CAT25512
- 1.8V to 5.5V Supply Voltage Range, Low Power CMOS Technology
- 128–byte Page Write Buffer, Self–timed Write Cycle
- Additional Identification Page with Permanent Write Protection
- Block Write Protection, Protect 1/4, 1/2 or Entire EEPROM Array
- 4,000,000 Program/Erase Cycles, 200 Year Data Retention
- Arduino library available now
  - See 'EEPROM_CAT25' in Arduino Libraries Manager.

Hardware Settings

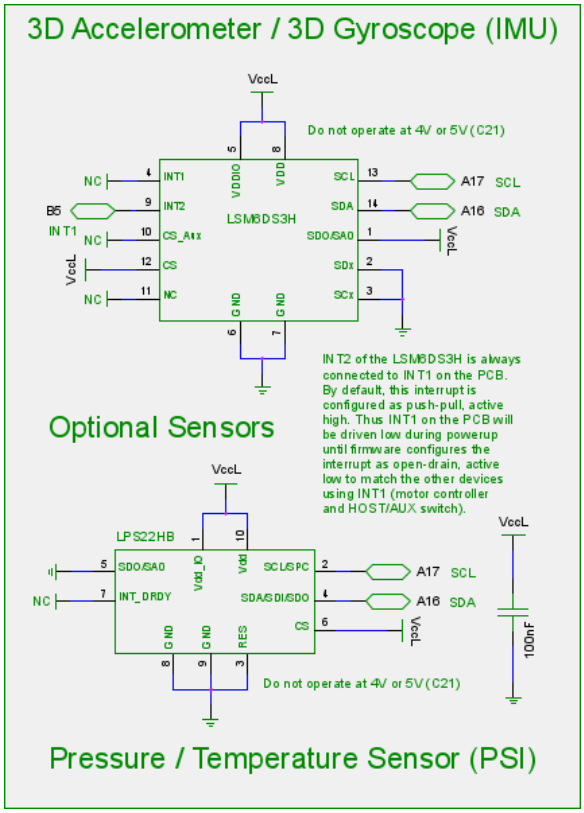
- Solder Jumper J39 (3-pad) must be set to VccL (pad away from ?) which will disable _HOLD_
- Solder Jumper J36 should be set, which will enable _WP_
  - Firmware must also enable write protection, thus writes can still be made with _WP_ low
  - J36 must not be cleared. There is no internal pullup resistor, so the _WP_ pin will float.

Sensors

Two sensor devices can optionally be installed for all board variants, which includes a 3D Accelerometer / 3D Gyroscope (IMU) and a Pressure / Temperature Sensor (PSI). The devices are connected to the microcontroller via the shared I2C bus at up to 400KHz. The I2C bus is also routed to the main header pins, to the I2C header, and to the Current Measurement IC (CUR, if installed). The IMU INT2 pin is also connected to the board INT1 line, with Data Ready, FIFO flags, pedometer, and several special function events available.

3D Accelerometer / 3D Gyroscope (IMU)

Features (from ST datasheet)



- ST Microelectronics LSM6DS3HTR
- 1.71V - 3.6V Supply Voltage Range
- Accelerometer ODR up to 6.66KHz
- Gyroscope ODR up to 1.66KHz
- 16-bit data output
- ±2/±4/±8/±16 g full scale
- ±125/±245/±500/±1000/±2000 dps full scale
- Smart FIFO, 4KB + 4KB (configurable)
- “Always-on” experience with low power
- Supply Current: 1.1mA@1.6KHz, 0.85mA@208Hz, 0.4mA@12.5Hz, 240uA@1.6KHz accel. only
- Standby Current: 6uA
- Supports I2C up to 400KHz
- INT2 pin connected to board INT1 line
  - Data Ready, FIFO flags, pedometer, and several special function events available
- -40C to 85C operating temperature

Hardware Settings

- **Do not operate at 4V or 5V** (C21 is 2.7V to 5.5V)
- Uses shared I2C bus on pins A16 (SDA) and A17 (SCL), with both pins pulled to VccL with 4.7Kohm resistors by default. I2C also used by pressure/temp. sensor and current measurement IC.
- The INT2 pin is also connected to the board INT1 line. **This pin is push-pull, active-high by default!** The pin will be driving low after power-up, which will not cause contention, but will mask real interrupts from two other possible sources on INT1, the motor controller (over-temperature fault), and the Host / AUX switch (over-current, UVLO, over-temperature). They are both open-drain, active-low. Thus, the IMU INT2 pin must be configured as open-drain, active low **first before using the IMU interrupt,** otherwise contention may occur.

Firmware Settings

- Arduino libraries available now
  - See 'SmartEverything LSM6DS3' and 'SparkFun LSM6DS3 Breakout' in Arduino Libraries Manager.
- I2C Address: 0x6B (0b1101011)

Pressure / Temperature Sensor (PSI)

Features (from ST datasheet)

- ST Microelectronics LPS22HBTR
- 1.71V - 3.6V Supply Voltage Range
- 260 to 1260 hPa absolute pressure range
- High overpressure capability: 20x full-scale
- Embedded temperature compensation
- 24-bit pressure data output
- 16-bit temperature data output
- ODR from 1 Hz to 75 Hz
- Embedded FIFO, 30 slots of 40-bit data
- Supply Current: 12uA low-noise mode (3uA low-current mode)
- Standby Current: 1uA
- Supports I2C up to 400KHz
- -40C to 85C operating temperature

Hardware Settings

- **Do not operate at 4V or 5V** (the C21 supports 2.7V to 5.5V)
- Uses shared I2C bus on pins A16 (SDA) and A17 (SCL), with both pins pulled to VccL by 4.7Kohm resistors. I2C also used by accelerometer/gyroscope and current measurement IC.

Firmware Settings

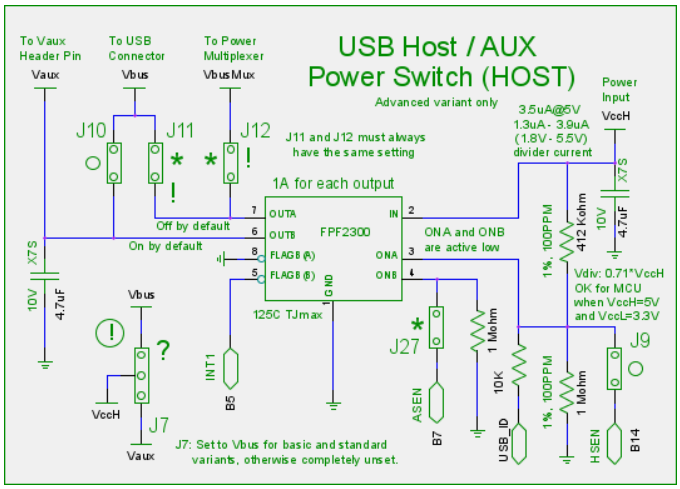
- Arduino library coming soon
- I2C Address: 0x5C (0b1011100)

USB Host / Aux Power Switch (HOST)

A dual-output switch with one amp output current per switch is installed on the advanced variant. By default, one switch output is connected to Vbus, and the other is connected to the Vaux header pin. The switch connected to Vbus can be used to output 5V from the buck converter to the Micro USB connector Vbus line for host mode applications. The Vbus switch is controlled automatically by the USB ID pin, or it can be controlled manually using the HSEN line (B14, disconnected by default). The Vaux switch can be controlled by the ASEN line (B7, disconnected by default). It is possible to tie both outputs together for up to two amps output on either Vbus or Vaux. See 'Thermal Derating' section.

Features (from Fairchild datasheet)

- Fairchild Semiconductor FPF2300
- Two 1A outputs, Slew rate controlled
- 1.3A (1.1A min.) over-current threshold, with 10ms blanking and 504ms auto restart.
- 1.8V to 5.5V supply voltage range
- RDSon = 75mOhm (140mOhm max.)
- Reverse current blocking when disabled
- Fault reporting on AUX output using board INT1 line (over-current, UVLO, over-temp.)
- 53uA (95uA max.) quiescent current
- 3uA max. shutdown current
- 1.65V UVLO, 4KV HBM ESD
- -40C to 85C ambient, thermal protection



USB Host (OUTA/ONA)

- ONA is controlled automatically by USB_ID through a 10K resistor
  - A USB Micro OTG cable will ground USB_ID if host side of cable is connected
- ONA is pulled to 0.71*VccH, thus OUTA is **off by default.**
  - 3.54V @ 5V VccH, overvoltage is well within injection current spec of MCUs @ 3.3V
  - Divider ratio allows VccH to be 1.8V to 5V, and to support the C21 @ 5V
  - Divider current is 1.3uA (1.8V) to 3.5uA (5V)
- ONA can be controlled manually using HSEN (B14) by soldering J9
  - Bring HSEN low to turn on. Pullup current is VccH / 412Kohm (12uA @ 5V VccH).
  - Bring HSEN high to turn off. If overriding USB_ID, current is VccL / 10K (~330uA).
- ONA pin state can be read using HSEN (B14) by soldering J9
  - Useful to determine if host side of OTG cable is plugged in
- OUTA by default is routed to the Micro USB connector Vbus pin through J11
  - When used as a device, Vbus can still power the Xeno through J11 and J12 to the power multiplexer. Reverse current to VccH is blocked when ONA is disabled (see notes below).

AUX Output (OUTB/ONB)

- ONB by default is controlled manually using ASEN (B7)
  - ONB is pulled to ground through a 1Mohm resistor, thus OUTB is **on by default**.
  - Bring ASEN low to turn on.
  - Bring ASEN high to turn off. Pulldown current is VccL / 1Mohm (3.3uA @ 3.3V VccL).
  - Disconnect J27 to disable ONB control and thus use pin B7 for other purposes. OUTB will always be on in this case, and will still provide over-current protection.
- OUTB is always connected to the AUX main header pin (with a 4.7uF output capacitor)
  - OUTB can be tied to USB Vbus through J10 (disconnected by default). See below.

USB Host and AUX Tied Together

- Set J10 to tie USB Vbus and Vaux together. Do not set J10 if Vbuck directly powers VccH.
  - As an output, this increases current to 2A (USB Host mode or Vaux). Output available only when both power mux and buck converter enabled. Set HSEN and ASEN to same value.
  - As an input, the Vaux pin can be used along with adjacent D-, D+, and Gnd for a panel-mount connector. Input available in all cases J10 is allowed to be set. See table below.

FLAGB Interrupt

- FLAGB(B) is always connected to the shared board INT1 (B5) line. It signals over-current, UVLO, or over-temperature on the AUX output by driving low. The accelerometer / gyroscope INT and motor controller Fault pins are also connected to INT1. See 'INT1' in 'Other Hardware'.
- If accelerometer / gyroscope installed, it must be configured before using anything on INT1.

Special notes regarding reverse current

- Do not allow the output voltage of the switch (ie: Vbus) to exceed the input by more than a small amount (300mV? This can be handled by reverse current protection). This could happen unintentionally due to a fault (VccH at 0V after buck fault, but Vbus still present). To avoid this possibility, use only configurations in the table below. **See 'Power Supply Configurations'.**

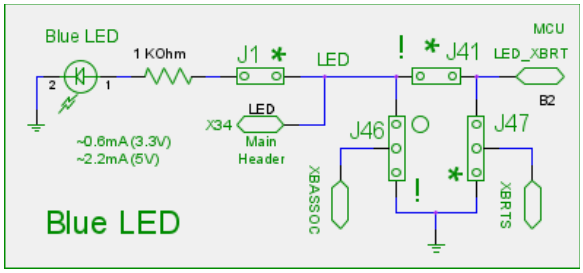
Advanced Power Configurations Table*

VccL-VccH	3.3V-VccL	3.3Vsw-VccL	Power Mux			USB Host / AUX			
J32	J19	J51	J8	J20	J21 (3-pad)	J7 (3-pad)	J10	J11	J12
OFF	ON	OFF	CHOOSE	ON	MUX_OUT - VccH	ALL OFF	CHOOSE	ON	ON
OFF	ON	OFF	OFF	OFF	VccH - Vbuck	ALL OFF	OFF	OFF	OFF
OFF	ON	OFF	OFF	OFF	ALL OFF	VccH - Vbus	CHOOSE	OFF	OFF
ON	OFF	CHOOSE	OFF	OFF	VccH - Vbuck	ALL OFF	OFF	OFF	OFF
ON	OFF	CHOOSE	OFF	OFF	ALL OFF	VccH - Vbus	CHOOSE	OFF	OFF
ON	OFF	CHOOSE	CHOOSE	ON	MUX_OUT - VccH	ALL OFF	CHOOSE	ON	ON

** Please read the corresponding section under 'Power Supply Configurations' → 'Advanced'.*

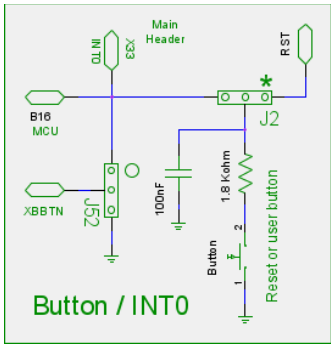
Other Hardware

Blue LED



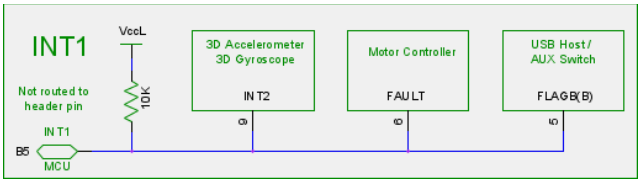
- Connected to LED_XBRT pin (B2, X34 on main header) through jumpers J41 and J1 by default.
- Drive LED_XBRT high to turn on.
- Forward voltage = 2.85V typ. (use VccL >= 3.3V)
- ~0.6mA @ 3.3V (~2.2mA @ 5V).
- Xbee radio ASSOC pin can instead drive the LED by setting J46 (circle side) and clearing J41. LED_XBRT can then optionally be connected to the Xbee radio RTS line (which by default is grounded) by setting J47 (opposite the asterisk). See Solder Jumpers section for more details.

Button (Reset / INT0)



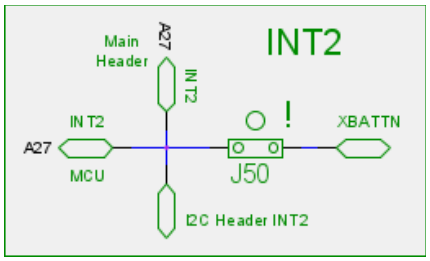
- Connected to RST by default through jumper J2. The reset button can be pressed twice in quick succession to enter the SAM-BA bootlader. There is a 10Kohm pullup resistor to VccL on RST.
- Button can connect to INT0 pin (B16, X33 on main header) instead by setting J2 (opposite the asterisk) and enabling the internal pullup.
- Pressing the button brings RST / INT0 low.
- 1.8K series resistor and 100nF capacitor installed for debouncing.
- Xbee radio BTN pin can also be connected to INT0 by setting J52 (circle side). See Solder Jumpers section for more details.

INT1



- INT1 (B5) not routed to a header pin
- Pulled up to VccL through 10Kohm
- Connects to pin 6 (FAULT) of Motor Controller for over-temperature (Standard & Advanced).
- Connects to pin 5 (FLAGB(B)) of USB Host / Aux Switch to signal over-current, UVLO, or over-temperature (Advanced only).
- Connects to pin 9 (INT2) of optional Accelerometer / Gyroscope. **This pin is push-pull, active-high by default!** The pin will be driving low after power-up, which will not cause contention, but will mask real interrupts from the motor controller and the Host / AUX switch, which are both open-drain, active-low. Thus, the IMU INT2 pin must be configured as open-drain, active low **first before using the IMU interrupt,** otherwise contention may occur.

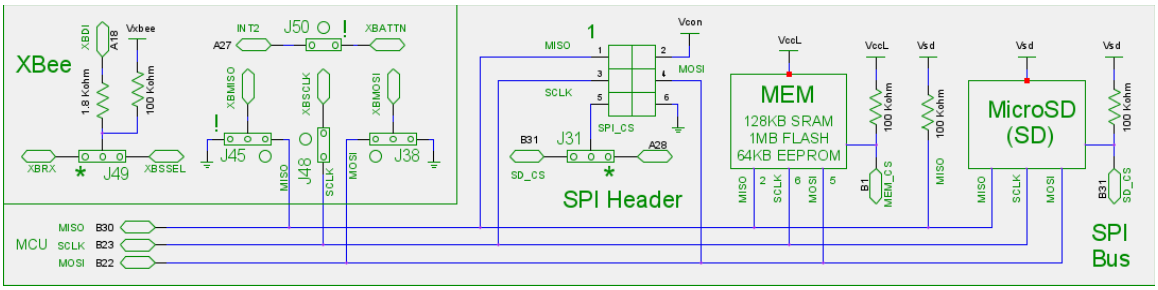
INT2



- INT2 (A27) requires internal pull resistor
- Always connects to main header and I2C header (INT2)
- Optionally connects to Xbee radio ATTN by setting J50. This is available when SPI mode is used. **Because XBATTN is push-pull, do not use other devices on INT2.**

SPI Bus

- Shared SPI bus is always connected to Micro SD, SPI header, and optional memory device.
- Can optionally connect to Xbee radio. See 'Xbee Radio' and 'Solder Jumpers' for details.
- SPI header power (Vcon) can be from VccL or 3.3Vsw (can be turned off). The chip select can be from A28 (not on L21/D51) or B31 (SD card cannot be used). See jumpers J43 and J31.
- 12MHz is the maximum clock speed. All devices are compatible with Mode 0.
- **The D51 requires 3 jumper wires from the main header to the SPI header. See 'Special Notes for D51'.**

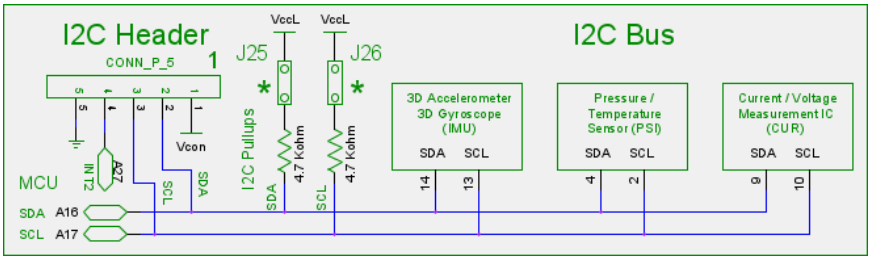


SPI Chip Select Table

Device	MCU Pin
Micro SD Card Slot (SD)	B31 (SD_CS), can be routed to SPI header
Memory Device (MEM)	B1 (MEM_CS)
SPI Header	A28 (except L21 or D51) or B31 (can no longer use SD card)
Xbee Radio	A18 (XBDI)

I2C Bus

- Connects to Current / Voltage Measurement IC (CUR) and 2 sensor ICs.
- Two 4.7Kohm pullup resistors to VccL through J25 (SDA) and J26 (SCL).

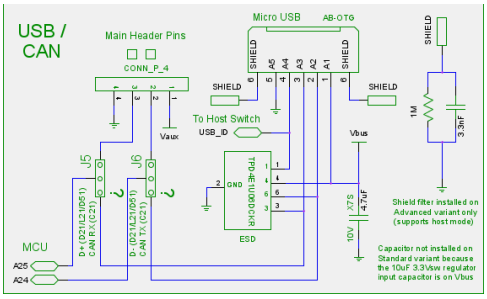


- I2C header with SCL, SDA, and INT2, with power (Vcon) from VccL or 3.3Vsw (see J43).

I2C Address Table

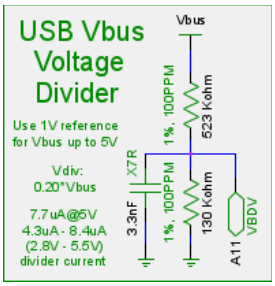
Device	Address
Current / Voltage Measurement IC (CUR), with Vin option	0x4C (0b1001100)
3D Accelerometer / 3D Gyroscope (IMU), optional	0x6B (0b1101011)
Pressure Sensor / Temperature Sensor (PSI), optional	0x5C (0b1011100)

Micro USB Connector



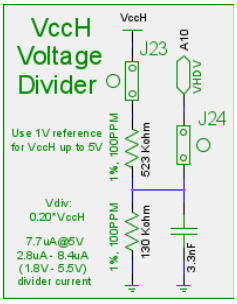
- Vbus (~5V) can power the board (all variants).
- ESD protection on D+, D-, and Vbus. D+/D- pullup will not trigger false Vbus detection.
- Vbus, D-, D+, and Gnd available on main header for panel mount connector use. See J5, J6, and J10.
  For the C21, these pins are routed to the CAN interface.
- Host mode support (Advanced). USB_ID / B14 (HSEN) controls Vbus (5V) output through Host/AUX switch.
- Shield to ground filter installed (Advanced).

Vbus Measurement (Advanced only)



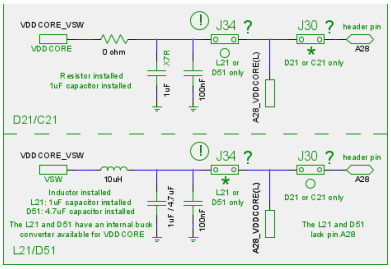
- Voltage divider with 523Kohm to USB Vbus, 130Kohm to ground, and a 3.3nF capacitor (to use SAR converter with high resistances).
- Always connected to A11 (VBDV).
- Voltage: 0.20*Vbus, Use 1V reference for Vbus up to 5V.
- Current consumption: 7.7uA@5V, 4.3uA - 8.4uA (2.8V - 5.5V).

VccH Measurement (Advanced only)



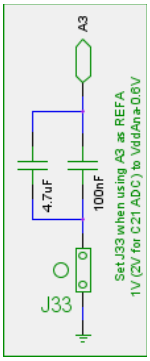
- Voltage divider with 523Kohm to VccH, 130Kohm to ground, and a 3.3nF capacitor (to use SAR converter with high resistances).
- Connected to A10 (VHDV) through J24 (disconnected by default).
- Powered by VccH through J23 (disconnected by default to save power).
- Voltage: 0.20*Vbus, Use 1V reference for Vbus up to 5V.
- Current consumption: 7.7uA@5V, 2.8uA - 8.4uA (1.8V - 5.5V).

VDDCORE Buck Converter (L21/D51 only)



- The L21 and D51 both feature an on-chip buck converter (in addition to the default internal linear regulator). To support this converter, an inductor is installed in place of the 0ohm resistor.
- The L21 and D51 both lack pin A28 (pin used for Vsw instead).
- Jumper configuration set at factory. Do not change J30 or J34.
- The D51 uses a 4.7uF capacitor on Vddcore, rather than 1uF.

Analog Reference Filter

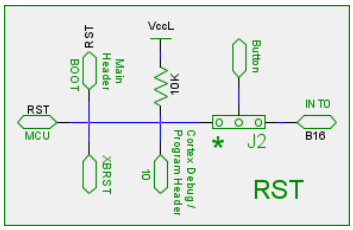


- J33 can be set to enable 100nF and 4.7uF capacitors on pin A3 (REFA) so that it can be used with an external reference (1V (2V for C21 ADC) to VddAna - 0.6V).

Mounting Holes

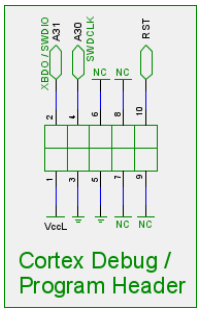
- Two mounting holes: 126mil (3.2mm) dia. hole, 212mil (5.4mm) dia. pad.
- The hole nearest the buck converter can be used to dissipate more heat.
- See 'Dimension Drawing' chapter for a drawing and link to drill file.

Reset



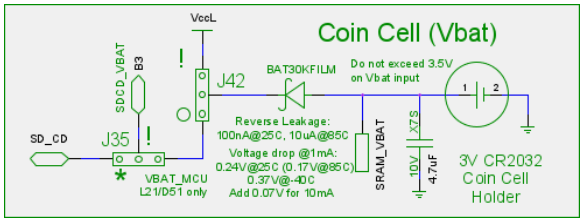
- Reset is always routed to the main header RST pin, to the Xbee header XBRST pin, and to the Cortex program/debug header.
- Reset is pulled to VccL through a 10Kohm resistor.
- By default, J2 connects reset to the button (and debounce filter). Press the button once to reset, **press twice in quick succession** to enter the USB CDC bootloader (Arduino compatible).

Cortex Header



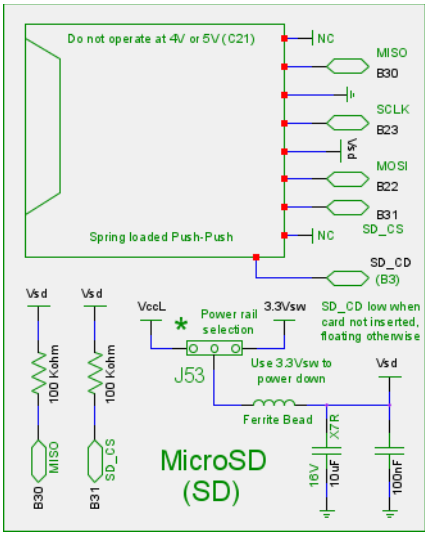
- 10-pin, 50-mil header (optional), for programming and debugging.
- **Do not install if it will interfere with XBee radio** (see 'Installing Headers').
- Be sure that nothing is connected to main header pins A30 or A31.
- A30 (SWDCLK) is pulled to VccL though a 1Kohm resistor.

Coin Cell (Vbat)



- Vbat (optional coin cell holder POS) can connect to SRAM Vbat, the Vbat pin (B3) of the L21 or D51, and/or VccL (through a schottky diode).
- POS pins must be shorted (holder does this).
- A 4.7uF capacitor is installed on Vbat.
- J42 connects Vbat either to VccL through a schottky diode, or to VBAT_MCU (L21 and D51 only). If connecting to VBAT_MCU (circle side), then J35 (SDCD_VBAT (B3)) must also be set to VBAT_MCU (opposite the asterisk).
- Diode voltage drop of 0.25V typ. (0.4V max.), reverse leakage of 0.1uA@25C (10uA @ 85C).
- If connecting to VccL (see J42), great care must be taken to limit current consumption to prevent excessive voltage drop as well as premature battery depletion. Set J42 to VccL only when using the main 3.3V regulator (VccH to VccL jumper not set) and only when the regulator input diode is used (J22 not set, which is the default).

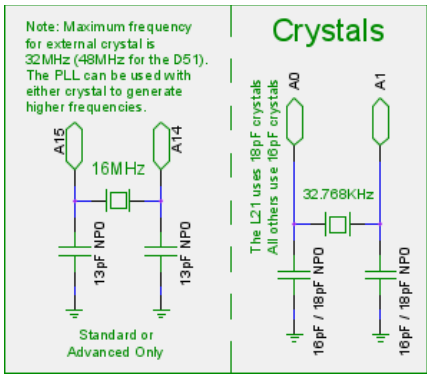
Micro SD Card Slot (SD)



- Connected to shared SPI bus at up to 12MHz.
- Vsd can connect to VccL (default) or 3.3Vsw via J53. Connection to 3.3Vsw allows powering down the card.
- Card detect (SD_CD/B3 through J35, set by default). SD_CD low when card NOT inserted, floating otherwise.
- Hot-insert cards (will not cause reset).
- Vsd power filter consists of 10uF and 100nF capacitors behind a ferrite bead (some DC resistance and HF filtering).
- 100Kohm pullup resistors on MISO (B30) and SD_CS (B31).
- **The D51 requires 3 jumper wires from the main header to the SPI header. See 'Special Notes for D51'.**

Crystals

- 32.768KHz crystal with two 16pF capacitors (18pF for L21). Use the high gain setting with the D51.
- 16MHz with two 13pF capacitors (Standard and Advanced)
- The PLL can be used with either to obtain higher frequencies



Clock Sources

- See the tables below for accuracy vs. power consumption vs. startup time. Note that the 32.768KHz crystal has by far the longest startup time, but excellent long term accuracy.

SAMD51 Clock Sources Table

Source	Frequency Range	Current (max.)	Startup Time typ. (max.)	Notes
GCLKIN	Up to 48MHz	NA	NA	Digital clock input
XOSC (x2)	8MHz-48MHz crystal	250uA (810uA) ENALC on	37K cycles (62K)	Up to 48MHz digital clock input, supply current based on 16MHz crystal
XOSC32K	32.768KHz crystal	1.9uA (3uA)	9K cycles (23K)	Use high gain setting
OSCULP32K	32.10-33.42KHz	Not Specified	NA	27.12-37.68KHz across temperature
FDPLL200M (x2)	32KHz-3.2MHz in, 96MHz-200MHz out	0.9mA (1.3mA) @ 96MHz	Lock: 54us (95us) @ 3.2MHz in	1.9% (2.7%) period jitter (32KHz in, 96MHz out), Datasheet lock time in ms.
DFLL48M open	47.2MHz-48.8MHz	400uA (850uA)	4.3us (7us)	45.8MHz-49.3MHz across temperature
DFLL48M closed	47.972MHz typical	400uA (850uA)	Lock: 429us (1145us)	0.42ns max. jitter, 0.73-33KHz input

** For reference only. See datasheet for electrical characteristics.*

SAMC21 Clock Sources Table

Source	Frequency Range	Current (max.)	Startup Time typ. (max.)	Notes
GCLKIN	Up to 48MHz	NA	NA	Digital clock input
XOSC	0.4MHz-32MHz crystal	429uA (699uA) AGC on	6K-12K cycles (20K-48K)	Up to 48MHz digital clock input, supply current based on 16MHz crystal
XOSC32K	32.768KHz crystal	1.53uA (2.84uA)	16K cycles (24K)	32.768KHz typical digital clock input
OSC32K	32.11-33.43KHz	0.864uA (1.08uA)	1 cycle (2 cycles)	25.55-37.36KHz across temperature
OSCULP32K	30.96-34.57KHz	Not Specified	NA	22.94-38.99KHz across temperature
OSC48M	47.52-48.48MHz	87uA (174uA)	22.5us (25.5us)	47.04-48.96MHz across temperature
FDPLL96M	32KHz-2MHz input, 48MHz-96MHz output	536uA (612uA) @48MHz out	Lock: 25us (35us) @ 2MHz in	1.8% (4%) period jitter (32KHz in, 48MHz out), Lock: 1.1ms (1.5ms) @ 32KHz in

** For reference only. See datasheet for electrical characteristics.*

SAML21 Clock Sources Table

Source	Frequency Range	Current (max.)	Startup Time typ. (max.)	Notes
GCLKIN	Up to 48MHz	NA	NA	Digital clock input
XOSC	0.4MHz-32MHz crystal	293uA (393uA) AGC on	5K-14K cycles (10K-48K)	Up to 24MHz digital clock input, supply current based on 16MHz crystal
XOSC32K	32.768KHz crystal	0.311uA (2.19uA)	25K cycles (82K)	32.768KHz (1MHz max.) digital input
OSC32K	32.57-33.05KHz	0.54uA (1.10uA)	1 cycle (2 cycles)	28.58-34.72KHz across temperature
OSCULP32K	31.77-34.03KHz	Not Specified	NA	26.29-38.39KHz across temperature
OSC16M	15.75-16.24MHz	141uA (169uA)	1.4us (3.1us)	Wake up time: 0.12us (0.25us)
FDPLL96M	32KHz-2MHz in, 48MHz-96MHz out	454uA (548uA) @48MHz out	Lock: 25us (35us) @ 2MHz in	1.9% (4%) period jitter (32KHz in, 48MHz out), Lock: 1ms (2ms) @ 32KHz in
DFLL48M open	46.6MHz-49MHz	286uA	8.3us (9.1us)	47.04-48.96MHz across temperature
DFLL48M closed	47.96-47.98MHz	362uA	Lock: 200us (700us)	0.51ns max. jitter, 0.73-33KHz input

** For reference only. See datasheet for electrical characteristics.*

SAMD21 Clock Sources Table

Source	Frequency Range	Current (max.)	Startup Time typ. (max.)	Notes
GCLKIN	Up to 48MHz	NA	NA	Digital clock input
XOSC	0.4MHz-32MHz crystal	307uA (552uA) AGC on	5K-14K cycles (10K-48K)	Up to 32MHz digital clock input, supply current based on 16MHz crystal
XOSC32K	32.768KHz crystal	1.22uA (2.19uA)	28K cycles (30K)	32.768KHz typical digital clock input
OSC32K	32.27-33.26KHz	0.67uA (1.32uA)	1 cycle (2 cycles)	28.50-34.74KHz across temperature
OSCULP32K	31.29-34.57KHz	0.125uA max.	10 cycles	25.55-38.01KHz across temperature
OSC8M	7.94-8.06MHz	64uA	2.1us (3us)	7.80-8.16MHz across temperature
FDPLL96M	32KHz-2MHz in, 48MHz-96MHz out	500uA (700uA) @48MHz out	Lock: 25us (50us) @ 2MHz in	1.5% (2%) period jitter (32KHz in, 48MHz out), Lock: 1.3ms (2ms) @ 32KHz in
DFLL48M open	47MHz-49MHz	403uA (453uA)	8us (9us)	47.04-48.96MHz across temperature
DFLL48M closed	47.96-47.98MHz	425uA (482uA)	Lock: 200us (500us)	0.42ns max. jitter, 0.73-33KHz input

** For reference only. See datasheet for electrical characteristics.*

Synchronization

- These MCUs have complex clock systems that require synchronization between clock domains (between the cpu and peripherals). The D21 uses version 1 of the synchronization system, while the rest use a simpler version 2. Please read 'Clock System' in the MCU datasheet.
- Sync delay: 5*PGCLK + 2*PAPB < D < 6*PGCLK + 3*PAPB

Headers and Pins

Power Header Pins

Pin	Header	Description
VccH	**Main (Y+ side)**	This is the main higher voltage rail used to power both 3.3V regulator inputs, the high-voltage side of the level shifter, and the Host/AUX switch. It is 5V by default. With the Basic and Standard, VccH is connected directly to USB Vbus (~5V). With the Advanced, VccH by default is supplied by the power multiplexer, which in turn selects between Vbuck (5V, 4V, or 3.6V) or USB Vbus (~5). The mux can be bypassed, with VccH connecting directly to Vbuck (5V, 4V, 3.6V, 2.9V, or 1.8V) or Vbus (~5). See 'Power Rails Table'.
VccL	**Main (Y+ side)**	This is the main lower voltage rail used to power the microcontroller, memory device, sensors, the low-voltage side of the level shifter, and the current measurement IC. It can optionally power the Vxbee, Vsd, and Vcon rails. By default, it supplies 3.3V from the main regulator. Alternatively, it can be directly connected to VccH, which in turn is supplied by either the buck converter (5V by default, 4V, 3.6V, 2.9V, 1.8V), USB Vbus (~5V), or the mux which selects between the two sources (3.6V minimum). Additionally, a regulated voltage can be applied to the VccL pin: 1.71V to 3.63V (2.7V to 5.5V for C21). See J19. See 'Power Rails Table'. The coin cell (Vbat) can be connected to VccL through a schottky diode so that when VccL is below 2.7V-2.8V (3V cell), then Vbat powers VccL. Use as a backup supply to VccL or for very low power operations. Do not use when VccL-VccH jumper set. **When connecting to VccH or using the VccL pin as input, do not operate the D21, L21, D51, SD card, sensors, or FLASH memory at >3.6V.**
3.3V	**Main (Y+ side)**	This is connected to the main 3.3V 250mA regulator output (3.3V). By default, the 3.3V rail it is routed to VccL through J19. See 'Main 3.3V Regulator'.
Gnd	**Main (Y+ side)**	Ground
Vaux	**Main (Y- side)**	This connects to the AUX output of the Host/AUX switch. It can also be tied to Vbus to double output current or use as input. See 'USB Host/AUX Switch'.
Gnd	**Main (Y- side)**	Ground
Vin	**Vin**	Vin can be 3V to 17V. Current from the Vin header pin flows through a PFET for reverse-polarity protection and through a current sense resistor to a 22uF bulk capacitor for use by the buck converter and/or motor controller. See 'Vin Section / Current Measurement IC'. The 22uF bulk ceramic capacitor (very low ESR) has a small resistance in series to increase the ESR (the buck converter also add a 1uF low-ESR ceramic), which helps control inductive voltage spiking that can result from the

Pin	Header	Description
		combination of low input capacitor ESR and inductive input wiring. Avoid excessive inductance on Vin (especially at 17V). Vin voltage, current, and power can be measured by the optional Current Measurement IC, with both analog output and I2C.
Gnd	**Vin**	Power Ground
Vmotor	**Motor Controller**	This connects to the Vmotor rail through a 2.5A standard fuse, for use with four independent open-drain outputs. The Vmotor rail can be 2.7V to 17V from either VccH (5V by default) or from Vin (up to 17V), depending on configuration. See 'Power Rails Table' and 'Motor Controller'.
Gnd	**Motor Controller**	Power Ground
VccH (2)	**Level Shifter (SERVO)**	These pins connect to VccH. When they are set at 5V (default), then two standard hobby servos (pwm-5V-Gnd) can be directly connected to pins 3-5-7 for one, and pins 4-6-8 for the other. See 'VccH' at the beginning of this table.
Gnd (2)	**Level Shifter (SERVO)**	Power Ground (2)
Vcon	**SPI**	Vcon is available on two external pins (both SPI and I2C headers). By default it is connected to VccL, but can instead connect to 3.3Vsw to supply more current and/or to turn off completely. See 'Power Rails Table'.
Gnd	**SPI**	Ground
Vcon	**I2C**	Vcon is available on two external pins (both SPI and I2C headers). By default it is connected to VccL, but can instead connect to 3.3Vsw to supply more current and/or to turn off completely. See 'Power Rails Table'.
Gnd	**I2C**	Ground
Batt_POS	**CR-2032 battery**	Positive side (dual pin) of the CR-2032 battery holder, which connects to Vbat. Vbat can connect to the SRAM Vbat pin (see J39), the Vbat pin of the L21 or D51 (pin B3, see J35 and J42), and/or to VccL through a schottky diode (see J42). A coin cell or other battery (up to 3.5V fully charged) can be connected. Note that the two pins are not connected together on the PCB, but both pins are connected to onboard hardware, so they must be shorted externally if connecting a battery without the holder. When using with VccL, care must be taken to limit current draw, and be sure that the main 3.3V regulator input diode is not bypassed (J22 not set, which is the default setting) so that the VccH is not back-powered through the regulator. See 'Coin Cell'.
Batt_NEG	**CR-2032 battery**	Negative side (single pin) of the CR-2032 battery holder, connects to ground.

** See also 'Power Rails Table' for addional information on all of the power rails.*

Main Header Pins

#	Port	Top Silk	IC	Peripheral Functions		Board Functions (bottom silk)		Notes
0	B0	B0	61	**O**	ADC	**B**	XBee_DTR_SLEEP **(XBDS)**	With all variants, this pin is free by default (analog or digital functions), but can connect to the Xbee radio DTR_SLEEP line. See J37. See 'XBee / XBee Pro Radio Socket' Chapter.
1	B1	B1	62	**O**	ADC	**+M**	MEM_CS **(MECS)**	When the optional memory device is installed, this pin is connected to the device chip select (which cannot be disconnected), and to a 100K pullup resistor to VccL, so the pin cannot be used. Otherwise, the pin can be used for analog or digital functions.
2	A2	A2	3	**O I**	ADC / DAC0 / EXTINT:2		**DAC**	This pin can be used for analog (including DAC output) or digital functions.
3	A3	A3	4	**O**	ADC / VREFA	**B**	REFA **(REFA)**	With all variants, this pin is free by default (analog or digital functions), but J33 can be set to enable both a 100nF capacitor and a 1uF capacitor for use with an external voltage reference. See J33.
4	A4	A4	13	**O**	ADC / VREFB / AC	**S**	3.3Vsw_VOUT **(3SVO)**	With all variants, this pin is free by default (analog or digital functions). With the Standard and Advanced variants, J40 can be set to connect the high current 3.3V regulator output (3.3Vsw) to this pin through a 1.8Kohm resistor for comparator (AIN0) threshold detection or ADC measurement (detect over-temp./over-current faults). See J40. See 'High Current 3.3V Regulator'.
5	A5	A5	14	**O**	ADC / DAC1 (L21/D51) / AC			This pin can be used for analog (including DAC out with L21/D51) or digital functions.
6	B6	B6	9	**O**	ADC	**S**	CURR_MON_VOUT **(CMVO)**	With all variants, this pin is free by default (analog or digital functions). With the Standard (with Vin option) and Advanced variants, J28 can be set to connect the Current Measurement IC (CUR) voltage output (Vout) to this pin through an RC filter for measurement. See J28. See 'Vin Section / Current Measurement IC'.
7	B7	B7	10	**O**	ADC	**+A**	AUX_SW_EN **(ASEN)**	With the advanced variant, the AUX switch enable input is connected to this pin

#	Port	Top Silk	IC	Peripheral Functions		Board Functions (bottom silk)		Notes
								through J27 by default. Otherwise, the pin can be used for analog or digital functions. See J27. See 'USB Host / Aux Power Switch (HOST)' Chapter.
8	B8	B8	11	**O I**	ADC / SERCOM4:0 / EXTINT:8		TX3	This pin can be used for analog or digital functions (no I2C).
9	B9	B9	12	**O I**	ADC / SERCOM4:1 / EXTINT:9		RX3	This pin can be used for analog or digital functions (no I2C).
10	A10	A10	19	**O I**	ADC / SERCOM2:2 / EXTINT:10	**A**	VccH_DIVIDER **(VHDV)** / MOSI1	With all variants, this pin is free by default (analog or digital functions). With the advanced variant, J23 and J24 can be set to connect the pin to the VccH voltage divider for measurement. See J23 and J24. See 'VccH Measurement' section in 'Other Hardware' Chapter. ADC on VDDIO.
11	A11	A11	20	**O I**	ADC / SERCOM2:3 / EXTINT:11	**+A**	Vbus_DIVIDER **(VBDV)** / SCK1	With the advanced variant, this pin is always connected to the Vbus voltage divider for measurement. Otherwise, the pin can be used for analog or digital functions. See 'Vbus Measurement' section in 'Other Hardware' Chapter. ADC on VDDIO.
12	A12	A12	29	**I ~**	SERCOM2:0 / TCC20 / EXTINT:12	**B**	XBee_CTS **(XBCT)** / SDA1 / MISO1	With all variants, this pin is free by default (digital functions), but can connect to the Xbee radio CTS line. See J29. See 'XBee / XBee Pro Radio Socket' Chapter.
13	A13	A13	30	**I ~**	SERCOM2:1 / TCC21 / EXTINT:13		SCL1 / SS1	This pin can be used for digital functions.
14	B14	B14	27	**I ~**	SERCOM4:2 / TC5(TC1):0 / EXTINT:13	**A**	HOST_SW_EN / USB_ID **(HSEN)**	With all variants, this pin is free by default (digital functions). With the advanced variant, J9 can be set to connect the HOST switch enable input to this pin (USB ID pin controls by default). See J9. See 'USB Host / Aux Power Switch (HOST)' Chapter.
15	B15	B15	28	**~**	SERCOM4:3 / TC5(TC1):1	**+A**	BUCK_FSW **(BKFS)**	With the advanced variant, this pin is always connected to the buck converter (BUCK) frequency selection pin (FSW). Otherwise, the pin can be used for digital functions. See 'Power Supply' Chapter.
16	A16	A16	35		SERCOM3:0	**+B**	I2C SDA **(SDA)**	With all variants, these pins connect

#	Port	Top Silk	IC	Peripheral Functions		Board Functions (bottom silk)		Notes
17	A17	A17	36		SERCOM3:1	**+B**	I2C SCL **(SCL)**	through J25 and J26 to 4.7Kohm pullup resistors to VccL for use with I2C. This is a shared bus that connects to the current measurement IC (if installed), the two optional sensor ICs, and the I2C header. Alternatively, these pins can be used for digital functions by clearing J25 and J26.
18	A18	A18	37	~	SERCOM1:2 / TC3(TC4):0	**+B**	TX1 / XBee_Din **(XBDI)**	With all variants, this pin connects through a 1.8Kohm resistor and J49 to the Xbee radio Din line. The pin can also be used as the chip select line when using SPI with the radio. See J49. There is always a 100Kohm resistor to Vxbee connected to this pin. See 'XBee / XBee Pro Radio Socket' Chapter.
19	A19	A19	38	~	TC3(TC4):1	**S**	CURR_MON_READINT **(CMRI)**	With all variants, this pin is free by default (digital functions). With the Standard (with Vin option) and Advanced variants, J14 can be set to connect the Current Measurement IC (CUR) Read/Integrate line (READINT) to this pin to select the operation mode (can be done via I2C). See J14. See 'Vin Section / Current Measurement IC' Chapter.
20	A20	A20	41	I ~	SERCOM3:2 / TC7(TC3):0 / EXTINT:4			This pin can be used for digital functions.
21	A21	A21	42	~	SERCOM3:3 / TC7(TC3):0			This pin can be used for digital functions.
22	A22	A22	43	~	TC4(TC0):0			This pin can be used for digital functions.
23	A23	A23	44	~	TC4(TC0):1			This pin can be used for digital functions. The D51 requires the use of this pin when using the shared SPI bus (Micro SD, MEM).
24	A24	A24	45		USB D- (D21/L21/D51) / CAN TX (C21)	**+B**	USB Micro D- **(D-)**	With all variants (except C21), these pins are disconnected. Instead, A24/A25 connect through J5/J6 to the Micro USB & ESD device. Alternatively (default for C21), use J5/J6 to connect A24/A25 to these header pins, which along with the adjacent Vbus and Gnd pins, can be used for panel-mount USB/CAN connector. See J5 and J6.
25	A25	A25	46		USB D+ (D21/L21/D51) / CAN RX (C21)	**+B**	USB Micro D+ **(D+)**	
26	---	---	---		NOT_A_PIN		NOT_A_PIN	Not a pin. For contiguous Arduino numbers.
27	A27	A27	51	I	EXTINT:15		INT2 (INT2)	With all variants, this pin is free by default (digital functions). It is also connected to the

#	Port	Top Silk	IC	Peripheral Functions		Board Functions (bottom silk)		Notes
								I2C header INT pin, and can also be connected to the Xbee radio ATTN line (SPI mode) through J50. See J50. See 'XBee / XBee Pro Radio Socket' Chapter.
28	A28	A28	53		A28 exists only on D21 and C21	**+B**	SPI_HEADER_CS **(SHCS)**	This pin is not connected with the L21 and D51 (the MCU pin is VDDCORE). With the D21 and C21, it is connected through J30 to the main header pin and through J31 to SPI header pin 5 (SPI_CS). It can be used for GPIO functions. See J30 and J31.
29	---	---	---		NOT_A_PIN		NOT_A_PIN	Not a pin. For contiguous Arduino numbers.
30	A30	A30	57		SWDCLK	**+B**	**SWDCLK**	With all variants, this pin connects to the Cortex header (SWDCLK) and is pulled to VccL though a 1Kohm resistor. It can be used for digital functions. It is important to leave this pin floating after reset, as it is used to detect debugger probes.
31	A31	A31	58		SWDIO / SERCOM1:3	**+B**	RX1 / XBee_Dout **(XBDO)**	With all variants, this pin connects to the Xbee radio Dout pin as well as the Cortex header (SWDIO). SWDIO is enabled only when using an external tool. The pin can be used for digital functions. See 'XBee / XBee Pro Radio Socket' Chapter.
32	B17	X32	40	**I ~**	EXTINT:1 / TC6(TC2):1	**+S**	MOTOR_PS **(MOPS)**	With the Standard and Advanced variants, this pin is always connected to the motor controller (MOTOR) Power Save (PS) line in addition to a 100Kohm pulldown resistor to ground (controller disabled by default). With the Basic, it can be used for digital functions. See 'Motor Controller' Chapter.
33	B16	X33	39	**I**	EXTINT:0	**+B**	INT0 **(INT0)**	With all variants, this pin always connects to the MCU. Additionally, it can connect to the button (instead of RST) through J2 and to the Xbee radio BTN line (so the radio can detect button presses for commissioning) through J52. See J2 and J52. See 'XBee / XBee Pro Radio Socket' and 'Button (Reset / INT0)' section in 'Other Hardware' chapter.
34	B2	X34	63	**O ~**	ADC / TC6(TC2):0	**+B**	LED **(LED)** / XBee_RTS **(XBRT)**	With all variants, this pin is connected to a blue LED through J1, J41, and a 1Kohm resistor. Alternatively, this pin can be used for digital functions by clearing J14. RTS and ASSOC LED can be used at

#	Port	Top Silk	IC	Peripheral Functions		Board Functions (bottom silk)		Notes
								same time. See 'Button (Reset / INT0)' section in 'Other Hardware' chapter.
35 to 42								See 'Motor Controller / Level Shifter Header Pins (SERVO)' in this chapter.
43 to 46								See 'SPI Header Pins' in this chapter.
47	B4	--- No Pin	5			**+B**	3.3Vsw_EN **(3SEN)**	With the Standard and Advanced, this pin is always connected to the high current 3.3V regulator (3.3Vsw) enable (3SEN) line and to a 100Kohm pull resistor to either VccL (on, default) or to ground. Unused with the Basic. See 'High Current 3.3V Regulator'.
48	B5	--- No Pin	6	**I**	EXTINT:5	**+B**	INT1 **(INT1)**	With all variants, this connects to the Accelerometer / Gyroscope INT, Motor Controller Fault, and USB Host / AUX Switch Flag. See 'INT1' in 'Other Hardware'.
49	B3	--- No Pin	64	**O I**	ADC / VBAT (L21/D51) / EXTINT:3	**+B**	VBAT / SD_CardDetect **(SDCD)**	With all variants, this by default connects through J35 to the Micro SD card slot card detect (CD) pin. Alternatively, it can connect to the coin cell (Vbat) through J35 and J42 to support the backup domain on the L21 and D51. See 'Coin Cell' and 'Micro SD'.
--	RST	RST	52		RESET	**+B**	RESET **(BOOT)**	The reset line connects to the MCU reset pin, the XBee header XBRST pin, the main header RST (BOOT) pin, the Cortex header RST pin, a 10K pullup to VccL, and by default to the button through J2. The button has an RC filter installed, which will place 100nF on reset, and 1.8K in series to the button. The button can be pressed once for reset, or twice in quick succession to enter the SAM-BA bootloader (if installed).

#: *The first column is the Arduino pin number, which usually corresponds to the port pin number (either A or B) but without the letter. Most pins can be used for more than one function.*
Peripheral Functions: *Shows MCU peripherals. O = analog, I = Interrupt, ~ = Timer (PWM).*
Board Functions: *Shows onboard hardware and board variant(s) with this hardware installed: B=Basic, S=Standard, A=Advanced, and M=Memory device installed. The Advanced variant has all of the hardware that the Standard and Basic have installed, and the Standard variant has all of the Basic hardware. + = By default, header pin in use by board function (may be able to free pin using jumper). No header pin for: A0, A1, A14, and A15 (crystals), and B3, B4, & B5 (used by onboard hardware).*

Motor Controller / Level Shifter Header Pins (SERVO)

Pin	#	Port	Silk	IC	Peripheral Functions		Board Functions (bottom silk)		Notes
1	**35**	A6	L35	15	**O I**	ADC / SERCOM0:2 / EXTINT:6	**+S**	LVL_SHIFT_0 **(L35)** / TX2	With the Standard and Advanced, these pins are connected to the Level Shifter (LVL) high side. With the Basic, the pins are connected directly to the MCU. See 'Level Shifter'.
2	**36**	A7	L36	16	**O I**	ADC / SERCOM0:3 / EXTINT:7	**+S**	LVL_SHIFT_1 **(L36)** / RX2	
3	**37**	A8	L37	17	**O ~**	ADC / SERCOM0:0 / TCC1:2 / EXTINT:NMI	**+S**	LVL_SHIFT_2 **(L37)**	With the Standard and Advanced, these pins are connected to the Level Shifter (LVL) high side. With the Basic, the pins are connected directly to the MCU. Can use with standard three pin hobby servos. See 'Level Shifter'.
4	**38**	A9	L38	18	**O ~**	ADC / SERCOM0:1 / TCC1:3	**+S**	LVL_SHIFT_3 **(L38)**	
5	**--**	**--**	**VccH**	**--**				Can use with standard three pin hobby servo connectors if 5V (default).	This is the main higher voltage rail used to power both 3.3V regulator inputs, the high-voltage side of the level shifter, and the Host/AUX switch. See 'Power Rails Table'.
6	**--**	**--**	**VccH**	**--**				See above	See above
7	**--**	**--**	**GND**	**--**					Power Ground
8	**--**	**--**	**GND**	**--**					Power Ground
9	**39**	B10	M39	23	**~**	TCC0:4 (TCC0:0)	**+S**	MOTOR_BOUT1 **(M39)**	With the Standard and Advanced, these pins are connected to the Motor Controller H-Bridge B (M39/M40) and H-Bridge A (M41/M42) outputs. Each pair of pins supports up to 1.13A with a voltage of 3V-17V. With the Basic, the pins are connected directly to the MCU. See 'Motor Controller'.
10	**40**	B11	M40	24	**~**	TCC0:5 (TCC0:1)	**+S**	MOTOR_BOUT2 **(M40)**	
11	**41**	B12	M41	25	**~**	TCC0:6 (TCC0:2)	**+S**	MOTOR_AOUT1 **(M41)**	
12	**42**	B13	M42	26	**~**	TCC0:7 (TCC0:3)	**+S**	MOTOR_AOUT2 **(M42)**	
13	**--**	**--**	**Vmotor**	**--**					This connects to the Vmotor rail through a **2.5A standard fuse**, for use with four independent open-drain outputs. The Vmotor rail can be 2.7V to 17V depending on configuration. See 'Power Rails Table'. See 'Motor Controller'.
14	**--**	**--**	**GND**	**--**					Power Ground

#: *The second column is the Arduino pin number, which usually corresponds to the port pin number (either A or B) but without the letter. Most pins can be used for more than one function.*
Peripheral Functions: *Shows MCU peripherals. O = analog, I = Interrupt, ~ = Timer (PWM).*
Board Functions: *Shows onboard hardware and board variant(s) with this hardware installed: B=Basic, S=Standard, A=Advanced, and M=Memory device installed. The Advanced variant has all of the hardware that the Standard and Basic have installed, and the Standard variant has all of the Basic hardware. + = By default, header pin in use by board function (may be able to free pin using jumper).*

Xbee Radio Header Pins (XBee Pro)

Pin	Pin Use	Most Digi Radios*	Digi Cellular	Skywire Cellular	GPS Bee	BLEBee v2.0.0	Bluetooth Bee
1	+	VCC	VCC	VCC	Vcc	3.3V	Vcc
2	D	DOUT	DOUT	DOUT	TX	TX	TX
3	D	DIN/CONFIG	DIN/CONFIG	DIN	RX	RX	RX
4	S, K	SPI_MISO	DIO12	GND	NC	NC	NC (P2_2)
5	+	RESET	RESET	RESET	NC (EX)	RESET	RESET
6	--	Unused	Unused	Unused	Unused	Unused	Unused
7	--	Unused	Unused	Unused	Unused	Unused	Unused
8	--	Unused	Unused	Unused	Unused	Unused	Unused
9	O	DTR/SLEEP	DTR/SLEEP	DTR	NC	NC	NC (P0_1)
10	+	GND	GND	GND	GND	Gnd	GND
11	S, K	SPI_MOSI	DIO4	GND	NC	NC	NC (P1_5)
12	O	CTS	CTS	CTS	NC	NC	CT
13	--	Unused	Unused	Unused	Unused	Unused	Unused
14	+	VREF (NC on some)	VREF	VREF	NC	NC	NC (P1_2)
15	O, K	ASSOCIATE	ASSOCIATE	GND	TP (1Hz-4Hz)	ASSOC	NC (P1_3)
16	O	RTS	RTS	RTS	NC	NC	RT
17	S	SPI_SSEL	DIO3	DIOx	NC	CH3	NC (P1_1)
18	S	SPI_CLK	DIO2	DIOx	NC	CH2	NC (P1_0)
19	O	SPI_ATTN	DIO1	ADC1	SCL	CH1	USB_N
20	O, K	CB	DIO0	PWR_ON	SDA	CH0	USB_P

*Pin Use Legend: + = Always connected, -- = Never connected, D = By default, this is connected, O = Can optionally be connected, S = Alternate SPI configuration, K = Must be GND for Skywire modems (see J52, J46, J45, J38). * = no SPI on DigiMesh*

SPI Header Pins (SPI)

Pin	#	Port	Top Silk	IC	Peripheral Functions		Board Functions (bottom silk)		Notes
1	**43**	B30	S43	59		SERCOM5:0	**+B**	SPI MISO **(S43)**	This pin is connected to the shared SPI bus MISO pin, which is always connected to the Micro SD, SPI header, and optional memory device. It can also connect to the Xbee radio. See 'Xbee Radio' for details. The D51 requires a jumper wire from main header pin A23 to SPI header pin 1 (also B30).
2	**--**	**--**	**Vcon**	**--**					Vcon is available on two external pins (both SPI and I2C headers). By default it is connected to VccL, but can instead connect to 3.3Vsw to supply more current and/or to turn off completely. See 'Power Rails Table'.
3	**44**	B23	S44	50		SERCOM5:3	**+B**	SPI SCK **(S44)**	This pin is connected to the shared SPI bus SCK pin. See 'MISO' pin above.
4	**45**	B22	S45	49		SERCOM5:2	**+B**	SPI MOSI **(S45)**	This pin is connected to the shared SPI bus MOSI pin. See 'MISO' pin above.
5	**46**	B31	S46	60		SERCOM5:1	**+B**	SS / SD_CS **(S46)**	This pin, which can be used for a chip select, is connected through J31 to either A28 (SHCS, no A28 on L21/D51) or B31 (shared with SD_CS, so SD card not used).
6	**--**	**--**	**GND**	**--**					Ground

#: *The second column is the Arduino pin number, which usually corresponds to the port pin number (either A or B) but without the letter. Most pins can be used for more than one function.*
Peripheral Functions: *Shows MCU peripherals. O = analog, I = Interrupt, ~ = Timer (PWM).*
Board Functions: *Shows onboard hardware and board variant(s) with this hardware installed: B=Basic, S=Standard, A=Advanced, and M=Memory device installed. The Advanced variant has all of the hardware that the Standard and Basic have installed, and the Standard variant has all of the Basic hardware. + = By default, header pin in use by board function (may be able to free pin using jumper).*
** The D51 requires a jumper wire from main header pin A23 to SPI header pin 1 (also B30). This is because SERCOM5 on the D51 must use IOSET4 (other MCUs can use B30, no IOSETs).*

I2C Header Pins (I2C)

Pin	#	Port	Top Silk	IC	Peripheral Functions		Board Functions (bottom silk)		Notes
1	**--**	**--**	**Vcon**	**--**					Vcon is available on two external pins (both SPI and I2C headers). By default it is connected to VccL, but can instead connect to 3.3Vsw to supply more current and/or to turn off completely. See 'Power Rails Table'.
2	**16**	A16	A16	35		SERCOM3:0	**+B**	I2C SDA **(SDA)**	With all variants, these pins connect through J25 and J26 to 4.7Kohm pullup resistors to VccL for use with I2C. This is a shared bus that connects to the current measurement IC (if installed), the two optional sensor ICs, and the I2C header. Instead, these can be used for digital functions by clearing J25 and J26.
3	**17**	A17	A17	36		SERCOM3:1	**+B**	I2C SCL **(SCL)**	
4	**27**	A27	A27	51	I	EXTINT:15		INT2 (INT2)	With all variants, this pin is free by default (digital functions). It is connected to the I2C header INT pin and the main header A27 pin, and can also be connected to the Xbee radio ATTN line (SPI mode) through J50. See J50. See 'XBee / XBee Pro Radio Socket' Chapter.
5	**--**	**--**	**GND**	**--**					Ground

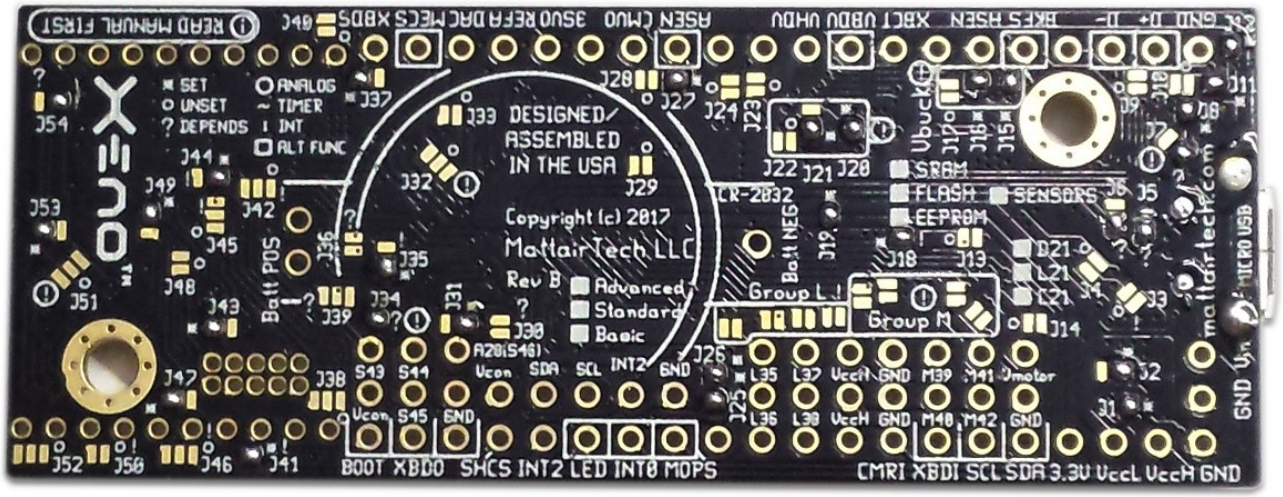
#: *The second column is the Arduino pin number, which usually corresponds to the port pin number (either A or B) but without the letter. Most pins can be used for more than one function.*
Peripheral Functions: *Shows MCU peripherals. O = analog, I = Interrupt, ~ = Timer (PWM).*
Board Functions: *Shows onboard hardware and board variant(s) with this hardware installed: B=Basic, S=Standard, A=Advanced, and M=Memory device installed. The Advanced variant has all of the hardware that the Standard and Basic have installed, and the Standard variant has all of the Basic hardware. + = By default, header pin in use by board function (may be able to free pin using jumper).*

Cortex Header Pins (PROG)

Pin	Function	Description
1	**VccL**	This is the main lower voltage rail used to power the microcontroller, memory device, sensors, the low-voltage side of the level shifter, and the current measurement IC. It can optionally power the Vxbee, Vsd, and Vcon rails. See 'Power Rails Table'.
2	**SWDIO / XBDO (A31)**	With all variants, this pin connects to the Xbee radio Dout pin as well as the Cortex header (SWDIO). SWDIO is enabled only when using an external tool. The pin can be used for digital functions. See 'XBee / XBee Pro Radio Socket' Chapter.
3	**Gnd**	Ground
4	**SWDCLK (A30)**	With all variants, this pin connects to the Cortex header (SWDCLK) and is pulled to VccL though a 1Kohm resistor. It can be used for digital functions. It is important to leave this pin floating after reset, as it is used to detect debugger probes.
5	**Gnd**	Ground
6	**NC**	No pcb connection
7	**NC**	No pcb connection
8	**NC**	No pcb connection
9	**NC**	No pcb connection
10	**RST**	The reset line connects to the MCU reset pin, the XBee header XBRST pin, the main header RST (BOOT) pin, the Cortex header RST pin, a 10K pullup to VccL, and by default to the button through J2. The button has an RC filter installed, which will place 100nF on reset, and 1.8K in series to the button. The button can be pressed once for reset, or twice in quick succession to enter the SAM-BA bootloader (if installed).

Solder Jumpers

Using Solder Jumpers



This image is of the Standard variant with the D51 installed.
*Silkscreen Legend: * = Set by default, o = Unset by default, ? = Depends on hardware options.*
The symbol also serves as a polarity indicator for 3-pad jumpers, referenced in the table below.

- There are 54 solder jumpers labeled with the J prefix.
  - The default setting for each jumper is indicated as follows:
    * = Set by default, o = Unset by default, ? = Depends on hardware options
  - Some jumpers have 3 pads. The 'default setting' symbol is used as a polarity indicator.
  - Some jumpers have 3 pads with no center pad connection, indicating greater care needed. In these cases, all three pads are soldered together.
  - Some jumpers are at a 45 degree angle, usually indicating greater care needed.
- There are 4 jumpers in Group M for the motor controller (factory only, do not set or unset).
- There are 4 jumpers in Group L for the level shifter (factory only, do not set or unset).

Solder Jumper Table

Jumper	*Default*	*Description*
J1: LED Enable 2-pad: (LED - Blue_LED)	* Default: Connected	This jumper connects the LED net (X34 pin on main header) to a blue LED through a 1Kohm resistor. The LED net can in turn be driven by either the LED_XBRT pin (B2) through J41 (set by default), or the Xbee radio ASSOC pin through J46. Drive the pin high to turn on the LED. Disconnect J1 if using the X34 header pin for purposes incompatible with the LED. See 'Blue LED' section of 'Other Hardware' chapter for details and a simplified schematic.
J2: Button Selection 3-pad: * = RST (INT0 (B16) - BUTTON - RST)	* Default: RST	This connects the button to the RST pin by default. The button can connect to the INT0 pin (B16, X33 on main header) instead by setting J2 (opposite the asterisk) and enabling the internal pullup. This button is debounced using a 1.8Kohm resistor and a 100nF capacitor. The pin is grounded when the button is pressed. The button can be completely disconnected by removing solder from all three pads. Xbee radio BTN pin can also be connected to INT0 by setting J52 (circle side). See 'Button (Reset / INT0)' section of 'Other Hardware' chapter for details and a simplified schematic.
J3: PFET Bypass 2-pad: (PFET_OUT - Vin)	**O** Default: NC **! Use Caution**	By default, this jumper is not set. Set this jumper to bypass the PFET. Note that the RDSon is very low, so the voltage drop is also very low. **Bypassing the PFET will disable reverse-polarity protection.**
J4: Sense Resistor Bypass 2-pad: (PFET_OUT - HVin)	**O** Default: NC	By default, this jumper is not set. Set this jumper to bypass the 10mOhm current sense resistor for the current measurement IC (PAC1921). Bypassing the sense resistor will disable power and current measurement (voltage measurement will still work).
J5: A25 to USB D+ 3-pad: ? = Micro_D+ (Micro_D+ - A25 - Header_D+)	**?** Default depends: D21 or L21: Micro_D+ C21: Header_D+	By default (D21/L21/D51), pins A24 and A25 are connected to the D- and D+ pins of the Mini USB connector (and ESD device) through jumpers J6 and J5. Alternatively, set J5 and J6 to the position opposite the question mark to connect A24 and A25 to the main header (default for the C21). These pins, along with the adjacent Vbus and Gnd pins can be used for a panel-mount USB connector (see J7). Connect all three pads to keep the ESD device in the circuit when using the header pins.
J6: A24 to USB D- 3-pad: ? = Micro_D- (Micro_D- - A24 - Header_D-)	**?** Default depends: D21 or L21: Micro_D- C21: Header_D-	

Jumper	Default	Description
J7: Vaux - VccH - Vbus 3-pad: **?** = Vbus (Vaux - VccH – Vbus) **Do not operate the D21, L21, D51, SD card, optional sensors, or optional FLASH memory at >3.6V. Thus, DO NOT CONNECT VccH to Vbus (or Vaux >3.6V) if the VccH-VccL jumper is set (J32), and one of the above is installed. See J32 for more information.**	**?** Default depends: Basic: Vbus Standard: Vbus Advanced: All NC **!! Use Extra Caution**	**Basic and Standard**: By default, J7 connects VccH directly to Vbus. Alternatively, set J7 to the position opposite the question mark to connect VccH to Vaux to use the pin as an input (panel-mount connector). **Advanced**: The power multiplexer is instead used to connect VccH to either Vbus or Vbuck (J7 is all NC). Alternatively, J7 can be set to the position with the question mark to connect VccH to Vbus, thus allowing bypassing of the MUX and comparator to save power when the buck converter is not used. See 'power Supply Configurations' and 'USB Host / Aux Power Switch'.
J8: Comparator Power 2-pad: (VccH - COMP_PWR)	* Default: Connected	Connected by default to supply both power and a reference voltage to the comparator REF (Vcc) pin (Advanced variant only) using a VccH/2 resistor divider. When J8 is connected, the comparator will switch the MUX to Vbuck when present (2.9V, 3.6V, 4V, and 5V). Disconnect J8 to force the MUX to stay in auto mode (switch to the higher of Vbus or Vbuck, to force Vbus to be selected when present). Only do this if Vbuck is 4V or lower. It is OK for the comparator IN and OUT pins to have a voltage present when un-powered, and the OUT pin will remain high impedance in this case. Also disconnect j8 if the MUX is bypassed, to save a few uA. See 'Power Supply' chapter for more information.
J9: Host Switch Enable 2-pad: (FPF2300_ONA - HSEN (B14))	**O** Default: NC	With the Advanced, the USB Host switch output is off by default, and is controlled automatically via the USB Micro connector ID pin. Set J9 to control manually (or read the pin state) using HSEN (B14). Low is on. See 'USB Host / Aux Power Switch'.
J10: Vbus - Vaux 2-pad: (Vaux - Vbus)	**O** Default: NC	J10 can be set in many cases to tie Vbus to Vaux, so that the output current is doubled to 2A or to allow Vbus access to a header pin (input or output depending on configuration). Do not set if VccH is connected directly to Vbuck. See 'power Supply Configurations' and 'USB Host / Aux Power Switch'.
J11: Vbus - Host Switch 2-pad: (Vbus - FPF2300_OUTA)	* Default: Connected **! Use Caution**	**J11 and J12 must always have the same setting**. They are on by default, which connects Vbus to both the power multiplexer and USB Host switch output. Do not use the Host switch without the multiplexer. See 'power Supply Configurations' and 'USB Host / Aux Power Switch'.
J12: Vbus - Mux 2-pad: (FPF2300_OUTA - VbusMux)	* Default: Connected **! Use Caution**	

Jumper	Default	Description
J13: HVin - Vmotor 2-pad: (Vmotor - HVin)	**O** Default: NC	By default, the motor controller is connected to VccH (5V default). With the Vin option installed, it can instead be powered by Vin (3V to 17V) by setting J13 and clearing J18. Do not connect both jumpers. See 'Motor Controller'.
J14: READINT 2-pad: (READINT - CMRI (A19))	**O** Default: NC	If the Vin option is installed, the _READ_/INT pin of the current measurement IC can be controlled manually using CMRI (A19) by soldering J14. Note that there is no pull resistor, so this pin should be connected if not overriding the pin using I2C control. Bring CMRI low to enter the Read state and high to enter the Integrate state. See 'Vin Section / Current Measurement IC'.
J15: Buck 412Kohm 2-pad: (GND - Buck_412Kohm)	* Default: Connected **!! Use Extra Caution**	Use these three jumpers to set the output of the buck converter (5V by default, 4V, 3.6V, 2.9V, or 1.8V). These jumpers are among several jumpers that must be changed as a group. **See 'Power Supply' chapter and 'Buck Converter Output Voltage Table'. Do not operate the D21, L21, D51, SD card, optional sensors, or optional FLASH memory at >3.6V.**
J16: Buck 130Kohm 2-pad: (Buck_130Kohm - GND)	* Default: Connected **!! Use Extra Caution**	
J17: Buck 1Mohm 2-pad: (Buck_1Mohm - Vbuck)	**O** Default: NC **!! Use Extra Caution**	
J18: VccH - Vmotor 2-pad: (VccH_Diode - Vmotor)	* Default: Connected **! Use Caution**	By default, J18 is set, connecting the motor controller to VccH (5V default) through a schottky diode. With the Vin option installed, it can instead be powered by Vin (3V to 17V) by setting J13 and clearing J18. Note that while there is a diode, reverse leakage (especially at high temperatures) can be excessive with both jumpers set, so, **do not connect both jumpers**. See 'Motor Controller'.
J19: 3.3V - VccL 2-pad: (3.3V - VccL)	* Default: Connected	By default, J19 is set, connecting the main 3.3V regulator output to VccL. Clear J19 when connecting VccL directly to VccH, or when using an externally regulated voltage on the VccL pin. J19 can also be used to measure VccL current by clearing the jumper and measuring across the 3.3V and VccL main header pins. See 'Power Supply' chapter.
J20: Vbuck - MUX_IN2 2-pad: (MUX_IN2 - Vbuck)	* Default: Connected **!! Use Extra Caution**	This connects the output of the buck converter (5V by default, 4V, or 3.6V) to one power multiplexer input (the other is USB Vbus @5V). J20 is one of several jumpers that must be changed as a group. **See 'Power Supply' chapter for configurations.**

Jumper	Default	Description
J21: VccH - Vmux / Vbuck 3-pad: * = Vmux (Vmux - VccH - Vbuck)	* Default: Vmux **!! Use Extra Caution**	This connects VccH either to the power multiplexer output, or to the output of the buck converter (5V by default, 4V, 3.6V, 2.9V, or 1.8V). J21 is one of several jumpers that must be changed as a group. **See 'Power Supply' chapter for configurations.**
J22: 3.3V Reg Diode Bypass 2-pad: (VccH - TPS727_IN)	**O** Default: NC **! Use Caution**	Set this jumper to short the input diode of the main 3.3V regulator to eliminate the voltage drop when using the buck converter at 3.6V. See 'Power Supply' chapter and 'Coin Cell' in 'Other Hardware'. **Do not use coin cell (Vbat) with VccL when set.**
J23: VccH Divider Enable 2-pad: (VccH_DIV_TOP - VccH)	**O** Default: NC	Set J23 to connect the VccH voltage divider top resistor to VccH. Set J24 as well to connect the divider to VHDV (A10) for ADC measurement. Keep both J23 and J24 disconnected to save a few uA. See 'VccH Divider' in 'Other Hardware'.
J24: VccH Divider Measure 2-pad: (VccH_DIV - VHDV (A10))	**O** Default: NC	
J25: I2C SDA Pullup 2-pad: (SDA (A16) - VccL)	* Default: Connected	J25 and J26 connect pin A16 and A17 through 4.7Kohm resistors to VccL for use with I2C SDA and SCL. This shared I2C bus connects to the current measurement IC (if installed), the two optional sensor ICs, and the I2C header. Instead, these can be used for digital functions by clearing J25 and J26.
J26: I2C SCL Pullup 2-pad: (SCL (A17) - VccL)	* Default: Connected	
J27: Aux Switch Enable 2-pad: (FPF2300_ONB - ASEN (B7))	* Default: Connected	With the Advanced, the AUX switch output is on by default (1Mohm pulldown), and is connected to ASEN (B7) through J27. Clear J27 to use for other puposes. See 'USB Host / Aux Power Switch'.
J28: Current Monitor Vout 2-pad: (CMVO (B6) - PAC1921_OUT)	**O** Default: NC	If the Vin option is installed, set J28 to connect the OUT pin of the current measurement IC to CMVO (B6) for low-latency ADC measurement. Otherwise, measurements can still be read from I2C registers. See 'Vin Section / Current Measurement IC'.
J29: XBee CTS 2-pad: (XBCTS - XBCT (A12))	**O** Default: NC	Set J29 to connect the Xbee radio CTS pin to XBCT (A12). The RTS pin can also be used (see J47) at the expense of the MCU controlling the blue LED (although the Xbee radio can then control it). See 'Xbee Radio Socket' chapter.
J30: A28 Header Pin 2-pad: (A28 - A28_VDDCORE)	**?** Default depends: D21/C21: Connected L21/D51: NC	**D21 or C21 installed:** This jumper connects the A28 MCU pin to header pin A28. With the L21 and D51, this pin is VDDCORE (because of internal buck converter). **Do not set J30 with the L21 or D51.**
J31: SPI_CS Select 3-pad: * = A28	* Default: A28	With the D21 or C21, this connects A28 to the SPI header pin 5 (SPI_CS). J30 must also be set. J31

Jumper	Default	Description
(SD_CS (B31) - SPI_CS - A28)		can instead connect SPI_CS to SD_CS (B31) by soldering away from the asterisk (SD card connot be used). **Do not set J30 with the L21 or D51.**
J32: VccH - VccL 3-pad: **O** = NC (VccH - NC – VccL) **Do not operate the D21, L21, D51, SD card, optional sensors, or optional FLASH memory at >3.6V.**	**O** Default: NC **!!! Use Extra Caution**	This jumper can be used to connect VccL directly to VccH, so that VccL can operate at a voltage other than 3.3V (5V, 4V, 3.6V, 2.9V, or 1.8V), and/or to increase efficiency and current capability (ie: cellular modem). J32 is one of several jumpers that must be changed as a group. **See 'Power Supply' chapter for configurations.**
J33: REFA capacitors 2-pad: (A3_CAPACITORS - GND)	**O** Default: NC	When using pin A3 as REFA, set J33 to enable both a 100nF capacitor and a 1uF capacitor from A3 to ground.
J34: VSW (L21) 2-pad: (VDDCORE_VSW - A28_VDDCORE)	**?** Default depends: D21/C21: NC L21/D51: Connected **!! Use Extra Caution**	**L21 or D51 installed:** This jumper connects the VDDCORE pin (pin A28 does not exist with the L21/D51) to a 10uH inductor to VSW. It will also connect to both a 1uF (4.7uF for D51) capacitor and a 100nF capacitor to ground. This supports the internal buck converter of the L21 and D51. **Do not set this jumper with the D21 or C21.**
J35: B3 - SD_CD / VBAT_MCU 3-pad: * = SD_CD (SD_CD - SDCD_VBAT (B3) - VBAT_MCU)	* Default: SD_CD **! Use Caution**	J35 connects SDCD_VBAT (B3) to either SD_CD (Micro SD card detect) or to VBAT_MCU, to support the Vbat battery backup pin (B3) found on the **L21 or D51 only**. If connecting to SD_CD (default), then pin B3 will be driven low when a card is inserted. **If connecting to VBAT_MCU (opposite the asterisk), then see J42 for more information. See 'Coin Cell (Vbat)' section of 'Other Hardware' chapter for details and a simplified schematic.**
J36: MEM _WP to Gnd 2-pad: (GND - MEM _WP)	**?** Default depends: SRAM: NC FLASH: NC EEPROM: Connected NO MEMORY: NC	**SRAM installed:** Pin is NC, so leave unconnected. **FLASH installed:** J36 is not set, so the FLASH _WP_ will be pulled high by an internal resistor. Set J36 to connect the _WP_ pin to GND. Firmware must then also enable write protection, otherwise writes can still be made with _WP_ low. **EEPROM installed:** J36 is set to connect the EEPROM _WP_ pin to GND. Firmware must also enable write protection, thus writes can still be made with _WP_ low. If J36 is not soldered, _WP_ will float, so leave it connected. **No memory device installed:** Leave unconnected

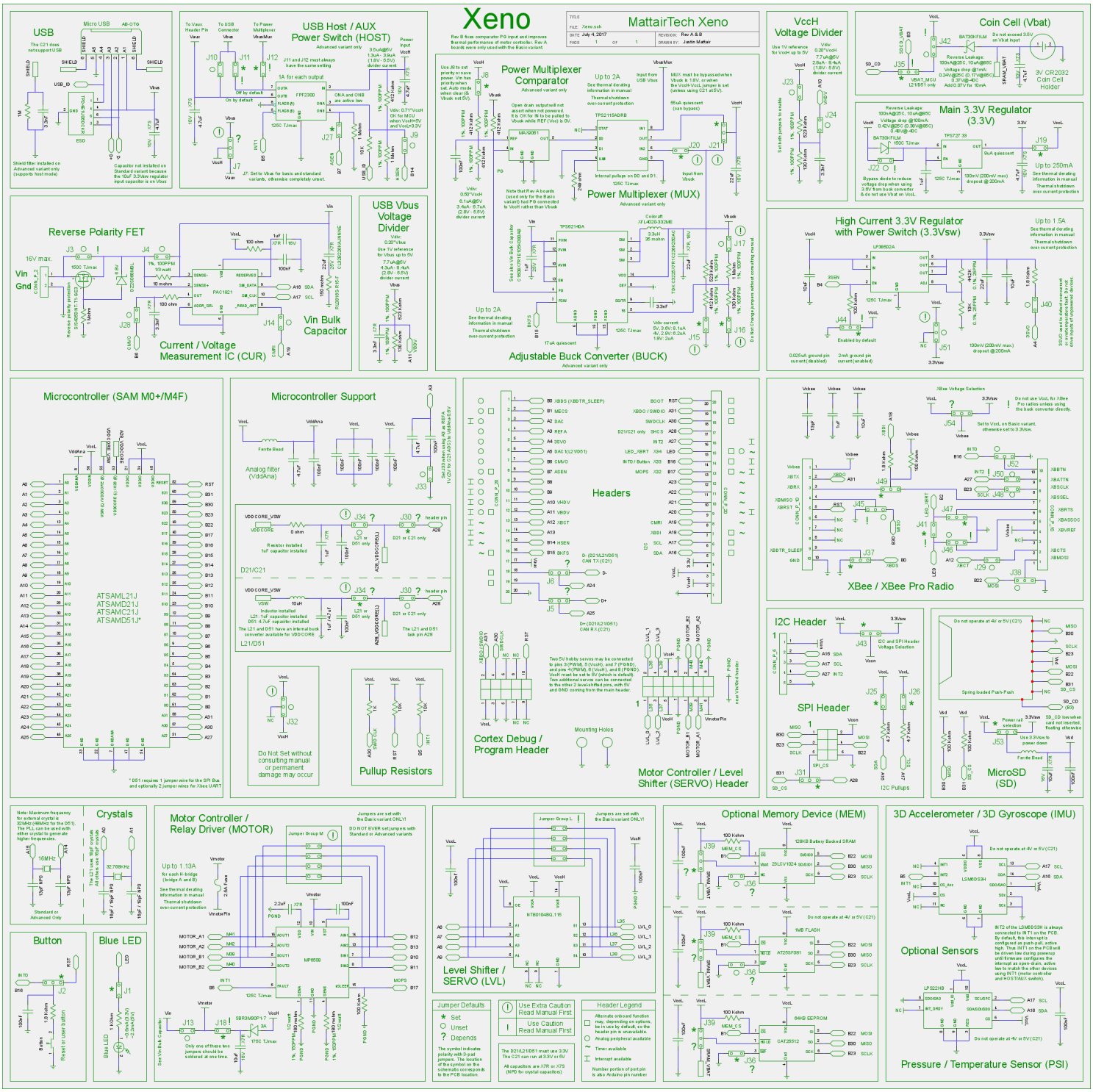
Jumper	Default	Description
J37: XBee DTR / SLEEP 3-pad: * = GND (GND - XBDTR_SLEEP - XBDS (B0))	* Default: GND	By default, J37 is set to GND, disabling pin sleep. Most Xbee radios support software sleep, but in some case, lower power consumption is possible with pin sleep. To enable pin sleep or DTR, set J37 opposite the asterisk and control the pin with XBDS (B0). Set J37 to GND with the Skywire cell modems.
J38: XBee MOSI 3-pad: **O** = MOSI (GND - XBMOSI - MOSI (B22))	**O** Default: NC	When using SPI for Xbee communication, solder J38 (circle side) to connect XBMOSI to the shared SPI bus MOSI line (B22). **Set J38 to GND only with the Skywire cell modems.**
J39: MEM_Vbat_hold to Vbat / Vcc 3-pad: **?** = SRAM_VBAT (SRAM_VBAT - MEM_Vbat_hold - VccL)	**?** Default depends: SRAM: SRAM_VBAT FLASH: VccL EEPROM: VccL NO MEMORY: NC	**SRAM installed:** J39 (solder near the ?) connects the SRAM Vbat pin directly to Vbat (coin cell holder POS) to support battery backup operation. **FLASH or EEPROM installed:** J39 (solder opposite the ?) connects the FLASH or EEPROM _HOLD_ pin to VccL, which disables the hold feature. **No memory device installed:** Leave unconnected
J40: 3.3Vsw Measure 2-pad: (3SVO (A4) - 3.3Vsw)	**O** Default: NC	Setting this jumper allows either measuring or digital reading of the 3.3Vsw rail voltage through 1.8Kohm resistor. This is useful to determine if the regulator has entered thermal or over-current shutdown. When this happens, the MCU **should not drive or pull lines high that are routed to devices powered by 3.3Vsw.** An interrupt can be attached. **See 'High Current 3.3V Regulator' section of 'Power Supply' chapter for details.**
J41: MCU LED 2-pad: (LED_XBRT (B2) - LED)	* Default: Connected **! Use Caution**	By default, the LED is connected to the MCU (LED_XBRT) through J41 and J1.The XBee radio ASSOC pin can instead drive the LED by setting J46 (circle side) and **clearing J41**. LED_XBRT can then optionally be connected to the radio RTS line (GND by default) by setting J47 (opposite the asterisk). See 'Blue LED' section of 'Other Hardware' chapter for details and a simplified schematic.
J42: Vbat Output Routing 3-pad: **O** = VBAT_MCU (VccL - Vbat - VBAT_MCU)	**O** Default: NC **! Use Caution**	J42 connects Vbat (coin cell holder POS) either to VccL through a schottky diode, or to **VBAT_MCU (L21 and D51 only)**. If connecting to VBAT_MCU (circle side), then J35 (SDCD_VBAT (B3)) must also be set to VBAT_MCU (opposite the asterisk). **If connecting to VccL, great care must be taken to limit current consumption** to prevent excessive voltage drop as well as premature battery depletion. **Set J42 to VccL only when using the main 3.3V**

Jumper	Default	Description
		regulator (VccH to VccL jumper not set) and **only when the regulator input diode is used** (J22 not set, which is the default). **Additionally, do not drive lines high that are routed to unpowered devices. See 'Coin Cell (Vbat)' section of 'Other Hardware' chapter for details and a simplified schematic.**
J43: Header Power Select 3-pad: * = VccL (VccL - Vcon - 3.3Vsw)	* Default: VccL	The Standard and Advanced variants come with the high-current 3.3V regulator with enable, which is connected to the 3.3Vsw rail. Devices connected to 3.3Vsw can consume more current than VccL, as well as be powered off completely. By default, Vcon (which is connected to both the SPI and I2C headers) is connected to VccL. Switch J43 (opposite the asterisk) to connect to 3.3Vsw. Note that when switching off Vcon, be sure not to drive or pull any lines high that are routed to devices powered by it.
J44: 3.3Vsw Default State 3-pad: * = VccL (GND - 3SEN - VccL)	* Default: VccL **! Use Caution**	This sets the direction of the pull resistor on the enable line of the high-current regulator (3.3Vsw). By default, J44 is set to VccL, so the regulator is enabled. Switch J44 to GND to keep the regulator disabled during powerup (until the MCU enables it). **Do not to drive or pull lines high** that are routed to devices powered by 3.3Vsw when it is disabled.
J45: XBee SPI MISO 3-pad: **O** = MISO (MISO (B30) - XBMISO - GND)	**O** Default: NC **! Use Caution**	When using SPI for Xbee communication, solder J45 (circle side) to connect XBMISO to the shared SPI bus MISO line (B30). **Set J45 to ground only with the Skywire cell modems.**
J46: XBee Associate 3-pad: **O** = LED (GND - XBASSOC - LED)	**O** Default: NC **! Use Caution**	By default, the LED is connected to the MCU (LED_XBRT) through J41 and J1.The XBee radio ASSOC pin can instead drive the LED by setting J46 (circle side) and **clearing J41**. LED_XBRT can then optionally be connected to the Xbee radio RTS line (which by default is grounded) by setting J47 (opposite the asterisk). See 'Blue LED' section of 'Other Hardware' chapter for details and a simplified schematic. **Set J46 to ground only with the Skywire cell modems.**
J47: XBee RTS 3-pad: * = GND (GND - XBRTS - LED_XBRT (B2))	* Default: GND	By default, the MCU LED_XBRT line is connected to the LED through J41 and J1, and the Xbee radio RTS line is grounded. The LED_XBRT line can instead be connected to the RTS line by clearing J41 and setting J47 (opposite the asterisk). See 'Blue LED' section of 'Other Hardware' chapter for details and a simplified schematic.

Jumper	Default	Description
J48: XBee SCLK 2-pad: (XBSCLK - SCLK (B23))	**O** Default: NC	When using SPI for Xbee communication, solder J48 to connect XBSCLK to the shared SPI bus SCLK line (B23).
J49: XBee RX / SSEL 3-pad: * = XBRX (XBSSEL - XBDI (A18) - XBRX)	* Default: XBRX	By default, XBDI (A18) is connected to XBRX (Xbee header) for use with UART mode. When using SPI, switch J49 (opposite the asterisk) to connect XBDI to XBSSEL for use as the SPI chip select.
J50: XBee ATTN 2-pad: (XBATTN - INT2 (A27))	**O** Default: NC **! Use Caution**	When using SPI for Xbee communication, J50 can be soldered to connect XBATTN to INT2 (A27) so that the radio can notify the MCU it has data. **INT2 is also routed to the main and I2C headers. Because XBATTN is push-pull, do not use other devices on INT2.**
J51: VccL - 3.3Vsw 3-pad: **O** = NC (VccL - NC - 3.3Vsw)	**O** Default: NC **!!! Use Extra Caution**	When VccL is tied to VccH directly, set all 3 pads of J51 to add 32uF from high current 3.3V regulator output (ie: **cell modem**) to VccL. J51 is one of several jumpers that must be changed as a group. **See 'Power Supply' chapter for configurations.**
J52: XBee Button 3-pad: **O** = INT0 (GND - XBBTN - INT0 (B16))	**O** Default: NC	The button is connected to the RST pin by default. The button can connect to the INT0 pin (B16, X33 on main header) instead by setting J2 (opposite the asterisk) and enabling the internal pullup. The XBee radio BTN pin can then also be connected to INT0 by setting J52 (circle side). See 'Button (Reset / INT0)' section of 'Other Hardware' chapter for details and a simplified schematic. **Set J52 to ground only with the Skywire cell modems.**
J53: SD Power Select 3-pad: * = VccL (VccL - Vsd - 3.3Vsw)	* Default: VccL	The Standard and Advanced variants come with the high-current 3.3V regulator with enable, which is connected to the 3.3Vsw rail. Devices connected to 3.3Vsw can consume more current than VccL, as well as be powered off completely. By default, Vsd (which is connected to the SD card socket) is connected to VccL. Switch J43 (opposite the asterisk) to connect to 3.3Vsw. Note that **when switching off Vsd, be sure not to drive or pull any lines high** that are routed to the Micro SD card.
J54: XBee Power Select 3-pad: **?** = VccL (VccL - Vxbee - 3.3Vsw)	**?** Default depends: Basic: VccL Standard: 3.3Vsw Advanced: 3.3Vsw **! Use Caution**	The Standard and Advanced variants come with the high-current 3.3V regulator with enable, which is connected to the 3.3Vsw rail. Devices connected to 3.3Vsw can consume more current than VccL, as well as be powered off completely. Note that **when switching off Vxbee, be sure not to drive or pull any lines high** that are routed to the radio. There is

Jumper	Default	Description
		a current limiting resistor installed in series with the TX line (XBDI) only. **Do not use VccL for XBee Pro radios unless using the buck converter directly.**
Jumper Group M 2-pad (x4): (B10 - MOTOR_B1 / M39), (B11 - MOTOR_B2 / M40), (B12 - MOTOR_A1 / M41), (B13 - MOTOR_A2 / M42)	**?** Default depends: Basic: All Connected Standard: All NC Advanced: All NC **!!! Use Extra Caution**	With the motor controller option installed (Standard and Advanced), all jumpers in Jumper Group M are clear. **Do not ever set these jumpers**, as doing so will short the inputs with the outputs. The jumpers are set for the Basic variant only, which provides header access to the pins.
Jumper Group L 2-pad (x4): (A6 - LVL_0 / L35), (A7 - LVL_1 / L36), (A8 - LVL_2 / L37), (A9 - LVL_3 / L38)	**?** Default depends: Basic: All Connected Standard: All NC Advanced: All NC **! Use Caution**	With the level shifter option installed (Standard and Advanced), all jumpers in Jumper Group L are clear. **Use caution when setting these** (ie: to bypass or when VccL = VccH), as doing so will short the inputs with the outputs, so be sure that the header pin voltage is compatible with the MCU. The jumpers are set by default for the Basic variant only, which provides header pin access.

Schematic


Xeno
MattairTech Xeno
USB
Micro USB
USB Host / AUX Power Switch (HOST)
Power Multiplexer Comparator
Power Multiplexer (MUX)
Adjustable Buck Converter (BUCK)
VccH Voltage Divider
Coin Cell (Vbat)
Main 3.3V Regulator (3.3V)
High Current 3.3V Regulator with Power Switch (3.3Vsw)
Reverse Polarity FET
Current / Voltage Measurement IC (CUR)
USB Vbus Voltage Divider
Vin Bulk Capacitor
Microcontroller (SAM M0+/M4F)
ATSAML21J
ATSAMD21J
ATSAMC21J
ATSAMD51J*
Microcontroller Support
Analog filter (VddAna)
Headers
Cortex Debug / Program Header
Motor Controller / Level Shifter (SERVO) Header
XBee / XBee Pro Radio
I2C Header
SPI Header
I2C Pullups
MicroSD (SD)
Crystals
Button
Blue LED
Motor Controller / Relay Driver (MOTOR)
Level Shifter / SERVO (LVL)
Optional Memory Device (MEM)
3D Accelerometer / 3D Gyroscope (IMU)
Optional Sensors
Pressure / Temperature Sensor (PSI)
Pullup Resistors
Jumper Defaults
Header Legend

Parts List

Note that part designators are not yet used, so use this table along with the schematic.

Part	Part #	Description	B	S	A	Notes
Microcontroller (SAM M0+)						
SAMD21 – 256KB	ATSAMD21J18A-AU	IC MCU 32BIT 256KB FLASH 64TQFP	1	1	1	option
SAML21 – 256KB	ATSAML21J18B-AU	IC MCU 32BIT 256KB FLASH 64TQFP	1	1	1	option
SAMC21 – 256KB	ATSAMC21J18A-AU	IC MCU 32BIT 256KB FLASH 64TQFP	1	1	1	option
SAMD51 – 1MB	ATSAMD51J20A-AU	120MHZ 1024KB FLASH 64 TQFP PKG	1	1	1	option
Crystals						
High Speed Crystal	7A-16.000MAHE-T	CRYSTAL 16.000 MHZ 12PF SMD	0	1	1	
Crystal capacitors	GRM1555C1H130GA01D	CAP CER 13PF 2% 50V C0G/NP0 0402	0	2	2	
32.768KHz crystal	9HT7-32.768KDZF-T	CRYSTAL 32.768KHZ 12.5PF SMD	1	1	1	
Crystal capacitors	GRM1555C1H180FA01D	CAP CER 18PF 1% 50V C0G/NP0 0402	2	2	2	L21 only
Crystal capacitors	GRM1555C1H160GA01D	CAP CER 16PF 2% 50V C0G/NP0 0402	2	2	2	Not L21
Microcontroller Support						
Ferrite bead	LI0603G221R-10	FERRITE 700MA 220 OHM 0603 SMD	1	1	1	
Analog filter cap	C1608X7S1A475K080AC	CAP CER 4.7UF 10V 10% X7S 0603	1	1	1	
Capacitor	GRM155R71C104KA88D	CAP CER .1UF 16V X7R 0402	7	7	7	
Capacitor	EMK107B7105KA-T	CAP CER 1.0UF 16V X7R 0603	1	1	1	
VDDCORE cap	EMK107B7105KA-T	CAP CER 1.0UF 16V X7R 0603	1	1	1	Not D51
VDDCORE cap	C1608X7S1A475K080AC	CAP CER 4.7UF 10V 10% X7S 0603	1	1	1	D51 only
Inductor	VLS201612CX-100M	FIXED IND 10UH 770MA 540 MOHM	1	1	1	L21 or D51
0 ohm resistor	ERJ-6GEY0R00V	RES SMD 0.0 OHM JUMPER 1/8W 0805	1	1	1	D21 or C21
Reset pullup	ERJ-2RKF1002X	RES SMD 10K OHM 1% 1/10W 0402	1	1	1	
USB						
USB Connector	0475890001	USB - micro AB Receptacle Connector 5 Position Surface Mount, Right Angle, Horizontal	1	1	1	
USB ESD	TPD4E1U06DCKR	TVS DIODE 5.5VWM 15VC SC70-6	1	1	1	
Shield resistor	ERJ-2RKF1004X	RES SMD 1M OHM 1% 1/10W 0402	0	0	1	
Shield capacitor	CL05B332KA5NNNC	CAP CER 3300PF 25V X7R 0402	0	0	1	
Vbus capacitor	C1608X7S1A475K080AC	CAP CER 4.7UF 10V 10% X7S 0603	1	0	1	
USB Vbus Voltage Divider						
Vbus divider top resistor	ERJ-2RKF5233X	RES SMD 523K OHM 1% 1/10W 0402	0	0	1	

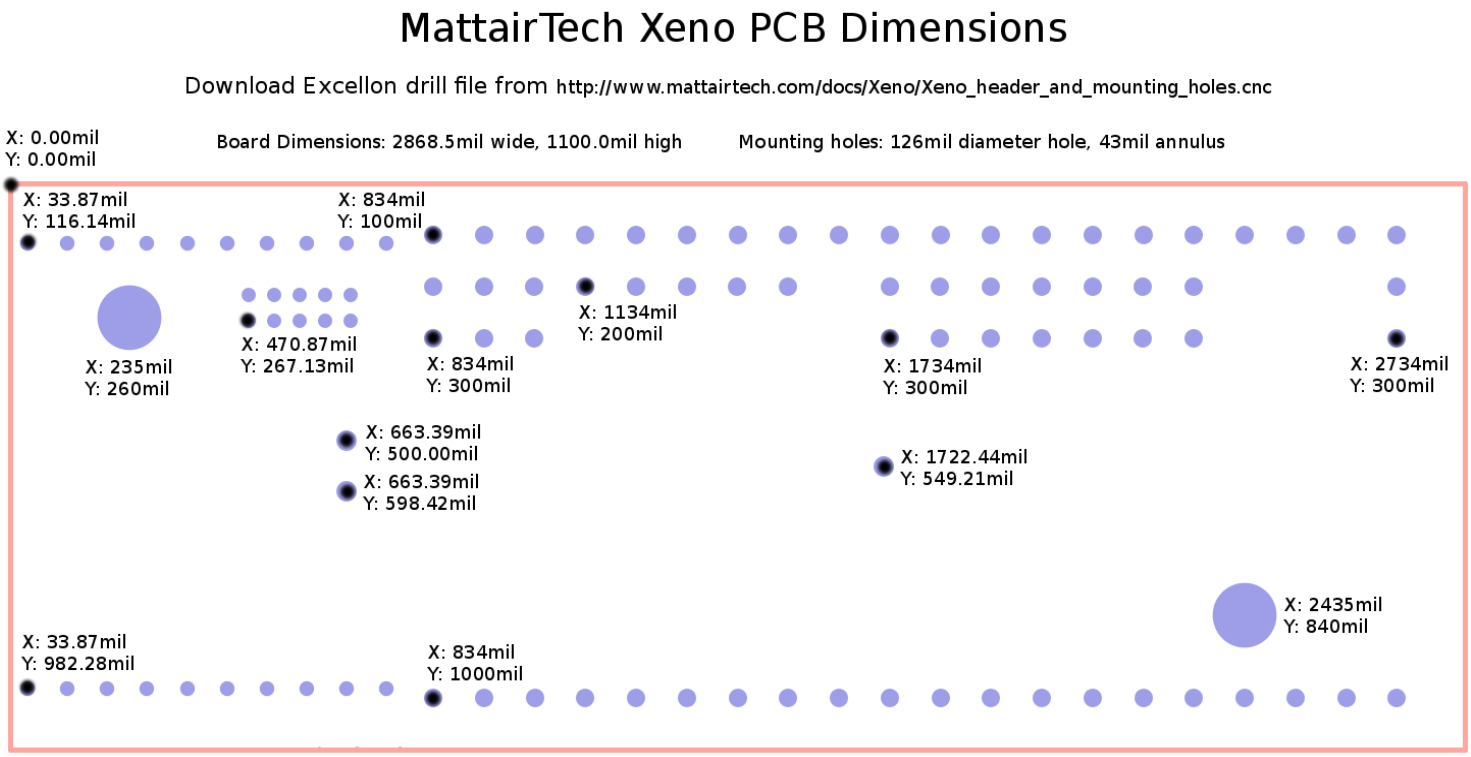
Part	Part #	Description	B	S	A	Notes
Vbus divider bottom resistor	ERJ-2RKF1303X	RES SMD 130K OHM 1% 1/10W 0402	0	0	1	
Vbus divider cap	CL05B332KA5NNNC	CAP CER 3300PF 25V X7R 0402	0	0	1	
USB Host / AUX Power Switch (HOST)						
USB host switch	FPF2300MPX	IC LOAD SW 2CH RESTART 8-MLP	0	0	1	
host switch caps	C1608X7S1A475K080AC	CAP CER 4.7UF 10V 10% X7S 0603	0	0	2	
USB_ID current limit resistor	ERJ-2RKF1002X	RES SMD 10K OHM 1% 1/10W 0402	0	0	1	
ONA divider top resistor	ERJ-2RKF4123X	RES SMD 412K OHM 1% 1/10W 0402	0	0	1	
ONA divider bottom resistor	ERJ-2RKF1004X	RES SMD 1M OHM 1% 1/10W 0402	0	0	1	
ONB Pulldown	ERJ-2RKF1004X	RES SMD 1M OHM 1% 1/10W 0402	0	0	1	
Power Multiplexer (MUX)						
Input Power Mux	TPS2115ADRBR	IC OR CTRLR SRC SELECT 8SON	0	0	1	
Mux current limit resistor	ERJ-2RKF2490X	RES SMD 249 OHM 1% 1/10W 0402	0	0	1	
Mux output capacitor	LMK316AB7226ML	22µF ±20% 10V X7R Ceramic Capacitor -55°C ~ 125°C Surface Mount, MLCC 1206	0	0	1	
Power Multiplexer Comparator						
Comparator	MAX9061EUK+T	IC COMPARATOR SGL LP SOT23-5	0	0	1	
PG pullup	ERJ-2RKF4123X	RES SMD 412K OHM 1% 1/10W 0402	0	0	1	
Comparator divider top resistor	ERJ-2RKF4123X	RES SMD 412K OHM 1% 1/10W 0402	0	0	1	
Comparator divider bottom resistor	ERJ-2RKF4123X	RES SMD 412K OHM 1% 1/10W 0402	0	0	1	
Capacitor	GRM155R71C104KA88D	CAP CER .1UF 16V X7R 0402	0	0	1	
VccH Voltage Divider						
VccH divider top	ERJ-2RKF5233X	RES SMD 523K OHM 1% 1/10W 0402	0	0	1	
VccH divider bottom	ERJ-2RKF1303X	RES SMD 130K OHM 1% 1/10W 0402	0	0	1	
VccH divider cap	CL05B332KA5NNNC	CAP CER 3300PF 25V X7R 0402	0	0	1	
Button						
button	PTS810 SJM 250 SMTR LFS	SWITCH TACTILE SPST-NO 0.05A 16V	1	1	1	
button resistor	ERJ-2RKF1801X	RES SMD 1.8K OHM 1% 1/10W 0402	1	1	1	
Capacitor	GRM155R71C104KA88D	CAP CER .1UF 16V X7R 0402	1	1	1	
Main 3.3V Regulator (MREG, 3.3V rail)						

Part	Part #	Description	B	S	A	Notes
3.3V regulator	TPS72733DSER	IC REG LDO 3.3V 0.25A 6WSON	1	1	1	
Input diode	BAT30KFILM	DIODE SCHOTTKY 30V 300MA SOD523	1	1	1	
Regulator Input cap	EMK107B7105KA-T	CAP CER 1.0UF 16V X7R 0603	1	1	1	
Regulator output capacitor	C1608X7S1A475K080AC	CAP CER 4.7UF 10V 10% X7S 0603	1	1	1	
Reverse Polarity FET						
Reverse polarity FET	SIS435DNT-T1-GE3	MOSFET P-CH 20V 30A 1212-8	0	0	1	
FET gate resistor	ERJ-2RKF1004X	RES SMD 1M OHM 1% 1/10W 0402	0	0	1	
FET gate zener clamp	DZ2S068M0L	DIODE ZENER 6.8V 150MW SSMINI2	0	0	1	
Current / Voltage Measurement IC (CUR)						
Current sense IC	PAC1921-1-AIA-TR	IC CURRENT SENSE MON SGL 10VDFN	0	0	1	
Current Sense Resistor	CSM0603FT10L0	RES SMD 0.01 OHM 1% 1/3W 0603	0	0	1	
power filter resistor	ERJ-2RKF1000X	RES SMD 100 OHM 1% 1/10W 0402	0	0	1	
power filter cap	EMK107B7105KA-T	CAP CER 1.0UF 16V X7R 0603	0	0	1	
Capacitor	GRM155R71C104KA88D	CAP CER .1UF 16V X7R 0402	0	0	1	
OUT pin filter resistor	ERJ-2RKF1000X	RES SMD 100 OHM 1% 1/10W 0402	0	0	1	
OUT pin filter cap	CL05B332KA5NNNC	CAP CER 3300PF 25V X7R 0402	0	0	1	
Vin Bulk Capacitor						
bulk capacitor	CL32B226KAJNNNE	22µF ±10% 25V Ceramic Capacitor X7R 1210	0	0	1	
Bulk cap resistor	RL0816S-R15-F	RES SMD 0.15 OHM 1% 1/5W 0603	0	0	1	
Adjustable Buck Converter (BUCK)						
ADJ buck converter	TPS62140ARGTR	IC REG BUCK ADJ 2A SYNC 16QFN	0	0	1	
Input capacitor	CGA3E1X7R1E105K080AC	CAP CER 1UF 25V X7R 0603	0	0	1	
power inductor	XFL4020-332ME	FIXED IND 3.3UH 5.2A 35MOHM SMD	0	0	1	
power capacitor	C3225X7R1C226K250AC	CAP CER 22UF 16V X7R 1210	0	0	1	
Soft-start capacitor	CL05B332KA5NNNC	CAP CER 3300PF 25V X7R 0402	0	0	1	
div top1 resistor	ERJ-2RKF5233X	RES SMD 523K OHM 1% 1/10W 0402	0	0	1	
div top2 resistor	ERJ-2RKF1004X	RES SMD 1M OHM 1% 1/10W 0402	0	0	1	
div bottom1 resistor	ERJ-2RKF4123X	RES SMD 412K OHM 1% 1/10W 0402	0	0	1	
div bottom2 resistor	ERJ-2RKF1303X	RES SMD 130K OHM 1% 1/10W 0402	0	0	1	

Part	Part #	Description	B	S	A	Notes
Motor Controller / Relay Driver (MOTOR)						
Motor controller	MP6508GR-P	Bipolar Motor Driver Power MOSFET PWM 16-QFN (4x4)	0	1	1	
Vdd capacitor	LMK107B7225KA-T	CAP CER 2.2UF 10V X7R 0603	0	1	1	
boost capacitor	GRM155R71C104KA88D	CAP CER .1UF 16V X7R 0402	0	1	1	
bulk cap (VccH)	CL21B106KOQNNNE	CAP CER 10UF 16V X7R 0805	0	1	1	
current sense resistors	ERJ-6DSFR18V	RES SMD 0.18 OHM 1% 1/2W 0805	0	2	2	
5V rail diode	SBR3M30P1-7	DIODE SBR 30V 3A POWERDI123	0	1	1	
pulldown resistor	ERJ-2RKF1003X	RES SMD 100K OHM 1% 1/10W 0402	0	1	1	
fuse	ERB-RD2R50X	FUSE BOARD MOUNT 2.5A 32VDC 0402	0	1	1	
Blue LED						
Status LED	LB Q39G-L2OO-35-1	LED CHIPLED BLUE 470NM 0603 SMD	1	1	1	
LED Resistor	ERJ-2RKF1001X	RES SMD 1K OHM 1% 1/10W 0402	1	1	1	
Pullup Resistors						
INT1 pullup	ERJ-2RKF1002X	RES SMD 10K OHM 1% 1/10W 0402	1	1	1	
TWI pullups	ERJ-2RKF4701X	Thick Film Resistors - SMD0402 4.7Kohms 1% Tol	2	2	2	
pull resistors	ERJ-2RKF1003X	RES SMD 100K OHM 1% 1/10W 0402	3	3	3	
SWCLK pullup	ERJ-2RKF1001X	RES SMD 1K OHM 1% 1/10W 0402	1	1	1	
High Current 3.3V Regulator with Power Switch (HREG, 3.3Vsw rail)						
High current 3.3V regulator	LP38502ASD-ADJ/NOPB	IC REG LDO ADJ 1.5A 8WSON	0	1	1	
input capacitor	CL21B106KOQNNNE	CAP CER 10UF 16V X7R 0805	0	1	1	
output capacitor	LMK316AB7226ML	22µF ±20% 10V X7R Ceramic Capacitor -55°C ~ 125°C Surface Mount, MLCC 1206	0	1	1	
output capacitor	CL21B106KOQNNNE	CAP CER 10UF 16V X7R 0805	0	1	1	
enable pull resistor	ERJ-2RKF1003X	RES SMD 100K OHM 1% 1/10W 0402	0	1	1	
Vdiv top resistor	ERA-2AEB4422X	RES SMD 44.2KOHM 0.1% 1/16W 0402	0	1	1	
Vdiv bottom resistor	ERA-2AEB103X	RES SMD 10K OHM 0.1% 1/16W 0402	0	1	1	
Rail measure resistor	ERJ-2RKF1801X	RES SMD 1.8K OHM 1% 1/10W 0402	0	1	1	
Coin Cell (Vbat)						
Vbat diode	BAT30KFILM	DIODE SCHOTTKY 30V 300MA SOD523	1	1	1	
Vbat bulk capacitor	C1608X7S1A475K080AC	CAP CER 4.7UF 10V 10% X7S 0603	1	1	1	
Battery holder	BU2032-1-HD-G	HOLDER COIN CELL 2032 PC PIN	0	0	0	

Part	Part #	Description	B	S	A	Notes
XBee / XBee Pro Radio						
Xbee bulk capacitor	CL21B106KOQNNNE	CAP CER 10UF 16V X7R 0805	1	1	1	
Xbee mf capacitor	EMK107B7105KA-T	CAP CER 1.0UF 16V X7R 0603	1	1	1	
Xbee hf capacitor	CL05C130JB5NCNC	CAP CER 13PF 50V 5% NP0 0402	1	1	1	
TX overcurrent resistor	ERJ-2RKF1801X	RES SMD 1.8K OHM 1% 1/10W 0402	1	1	1	
Micro SD Card Slot (SD)						
MicroSD card slot	1010058159	CONN MINI MICRO-SD 8PIN PCB GOLD	1	1	1	
Ferrite bead	LI0603G221R-10	FERRITE 700MA 220 OHM 0603 SMD	1	1	1	
bulk capacitor	CL21B106KOQNNNE	CAP CER 10UF 16V X7R 0805	1	1	1	
hf capacitor	GRM155R71C104KA88D	CAP CER .1UF 16V X7R 0402	1	1	1	
3D Accelerometer / 3D Gyroscope (IMU), Pressure / Temperature Sensor (PSI)						
Accelerometer/gyroscope	LSM6DS3HTR	INEMO INERTIAL MODULE: 3D ACCELEROMETER / 3D GYRO	0	0	0	optional
pressure sensor	LPS22HBTR	MEMS NANO PRESSURE SENSOR	0	0	0	optional
Capacitor	GRM155R71C104KA88D	CAP CER .1UF 16V X7R 0402	0	0	0	optional
Memory Device (MEM)						
SRAM	23LCV1024-I/ST	IC NVSRAM 1MBIT 20MHZ 8TSSOP	0	0	0	optional
EEPROM	CAT25512YI-GT3	IC EEPROM 512KBIT 20MHZ 8TSSOP	0	0	0	optional
FLASH	AT25SF081-XMHD-T	IC FLASH 8MBIT 104MHZ 8TSSOP	0	0	0	optional
CS pull resistor	ERJ-2RKF1003X	RES SMD 100K OHM 1% 1/10W 0402	0	0	0	optional
Capacitor	GRM155R71C104KA88D	CAP CER .1UF 16V X7R 0402	0	0	0	optional
Level Shifter (LVL)						
level shifter	NTB0104BQ,115	TXRX TRANSLATING 3STATE 14DHVQFN	0	1	1	
shifter capacitors	GRM155R71C104KA88D	CAP CER .1UF 16V X7R 0402	0	2	2	
Headers						
1-row headers		CONN HEADER 36-40PS .100 STR GOLD	0	0	0	
6-pin headers		CONN HEADER VERT DUAL 6POS GOLD	0	0	0	
Xbee headers	NPPN101BFCN-RC	10 Position Header Connector 0.079" (2.00mm) Through Hole Gold	0	0	0	
Cortex Header	GRPB052VWVN-RC	CONN HEADER .050" 10PS DL PCB AU	1	1	1	

Dimension Drawing



The two main headers, as well as the Motor controller / Level shifter (Servo), I2C, SPI, and Vin headers are all on a common 100mil grid. The XBee header pins have 2mm pin spacing, and the Cortex programming header has 50mil pin spacing.

MattairTech Arduino SAM M0+ Core

Please visit https://github.com/mattairtech/ArduinoCore-samd for the latest version.

The MattairTech SAM M0+ Core is a fork from arduino/ArduinoCore-samd on GitHub, which will be used to maintain Arduino support for MattairTech branded boards (see https://www.mattairtech.com/) as well as for "Generic" boards.

- Supports the SAMD21, SAMD11, SAML21, and SAMC21.
- Supports four clock sources (two crystals, internal oscillator, and USB calibrated).
- USB CDC Bootloader with optional SDCard support

This core is intended to be installed using Boards Manager (see below). To update from a previous version, click on MattairTech SAM M0+ Boards in Boards Manager, then click Update.

New Version Numbering The MattairTech version number will now track with the Arduino version number, to better understand which upstream changes have been merged in. See the CHANGELOG for details on upstream commits and MattairTech additions that have been merged.

What's New - Release Version (1.6.16)

- Added MattairTech Xeno support (64-pin D21, L21, and C21)
- Changed version numbering to match Arduino SAMD core to indicate which upstream changes have been merged in.
  - Release version 1.6.7 then skips to 1.6.16.
    Beta version 1.6.8-beta-b2 skips to 1.6.16-beta-b0.
- Merged in changes from upstream SAMD CORE 1.6.16 2017.08.23:
  - PWMs now can perform real 16-bit resolution if analogWriteResolution(16) is set. Thanks @Adminius
  - USB CDC: fixed issue of available() getting stuck when receiving ZLP's
  - Serial (UART) tx is now buffered.
  - Updated Stream and Print class
  - Native USB now supports USB Serial Number
  - Fixed pgm_read_ptr compatibility macro. Thanks @nkrkv
- Documentation updates

What's New Beta (1.6.16-beta)

See Beta Builds section for installation instructions.

1.6.16-beta-b0:

Beta version 1.6.16-beta-b0 became release version 1.6.16. See above

1.6.8-beta-b2:

- Added SD Card firmware loading support to the bootloader (4KB and 8KB)
- Removed SDU library, as the bootloader now supports SD cards directly
- Removed automatic page writes from bootloader (may have caused bricked board during development)
- Fixed bootloader compilation on Windows
- Added more Serial, SPI, and WIRE instances to MT-D21E (rev A and B)
- Added support for up to 6 SERCOM on the L21E (32-pin)
- Merged in changes from upstream SAMD CORE 1.6.16 (not released yet):
  - USB CDC: fixed issue of available() getting stuck when receiving ZLP's
- Merged in changes from upstream SAMD CORE 1.6.15 2017.04.27 (not relevant)
- Merged in changes from upstream SAMD CORE 1.6.14 2017.04.04:
  - Added lowpower function on USB subsystem
- Documentation updates

1.6.8-beta-b1:

- Fixed auto-reset not working on some versions of Windows
- Documentation updates

1.6.8-beta-b0:

- Added L21 and C21 support. Improved D11D and D11C support.
  - Use Tools->Microcontroller menu to select mcu.
- Both the core and bootloader have added support for:
  - external high-speed crystal (400KHz - 32MHz) using PLL
  - external 32.768KHz crystal using PLL
  - internal oscillator with USB calibration using DFLL
  - internal oscillator using DFLL in open-loop mode (or 48MHz RC oscillator with C21)
  - PLL_FRACTIONAL_ENABLED and PLL_FAST_STARTUP options
  - The clock source is selectable in the Tools->Clock Source menu
- New Tools->Serial Config menu for selecting different combinations of serial peripherals
- New Tools->Bootloader Size menu allows selection of bootloader size
- New Tools->USB Config menu simplifies USB configuration compared to previous core
- Updated variant.cpp table format for future CCL and GCLK use. See VARIANT_COMPLIANCE_CHANGELOG.
- Updated bootloader.
- Updated bossac upload tool (fixed support for SAML and SAMC)
- New CMSIS-Atmel package (this is different than from Arduino)
- Merged in all changes from upstream through SAMD CORE 1.6.14 (April 2017)

Features Summary

Feature	21J (64 pin)	21G (48 pin)	21E (32 pin)	D11 (24, 20, or 14 pin)
Board Variants	MattairTech Xeno, Generic 21J	Arduino Zero, Arduino M0, Generic 21G	MT-D21E, Generic 21E	MT-D11, Generic D11D14AM, Generic D11D14AS, Generic D11C14A
Processor	48 MHz 32-bit ARM Cortex M0+	48 MHz 32-bit ARM Cortex M0+	48 MHz 32-bit ARM Cortex M0+	48 MHz 32-bit ARM Cortex M0+
Flash Memory	Up to 256KB, L21 & C21 have RWW	Up to 256KB, L21 & C21 have RWW	Up to 256KB, L21 & C21 have RWW	16 KB (4KB used by bootloader)
SRAM	Up to 32KB (plus <=8KB LPSRAM on L21)	Up to 32KB (plus <=8KB LPSRAM on L21)	Up to 32KB (plus <=8KB LPSRAM on L21)	4 KB
Digital Pins	52 (51 for L21)	38 (37 for L21)	26 (25 for L21)	24-pin: 21, 20-pin: 17, 14-pin: 11
Analog Inputs	18 channels, 12-bit	14 channels, 12-bit	10 channels, 12-bit	24-pin: 10, 20-pin: 8, 14-pin: 5 (12-bit)
Analog Outputs	One 10-bit (two 12-bit on L21)	One 10-bit (two 12-bit on L21)	One 10-bit (two 12-bit on L21)	One 10-bit
PWM Outputs	18	14	14	8 (6 for 14-pin)
Interrupts	16	16	16	8 (7 for 14-pin)
USB	Full Speed Device and Host (not C21)	Full Speed Device and Host (not C21)	Full Speed Device and Host (not C21)	Full Speed Device
SERCOM	6	6	4	3 (2 for 14-pin)
UART (Serial)	Up to 3 (more later)	Up to 3	Up to 4 (6 for L21)	Up to 2
SPI	Up to 2 (more later)	Up to 2	Up to 2	Up to 1
I2C (WIRE)	Up to 2 (more later)	Up to 2	Up to 2	Up to 1
I2S	D21 only	D21 only	D21 only	Not present
Voltage	1.62V-3.63V (2.7V-5.5V for the C21)	1.62V-3.63V (2.7V-5.5V for the C21)	1.62V-3.63V (2.7V-5.5V for the C21)	1.62V-3.63V
I/O Pin Current	D21: 7mA, L21: 5mA, C21: 6mA@5V	D21: 7mA, L21: 5mA, C21: 6mA@5V	D21: 7mA, L21: 5mA, C21: 6mA@5V	7 mA

** The maximum number of UART/SPI/I2C is the number of SERCOM. The number listed above for UART/SPI/I2C indicated how many are currently configurable through the Arduino IDE menu.*

Board Variants

Pin configuration and peripheral assignment information is now in the README.md for each board variant. README.md also now includes technical information on the new PinDescription table format.

- MattairTech Xeno (SAMx21Jxxx)
- MattairTech MT-D21E Rev B (SAMx21Exxx)
- MattairTech MT-D21E Rev A (SAMD21ExxA)
- MattairTech MT-D11 (SAMD11D14AM)
- MattairTech Generic D11C14A
- MattairTech Generic D11D14AS (coming soon)
- MattairTech Generic D11D14AM (coming soon)
- MattairTech Generic x21E (coming soon)
- MattairTech Generic x21G (coming soon)
- MattairTech Generic x21J (coming soon)
- Arduino Zero (arduino.cc)
- Arduino M0 (arduino.org)

Tools Menu Additions

Depending on the board variant, different menu options will appear in the Tools menu.

Microcontroller Menu

This menu will appear with boards that have multiple microcontroller options.

Clock Source Menu

There are up to four clock source choices, depending on board variant and microcontroller. They are:

- 32KHZ_CRYSTAL (default)
- HIGH_SPEED_CRYSTAL
- INTERNAL_OSCILLATOR
- INTERNAL_USB_CALIBRATED_OSCILLATOR

See Clock Source section for more information.

Bootloader Size Menu

With the D21, L21, and C21, the bootloader size can be configured as:

- 8KB_BOOTLOADER (default)
- 16KB_BOOTLOADER
- NO_BOOTLOADER

With the D11, the bootloader size can be configured as:

- 4KB_BOOTLOADER (default)
- NO_BOOTLOADER

Choose NO_BOOTLOADER if not using a bootloader (an external programmer will be used for sketch upload).

Serial Config Menu

This menu is used to select different combinations of serial peripherals. It adds additional UART, SPI, and WIRE instances. This is also useful for the D11, which has a reduced pin count and number of SERCOMs. It can also be used to reduce FLASH and SRAM usage by selecting fewer UART peripherals, which are instantiated in the core, rather than only when including a library (like SPI and WIRE). Note that with options where there is more than one SPI or WIRE, the additional instances will consume a small amount of RAM, but neither the peripheral nor the pins are configured until begin() method is called (thus, the pins can be used for other purposes).

Use the ASCII art rendering at the top of the README.md file of the board variant used in order to determine the mapping of instances to pins. When USB CDC is enabled, Serial refers to SerialUSB, otherwise it refers to Serial1 (TX1/RX1).

USB Config Menu

This menu will appear with all microcontrollers except the C21, which does not have USB. The options are:

- CDC_ONLY (default)
- CDC_HID
- WITH_CDC
- HID_ONLY
- WITHOUT_CDC
- USB_DISABLED

Choose an option that best matches your code and library usage. Each option results in a different USB PID. Choose an option with CDC if you want auto-reset to function, or the serial monitor over USB. If CDC is not enabled, Serial will refer to Serial1 instead of SerialUSB. These options can be used to optimize FLASH and SRAM usage by allowing CDC to be disabled (or USB completely disabled).

Clock Source

There are up to four clock source choices, depending on board features and microcontroller. Since currently the cpu must run at 48MHz, the PLL or DFLL must be used (the SAMC can use OSC48M).

- **32KHZ_CRYSTAL (default)**
  - Uses both XOSC32K and FDPLL96M
  - High long-term accuracy, slow startup, medium current (PLL)
- **HIGH_SPEED_CRYSTAL**
  - Uses both XOSC and FDPLL96M
  - High accuracy, medium startup, high current (XOSC and PLL)
- **INTERNAL_OSCILLATOR**
  - Uses DFLL48M in open-loop mode (SAMC uses OSC48M)
  - Low accuracy, fast startup, medium-low current (low current with SAMC)
- **INTERNAL_USB_CALIBRATED_OSCILLATOR** (not available with SAMC)
  - Uses DFLL48M in closed-loop mode
  - High accuracy, medium-fast startup, medium current

External 32.768KHz Crystal

The PLL will be used with the 32.768KHz crystal. PLL_FRACTIONAL_ENABLED can be defined, which will result in a more accurate 48MHz output frequency at the expense of increased jitter.

External High-Speed Crystal

HS_CRYSTAL_FREQUENCY_HERTZ must be defined with the external crystal frequency in Hertz. The crystal frequency must be between 400000Hz and 32000000Hz. The PLL will be used.

PLL_FRACTIONAL_ENABLED can be defined, which will result in a more accurate 48MHz output frequency at the expense of increased jitter. If PLL_FAST_STARTUP is defined, the crystal will be divided down to 1MHz - 2MHz, rather than 32KHz - 64KHz, before being multiplied by the PLL. This will result in a faster lock time for the PLL, however, it will also result in a less accurate PLL output frequency if the crystal is not divisible (without remainder) by 1MHz. In this case, define PLL_FRACTIONAL_ENABLED as well. By default, PLL_FAST_STARTUP is disabled. PLL_FAST_STARTUP is also useful for USB host mode applications. See datasheet USB electrical characteristics. The crystal frequency must be at least 1000000Hz when PLL_FAST_STARTUP is defined.

Internal Oscillator

The DFLL will be used in open-loop mode, except with the C21 which lacks a DFLL, so the internal 48MHz RC oscillator is used instead. NVM_SW_CALIB_DFLL48M_FINE_VAL is the fine calibration value for DFLL open-loop mode. The coarse calibration value is loaded from NVM OTP (factory calibration values).

Internal Oscillator with USB Calibration

This is available for the D21, D11, or L21. It will also use the DFLL in open-loop mode, except when connected to a USB port with data lines (and not suspended), then it will calibrate against the USB SOF signal. NVM_SW_CALIB_DFLL48M_FINE_VAL is the fine calibration value for DFLL open-loop mode. The coarse calibration value is loaded from NVM OTP (factory calibration values).

Clock Generators Currently Used

0. MAIN (mcu)
1. XOSC (high speed crystal)
2. OSCULP32K (initialized at reset for WDT on D21 and D11)
3. OSC_HS (the reset default internal RC oscillator is put here at 8MHz, except with C21)

Analog Reference

D21
* AR_DEFAULT uses 1/2X gain on each input.
* The external reference should be between 1.0V and VDDANA-0.6V.

L21
* AR_DEFAULT = AR_INTERNAL_INTVCC2
* Both AR_INTREF and AR_INTERNAL1V0 has the same effect as AR_INTREF_1V0.
* The external reference should be between 1v and VDDANA-0.6v=2.7v.

C21
* AR_DEFAULT = AR_INTERNAL_INTVCC2
* Both AR_INTREF and AR_INTERNAL1V0 has the same effect as AR_INTREF_1V024.
* The external reference should be between 1v and VDDANA-0.6v=2.7v.

Warning : The maximum reference voltage is Vcc (up to 3.6 volts for the SAMD/SAML, 5V for the SAMC)

Reference Selection Table

D21 / D11	Volts	L21	Volts	C21	Volts
AR_DEFAULT	1/2 VCC	AR_DEFAULT	VCC	AR_DEFAULT	VCC
AR_INTERNAL1V0	1.00V	AR_INTREF	1.00V	AR_INTREF	1.024V
AR_INTERNAL_INTVCC0	1/1.48 VCC	AR_INTREF_1V0	1.00V	AR_INTREF_1V024	1.024V
AR_INTERNAL_INTVCC1	1/2 VCC	AR_INTREF_1V1	1.10V	AR_INTREF_2V048	2.048V
AR_EXTERNAL_REFA	REFA	AR_INTREF_1V2	1.20V	AR_INTREF_4V096	4.096V
AR_EXTERNAL_REFB	REFB	AR_INTREF_1V25	1.25V	AR_INTERNAL1V0	1.024V
---		AR_INTREF_2V0	2.00V	AR_INTERNAL_INTVCC0	1/1.6 VCC
---		AR_INTREF_2V2	2.20V	AR_INTERNAL_INTVCC1	1/2 VCC
---		AR_INTREF_2V4	2.40V	AR_INTERNAL_INTVCC2	VCC
---		AR_INTREF_2V5	2.50V	AR_EXTERNAL_REFA	REFA
---		AR_INTERNAL1V0	1.00V	AR_EXTERNAL_DAC	DAC
---		AR_INTERNAL_INTVCC0	1/1.6 VCC		
---		AR_INTERNAL_INTVCC1	1/2 VCC		
---		AR_INTERNAL_INTVCC2	VCC		
---		AR_EXTERNAL_REFA	REFA		
---		AR_EXTERNAL_REFB	REFB		

Common Settings

* AR_INTERNAL = AR_INTERNAL_INTVCC0
* AR_INTERNAL2V23 = AR_INTERNAL_INTVCC0
* AR_INTERNAL1V65 = AR_INTERNAL_INTVCC1
* AR_EXTERNAL = AR_EXTERNAL_REFA

When using AR_INTERNAL2V23 or AR_INTERNAL1V65, these voltages are correct only when Vcc = 3.3V)

Chip Specific Notes

SAMD21

- When USB is disabled, pullups will be enabled on PA24 and PA24 to avoid excessive current consumption (<1mA) due to floating pins. Note that it is not necessary to enable pull resistors on any other pins that are floating. Errata: Disable pull resistors on PA24 and PA25 manually before switching to a peripheral.

SAML21

- There are two DACs, DAC0 and DAC1. Both are supported. Because changing the configuration of one DAC requires disabling both, there will be about a 40us period when the second DAC is disabled. Most of this time is due to an errata that requires a delay of at least 30us when turning off the DAC while refresh is on. The L21 DACs have a refresh setting which must be enabled in this core.
- The analog reference has additional options on the L21 and C21. See Analog Reference section.
- On the L21, SERCOM5 is in a low power domain. The Fm+ and HS modes of I2C (wire) are not supported.
- The SAML and SAMC have double-buffered TCs, which are supported in the core.
- The CHANGE and RISING interrupt modes on pin A31 do not seem to work properly on the L21.
- The L21 has two performance levels that affect power consumption. During powerup, the L21 starts at the lowest performance level (PL0). The startup code changes to the highest performance level (PL2) in order to support 48MHz and USB (among other things).
- Two Flash Wait States are inserted for the L21 and C21 (the D21/D11 use one wait state).
- pinPeripheral now handles disabling the DAC (if active). Note that on the L21, the DAC output would interfere with other peripherals if left enabled, even if the anaolog peripheral is not selected.

SAMC21

- There are two SAR ADCs. Both are supported. The PinDescription table determines the peripheral instance and pin mapping.
- The analog reference has additional options on the L21 and C21. See Analog Reference section.
- The SAML and SAMC have double-buffered TCs, which are supported in the core.
- Two Flash Wait States are inserted for the L21 and C21 (the D21/D11 use one wait state).
- The C21 requires internal pull resistors to be activated on floating pins to minimize power consumption (not needed on D21/D11 or L21).
- The C21 uses the minimum sampling time so that rail-to-rail and offset compensation works. Offset compensation adds 3 ADC clock cycles, so the total is 4 clock cycles. The D21, D11, and L21 use the maximum sampling time.

SAMD11

- The D11D has three SERCOM. The D11C has two sercom (no sercom2).
- TONE: TC5 does not exist on the D11. Using TC2 instead (TC1 on the D11C14 as TC2 is not routed to pins). It will conflict with the 2 associated TC analogWrite() pins.
- When USB is disabled, pullups will be enabled on PA24 and PA24 to avoid excessive current consumption (<1mA) due to floating pins. Note that it is not necessary to enable pull resistors on any other pins that are floating. Errata: Disable pull resistors on PA24 and PA25 manually before switching to a peripheral.

Reducing SRAM/FLASH Usage on the D11

TODO

Differences Between MattairTech and Arduino Cores

- Communications interfaces are mostly unchanged, including USB
- All pins have high drive strength enabled by default
- All pins (digital and analog) setup in STARTUP mode (enable INEN and set default pull direction to pullup (pullup will not be enabled))
- INEN enabled for both input and output (but not analog)
- pinPeripheral now handles disabling the DAC (if active). Note that on the L21, the DAC output would interfere with other peripherals if left enabled, even if the analog peripheral is not selected.
- Pull resistors enabled only if pin attributes allow and only if pin is not configured as output.
- Pull direction (pullup or pulldown) is now set with pinMode only (defaults to pullup if pinMode never called).
- At least on the L21, pin A31 must be set as an input. It is possible that debugger probe detection is being falsely detected (even with a pullup on A31 (SWCLK)), which would change the peripheral mux of A31 to COM. This might not normally be a problem, but one strange effect is that Serial2 loses characters if pin A31 is not set as INPUT. So, the startup code calls pinMode(31, INPUT).
- Todo: Table summarizing which core files are modified and by how much
- Todo: List changes due to adding/changing features vs porting to new chip

Serial Monitor

To print to the Serial Monitor over USB, use 'Serial'. Serial points to SerialUSB (Serial1 and Serial2 are UARTs). Unlike most Arduino boards (ie. Uno), SAMD boards do not automatically reset when the serial monitor is opened. To see what your sketch outputs to the serial monitor from the beginning, the sketch must wait for the SerialUSB port to open first. Add the following to setup():

```
while (!Serial) ;
```

Remember that if the sketch needs to run without SerialUSB connected, another approach must be used. You can also reset the board manually with the Reset button if you wish to restart your sketch. However, pressing the Reset button will reset the SAMD chip, which in turn will reset USB communication. This interruption means that if the serial monitor is open, it will be necessary to close and re-open it to restart communication.

When USB CDC is not enabled, Serial will instead refer to Serial1, which is the first UART.

Code Size and RAM Usage (1.6.5-mt2)

TODO: Update this. Maybe just for D11 and move to D11 Chip Specific Notes.

Sketch and Configuration	*MT-D21E (Code + RAM)*	*MT-D11 (Code + RAM)*
Blink (CDC + HID + UART)	7564 + 1524	7452 + 1424
Blink (CDC + UART)	6588 + 1496	6484 + 1396
Blink (CDC Only)	5248 + 1304	5192 + 1300
Blink (UART Only)	3828 + 336	3716 + 236
Blink (No USB or UART)	2472 + 144	2416 + 140
Datalogger (No USB or UART)	10340 + 948	10260 + 944

- 180 bytes of flash can be saved on the MT-D11 by using PIN_MAP_COMPACT (see 'New PinDescription Table' below).
- Datalogger compiled without USB or UART support, but with SPI and SD (with FAT filesystem) support. Serial output was disabled.
- Note that USB CDC is required for auto-reset into the bootloader to work (otherwise, manually press reset twice in quick succession).
- USB uses primarily 3 buffers totaling 1024 bytes. The UART uses a 96 byte buffer. The

banzai() function (used for auto-reset) resides in RAM and uses 72 bytes.
- Any combination of CDC, HID, or UART can be used (or no combination), by using the Tools->Communication menu.

Detailed Memory Usage Output After Compilation

The flash used message at the end of compilation is not correct. The number shown represents the .text segment only. However, Flash usage = .text + .data segments (RAM usage = .data + .bss segments). In this release, two programs are run at the end of compilation to provide more detailed memory usage. To enable this output, go to File->Preferences and beside "Show verbose output during:", check "compilation".

Just above the normal flash usage message, is the output from the size utility. However, this output is also incorrect, as it shows .text+.data in the .text field, but 0 in the .data field. However, the .text field does show the total flash used. The .data field can be determined by subtracting the value from the normal flash usage message (.text) from the value in the .text field (.text+.data). The .bss field is correct.

Above the size utility output is the output from the nm utility. The values on the left are in bytes. The letters stand for: T(t)=.text, D(d)=.data, B(b)=.bss, and everything else (ie: W) resides in flash (in most cases).

Installation

Driver Installation

Windows

Prior to core version 1.6.6-mt1, sketches compiled with both CDC and HID USB code by default, thus requiring a CDC driver for the bootloader and a CDC-HID driver for sketches. Now that PluggableUSB is supported, sketches compile with only CDC code by default. Thus, only one driver is needed. Since HID and MIDI are currently supported (and MSD potentially in the future), driver installation will be required for each different combination of USB devices. There are currently four USB composite device combinations that include CDC as well as a CDC only device. Each supported combination has a unique USB VID:PID pair, and these are listed in the .inf file. Once the first device is installed (the CDC only device), future installations *might* be automatic, otherwise, you may direct the installer to the same .inf file. The drivers are signed and support both 32 and 64 bit versions of Windows XP(SP3), Vista, 7, 8, and 10. Note that the Windows 10 generic CDC drivers work as well.

1. If you do not already have the SAM-BA bootloader installed, see below.
2. Download https://www.mattairtech.com/software/MattairTech_CDC_Driver_Signed.zip and unzip into any folder. Note that the Windows 10 generic CDC drivers work as well.
3. Plug in the board. The LED should fade when the bootloader is running (or blink if the test sketch is running).
4. Windows will detect the board. Point the installer to the folder from above to install the bootloader driver.
5. If you don't intend on using Arduino, you can skip the rest of this list. See Using Bossac Standalone below.
6. If you do not already have the test firmware installed (comes preinstalled), see Using Bossac Standalone below.
7. Press the reset button to run the test firmware (if needed). The LED will blink.
8. Windows will detect the board. Point the installer to the above folder to install the sketch driver (if needed).
9. Continue with SAM M0+ Core Installation below.

Linux

0. No driver installation is needed.
1. On some distros, you may need to add your user to the same group as the port (ie: dialout) or set udev rules (See the file https://github.com/mattairtech/ArduinoCore-samd/tree/master/drivers/99-mattairtech-USB-CDC.rules).
2. You MAY have to install and use Arduino as the root user in order to get reliable access to the serial port.
    - This is true even when group permissions are set correctly, and it may fail after previously working.
    - You can also create/modify a udev rule to set permissions on the port so *everyone* can

read / write.

3. If you are running modemmanager (ie: Ubuntu), disable it, or use the udev rules file above.
4. Continue with SAM M0+ Core Installation below.

OS X

OS X support was added in version 1.6.7-beta-b0.

0. No driver installation is needed.
1. Plug in the board. You may get a dialog box asking if you wish to open the “Network Preferences”:
    - Click the "Network Preferences..." button, then click "Apply".
    - The board will show up as “Not Configured”, but it will work fine.
2. Continue with SAM M0+ Core Installation below.

SAM M0+ Core Installation

- To update from a previous version, click on MattairTech SAM M0+ Boards in Boards Manager, then click Update.

1. The MattairTech SAM M0+ Core requires Arduino IDE 1.6.7 or above (including 1.8.x).
2. In the Arduino IDE, click File->Preferences.
3. Click the button next to Additional Boards Manager URLs.
4. Add https://www.mattairtech.com/software/arduino/package_MattairTech_index.json.
5. Save preferences, then open the Boards Manager.
6. Install the Arduino SAM M0+ Boards package. Use version 1.6.2 or higher.
7. Install the MattairTech SAM M0+ Boards package.
8. Close Boards Manager, then click Tools->Board->(choose board).
9. Select the MCU with the now visible Tools->Microcontroller menu (if present).
10. If you do not already have the bootloader or blink sketch installed, see SAM-BA USB CDC Bootloader below.
11. Plug in the board. The blink sketch should be running.
12. Click Tools->Port and choose the COM port. Note that the board indicated may not match the chosen board*
13. You can now upload your own sketch.

** Currently, with MattairTech boards, USB PIDs are shared across boards (but they are different based on Tools->USB Config). This will result in Tools->Port showing "MattairTech MT-D21E (rev B)" for all MattairTech boards.*

Uploading the First Sketch

1. In the Arduino IDE (1.6.7 or above), open File->Examples->01.Basics->Blink.
2. Change the three instances of '13' to 'LED_BUILTIN'.
3. Be sure the correct options are selected in the Tools menu (see Core Installation above).
4. With the board plugged in, select the correct port from Tools->Port.
5. Click the Upload button. After compiling, the sketch should be transferred to the board.
6. Once the bootloader exits, the blink sketch should be running.

Beta Builds

Periodically, a beta is released for testing.

The beta builds are available through Boards Manager. If you want to install them:

1. Open the **Preferences** of the Arduino IDE.
2. Add this URL https://www.mattairtech.com/software/arduino/beta/package_MattairTech_index.json in the **Additional Boards Manager URLs** field, and click OK.
3. Open the **Boards Manager** (menu Tools->Board->Board Manager...)
4. Install **MattairTech SAM M0+ Boards - Beta build**
5. Select one of the boards under **MattairTech SAM M0+ Beta Build XX** in Tools->Board menu
6. Compile/Upload as usual

The Arduino IDE will notify the user if an update to the beta is available, which can then be installed automatically. Alternatively, if a particular beta is needed, replace the url in step 2 with: https://www.mattairtech.com/software/arduino/beta/package_MattairTech_sam_m0p-${VERSION}-beta-b${BUILD_NUMBER}_index.json where ${VERSION} and ${BUILD_NUMBER} match the beta name as shown in the CHANGELOG (ie: package_MattairTech_sam_m0p-1.6.7-beta-b0_index.json). In this case, the IDE will not notify the user of updates.

New PinDescription Table

Technical information on the new PinDescription table format is now in the README.md that accompanies each board variant. See board variants above.

Note that a new column (GCLKCCL) was added for 1.6.8-beta-b0.

MATTAIRTECH_ARDUINO_SAMD_VARIANT_COMPLIANCE in variant.h is used to track versions. If using board variant files with the old format, the new core will still read the table the old way, losing any new features introduced by the new column. Additionally, new definitions have been added for L21 and C21 support.

Each pin can have multiple functions.

The PinDescription table describes how each of the pins can be used by the Arduino core. Each pin can have multiple functions (ie: ADC input, digital output, PWM, communications, etc.), and the PinDescription table configures which functions can be used for each pin. This table is mainly accessed by the pinPeripheral function in wiring_private.c, which is used to attach a pin to a particular peripheral function. The communications drivers (ie: SPI, I2C, and UART), analogRead(), analogWrite(), analogReference(), attachInterrupt(), and pinMode() all call pinPeripheral() to verify that the pin can perform the function requested, and to configure the pin for that function. Most of the contents of pinMode() are now in pinPeripheral().

Pin Mapping

There are different ways that pins can be mapped. Typically, there is no relation between the arduino pin number used, and the actual port pin designator. Thus, the pcb must be printed with the arduino numbering, otherwise, if the port pin is printed, a cross reference table is needed to find the arduino pin number. However, this results in the least amount of space used by the table. Another method, used by default by the MT-D21E and MT-D11, maps Arduino pin numbers to the actual port pin number (ie: Arduino pin 28 = Port A28). This works well when there is only one port (or if the PORTB pins are used for onboard functions and not broken out). PIO_NOT_A_PIN entries must be added for pins that are used for other purposes or for pins that do not exist (especially the D11), so some FLASH space may be wasted. For an example of both types, see variant.cpp from the MT-D11 variant.

See Board Variants above for more technical information on the PinDescription table.

See WVariant.h for the definitions used in the table.

MattairTech Libraries

Available Now

Use Libraries Manager to install

- SRAM_23LC - Library for Microchip Technology Inc. 23LC (23LCV, 23A, 23K) SPI SRAM chips
  - Byte and block transfers
- EEPROM_CAT25 - Library for On Semiconductor CAT25 SPI EEPROM chips
  - Byte, block, and page transfers

Under Development

- ZeroTimers - 8/16/24/32 bit timer library with API based on TimerOne
  - PWM
  - interrupt
  - Input capture
- POWER_PAC1921 - Library for Microchip Technologies high-side power/current/voltage monitor with I2C and analog out

- SENSOR_LPS22HB - Library for ST MEMS nano pressure sensor / temperature sensor with I2C
- SENSOR_LSM6DS3H - Library for ST iNemo inertial module: 3D accelerometer / 3D gyroscope with I2C and interrupt
- FLASH_AT25 - Library for Adesto Technologies AT25 SPI serial FLASH devices

Possible Future

- Several I2C (Wire) sensor devices
- TFT LCD (CFAF128128B-0145T)
- IR decoder
- I2S DAC/AMP and I2S MEMS microphone
- Battery management IC
- XBee/Xbee Pro devices?

Core Future Additions/Changes

Under Development

- PlatformIO support
- SAM D51 (M4F) support (pin compatible with D21, similar peripherals and pin mapping)
- Fix programming port for Arduino Zero and M0 board variants
- Reduce SRAM usage by USB endpoint buffers by only allocating endpoints actually used (D11 especially)

Possible Future

- Features for lower power consumption (library?) Fall 2017?
- Reliability and security enhancements
- USB Host mode CDC ACM (partially complete; BSD-like license?)
- SD card library? Port of FatFS and/or Petit FatFS?
- Optional use of single on-board LED as USB activity LED
- MSC (Mass Storage) USB Device Class
- Polyphonic tone
- Wired-AND, Wired-OR for port pins
- High-speed port pin access (IOBUS)

ChangeLog

The Changelog has moved to a separate file named CHANGELOG. The most recent changes are still in the 'What's New' section above.

Troubleshooting

- **Tools->Port shows wrong board**
    - Currently, with MattairTech boards, USB PIDs are shared across boards (but they are different based on Tools->USB Config). This will result in Tools->Port showing "MattairTech MT-D21E (rev B)" for all MattairTech boards.
- **Tools->USB Config menu**
    - Currently, the Tools->USB Config menu (was Tools->Communications) must be used to select the communications configuration. This configuration must match the included libraries. For example, when including the HID and Keyboard libraries, you must select an option that includes HID (all options except CDC_ONLY or USB_DISABLED). This menu is currently needed to select the USB PID that matches the USB device configuration (needed for some versions of Windows). It is also used to control if CDC support is compiled (CDC is always enabled in the stock Arduino core). Auto reset requires CDC to be enabled.
        - Be sure that the Tools->Communications menu matches the sketch and libraries you are compiling.
        - Different combinations of USB devices will result in different COM port assingments in Windows.
- **Incude platform specific libraries**
    - You may need to manually include platform specific libraries such as SPI.h, Wire.h, and HID.h.
- **Errors when compiling, uploading, or burning the bootloader**
    - Be sure to install the Arduino samd core before installing the MattairTech sam m0+ core. If you have problems upgrading the IDE to 1.6.6, you may need to uninstall both the Arduino and MattairTech cores, then re-install in the proper order. Use Arduino core 1.6.2 or above.
- **On Linux, disable modem manager (Ubuntu)**
- **Do not perform a manual auto-reset** (using a terminal program to change baud to 1200)
- **Boards Manager must be opened twice to see some updates** (only applies to some old IDE versions)
- **Boards manager might not install/uninstall the core or tools properly if the contents of the arduino15 directory has been manually modified**
    - Be sure to delete all manually installed folders (not just files)

SAM-BA USB CDC Bootloader (Arduino Compatible)

This bootloader is based on the Arduino Zero bootloader which is a part of the Arduino SAMD core. It provides a USB-CDC and/or TTL serial communications interface to a host running the bossac command line firmware programming utility (or the Arduino IDE) running on Windows, Linux, or OS X. Optionally, SD Card firmware loading is supported, using SDSC or SDHC cards with a FAT16 or FAT32 filesystem. This version adds support for the D11, L21, and C21 microcontrollers. It also adds support for four different clock sources (two external crystals and two internal oscillator options). There are additional board definitions added, and binaries for most board/chip combinations are pre-built.

See bootloaders/zero/README.md for more technical information on the bootloader.

Features

- SAM-BA USB CDC and UART interfaces with optional terminal mode
- SD Card interface (both USB CDC and SD Card support fits in 8KB)
- Four different clock sources (two external crystals and two internal oscillator options)
- Arduino IDE auto-reset and double-tap reset button support
- Arduino extended commands for faster firmware loading
- Supports the D21, L21, C21, and D11 SAM M0+ chips
- Bossac command line utility for Windows, Linux, and OS X

The bootloader can be started by:

- Tapping reset twice in quick succession (BOOT_DOUBLE_TAP).
- Holding down button A (BOOT_LOAD_PIN) while powering up.
- Clicking 'Upload Sketch' in the Arduino IDE, which will automatically start the bootloader (when CDC is enabled).
- If the application (sketch) area is blank, the bootloader will run.

Otherwise, it jumps to application and starts execution from there.
The LED will PWM fade during bootloader execution.

Bossac

Bossac is a command line utility for uploading firmware to SAM-BA bootloaders. It runs on Windows. Linux, and OS X. It is used by Arduino to upload firmware to SAM and SAM M0+ boards. The version described here adds to the Arduino version (https://github.com/shumatech/BOSSA, Arduino branch), which in turn is a fork from the original Bossa (http://www.shumatech.com/web/products/bossa). It adds support for more SAM M0+ chips (D21, L21, C21, and D11). Note that only the Arduino or Mattairtech versions of bossac are currently supported for SAM M0+ chips. Neither the stock bossac (or Bossa) nor the Atmel SAM-BA upload tool will work.

Bootloader Firmware Installation

Bootloader Installation Using the Arduino IDE

1. If you do not already have the MattairTech SAM M0+ core installed, see SAM M0+ Core Installation above.
2. Plug in the SAM M0+ board. The bootloader must be running to (press reset twice within 500ms).
3. Plug an Atmel ICE into USB, then connect it to the powered SAM M0+ board. A green LED should light on the Atmel ICE.
4. Click Tools->Programmer->Atmel ICE.
5. Click Tools->Board->MattairTech MT-D21E (or whichever board you are using).
6. Click Tools->Microcontroller and select your MCU (if menu present).
7. Click Tools->Burn Bootloader. Ignore any messages about not supporting shutdown or reset.
8. Continue with driver installation above.

A running sketch *may* interfere with the bootloader installation process. Be sure you are running the existing bootloader or using a blank chip.

Bootloader Installation Using Another Tool (ie: Atmel Studio, openocd)

1. Download the bootloader from https://www.mattairtech.com/software/arduino/SAM-BA-bootloaders-zero-mattairtech.zip.
2. Unzip to any directory. Be sure that a bootloader is available for your particular MCU.
3. Follow the procedures for your upload tool to upload the firmware.
    - Perform a chip erase first. Be sure no BOOTPROT bits are set.
    - Install the binary file to 0x00000000 of the FLASH.
    - You can optionally set the BOOTPROT bits to 8KB (or 4KB for the MT-D11). The Arduino installation method does not set these.
    - You can optionally set the EEPROM bits or anything else. The Arduino installation method uses factory defaults.
4. Continue with driver installation above.

Bootloader Binaries

The bootloaders/zero/binaries directory contains the SAM-BA m0+ bootloaders built by the build_all_bootloaders.sh script from the 'MattairTech SAM M0+ Boards' Arduino core, which is available at https://github.com/mattairtech/ArduinoCore-samd. Each board and chip combination has two bootloaders available:

SAM-BA interface only

- USB CDC only for all MattairTech boards

- Both USB CDC and UART for most Arduino boards
- The Generic board variants minimize external pin usage
  - Only the SAM-BA interface pins are used (no crystal, LED, etc.)
- Filename is: sam_ba_$(BOARD_ID)_$(MCU)

SAM-BA interface and SD Card interface

- USB CDC only for all Arduino and most MattairTech boards
- No SAM-BA interface for the D11 chips
- All board variants define SDCARD_USE_PIN1 (except D11)
- The Generic board variants use the LED
- SDCARD_AUTORUN_DISABLED is defined
- Filename is: sam_ba_sdcard_$(BOARD_ID)_$(MCU)

MattairTech Boards

MattairTech boards are all configured with only one interface: SAM_BA_USBCDC_ONLY (except C21, which uses SAM_BA_UART_ONLY). CLOCKCONFIG_CLOCK_SOURCE is set to CLOCKCONFIG_INTERNAL_USB (CLOCKCONFIG_INTERNAL for the C21). Only the main LED is defined. BOOT_LOAD_PIN is not defined, but BOOT_DOUBLE_TAP_ENABLED is. When the SD Card interface is enabled, SDCARD_AUTORUN_DISABLED and SDCARD_USE_PIN1 are defined.

MattairTech/Generic D11 Boards

All boards are configured with only the USB CDC interface, except when SDCARD_ENABLED is defined, then only the SD Card interface is enabled. ARDUINO_EXTENDED_CAPABILITIES is set to 0 (disabled). TERMINAL_MODE_ENABLED is not defined. As of 1.6.8-beta-b2, USB_VENDOR_STRINGS_ENABLED is now defined. BOOT_LOAD_PIN is not defined, but BOOT_DOUBLE_TAP_ENABLED is. When the SD Card interface is enabled, SDCARD_AUTORUN_DISABLED is defined (but not SDCARD_USE_PIN1).

Generic Boards

The generic boards are all configured to minimize external hardware requirements. Only one interface is enabled: SAM_BA_USBCDC_ONLY (except C21, which uses SAM_BA_UART_ONLY). CLOCKCONFIG_CLOCK_SOURCE is set to CLOCKCONFIG_INTERNAL_USB (CLOCKCONFIG_INTERNAL for the C21), so no crystal is required. No LEDs are defined. BOOT_LOAD_PIN is not defined, but BOOT_DOUBLE_TAP_ENABLED is, since it uses the reset pin. When the SD Card interface is enabled, SDCARD_AUTORUN_DISABLED and SDCARD_USE_PIN1 are defined.

Bossac Utility

This version of the bootloader requires bossac (1.7.0-mattairtech-1) or above.

See the MattairTech SAM M0+ Core README.md "Driver Installation" for installation instructions.

Bossac Utility Installation

If using the Arduino IDE to upload firmware, then this will be installed automatically when intalling the core. If using Bossac standalone, download bossac directly at:

- https://www.mattairtech.com/software/arduino/bossac-1.7.0-mattairtech-1-mingw32.tar.gz (Windows 32 bit and 64 bit)
- https://www.mattairtech.com/software/arduino/bossac-1.7.0-mattairtech-1-x86_64-linux-gnu.tar.gz (Linux 64 bit)
- https://www.mattairtech.com/software/arduino/bossac-1.7.0-mattairtech-1-i686-linux-gnu.tar.gz (Linux 32 bit)
- https://www.mattairtech.com/software/arduino/bossac-1.7.0-mattairtech-1-x86_64-apple-darwin.tar.gz (OS X 64 bit)

Linux 64 bit users can also download Bossa (GUI) and bossash (shell) from:

- https://www.mattairtech.com/software/arduino/Bossa-1.7.0-mattairtech-1-x86_64-linux-gnu.tar.gz (Linux 64 bit)

Note that the SAM-BA tools from Atmel will not work, and the version of bossac from the Arduino SAMD Core currently does not support the L21, C21, or D11 (but it does support the D21).

Using Bossac Standalone

TODO: Update https://www.mattairtech.com/software/SAM-BA-bootloader-test-firmware.zip with new chips (L21 and C21).

When using Bossac standalone, you will need to ensure that your application starts at 0x00002000 for 8 KB bootloaders, and 0x00001000 for 4 KB bootloaders. This is because the bootloader resides at 0x00000000. This can be accomplished by passing the following flag to the linker (typically LDFLAGS in your makefile; adjust for your bootloader size):

```
-Wl,--section-start=.text=0x2000
```

You can also use a linker script. See the MattairTech SAM M0+ package for examples. Be sure to generate and use a binary file. Many makefiles are set up to generate an elf, hex, and bin already.

As an example, bossac will be used to upload the test firmware (blink sketch):

1. Download firmware from https://www.mattairtech.com/software/SAM-BA-bootloader-test-firmware.zip and unzip.

2. If you have not already installed the bootloader driver, see Driver Installation above.
3. Be sure there is a binary that matches your chip. On the command line (change the binary to match yours):

```
bossac.exe -d --port=COM5 -U true -i -e -w -v Blink_Demo_ATSAMD21E18A.bin -R
```

4. On Linux --port might be /dev/ttyACM0. If the device is not found, remove the --port argument for auto-detection.
5. See http://manpages.ubuntu.com/manpages/vivid/man1/bossac.1.html for details.
6. The board should reset automatically and the sketch should be running.

See bootloaders/zero/README.md for more technical information on the bootloader.

USB Mass Storage Bootloader

This currently describes MT-D21E boards only. Please contact me and I can add Xeno support.

Source code and binaries available at https://github.com/mattairtech/SAMD-MSD-Bootloader.

A USB Mass Storage Class device (MSC or MSD) bootloader can be optionally installed. This will allow programming of the FLASH without an external programmer. Additionally, no special software is required on the host computer. The bootloader occupies the first 16KB of FLASH, leaving the rest for the user firmware. The BOOTPROT fuse bits (2:0) can be set 0x01, which will protect the first 16KB of FLASH from internal or external programming (from 0x00000000 to 0x00004000). Note that the MSD bootloader does not use an external crystal, as it uses USB clock recovery (DFLL tuned using the USB SOF signal).

Installation

The Mass Storage Bootloader can be installed using atprogram.exe (included with Atmel Studio) with the following commands (ATSAMD21E18A shown):

```
atprogram -t atmelice -i SWD -d atsamd21e18a -cl 500khz program -c --verify -f
c:\msd_bootloader_256_flash.hex
```

The three BOOTPROT fuse bits (2:0) can be set to 0x01 (16KB):

```
atprogram -t atmelice -i SWD -d atsamd21e18a -cl 500khz write -fs -o 0x00804000 --values f9
```

The blink program (compiled with an offset of 0x00004000) can then be installed using the Mass Storage Bootloader.

Special Requirements when Compiling Software

- Because the user firmware will begin executing at FLASH byte address 0x00004000, you must pass the following flag to the linker (typically LDFLAGS in your makefile):

  ```
  -Wl,--section-start=.text=0x4000
  ```

- If using the MattairTech SAM M0+ Arduino core, be sure to select 16KB bootloader size in the Tools->Bootloader Size menu.
- Be sure to generate a binary file. Most makefiles are set up to generate an elf, hex, and bin already. You will need the bin file.
- You will need to rename the binary file to FLASH.BIN.

Entering the bootloader and programming the firmware

- Enter the bootloader by pressing button A while powering up the board from USB. Or, hold button A while pressing and releasing button B (if configured as RST). Button A must be connected to pin A27 via solder jumper J13 (this will already be soldered if you ordered the

bootloader option). Note that when no user firmware is installed, the bootloader will not automatically run, so you must always use the bootloader button. When the bootloader is run for the first time, the host operating system may take a small amount of time to install drivers. Drivers are already included with the OS, so there is nothing more to download. Once loaded, the LED will begin blinking at 2Hz.

- Mount the “FLASH disk” if it is not mounted automatically. The only file on the entire volume will be FLASH.BIN. This file represents the entire FLASH contents and will always exist. The file date will always be the same upon mounting (2/14/1989). You can read this file simply by copying it to your hard drive. It will include the installed firmware plus 0xFF for the remainder of the file (up to the end of the FLASH).
- Program the FLASH by copying your new FLASH.BIN over the existing copy on the “FLASH disk”. On Windows, you can do this with a file manager. On OS-X (and possibly Linux), you will need to use the cp command, which should already be present. Open up a console (Terminal on OS-X) and type (adjust for your system):

```
cp FLASH.BIN '/run/media/cygnus/MT-D21E MSD'
```

- Be sure to unmount the volume before running your new firmware, so that any disk caches are flushed.
- To run your firmware, simply reset or cycle power without pressing button A.
- Technical notes: The startup portion of the bootloader will run prior to executing your firmware. This startup code will enable the button A pullup resistor, wait 8ms for the debouncing capacitor to charge, then test the state of the button. If it is not pressed, the user firmware will be executed as follows:
    - The stack pointer location will be rebased to 0x00004000
    - The interrupt vector table will be rebased to (0x00004000 & SCB_VTOR_TBLOFF_Msk)
    - A jump will be perfomed to the user firmware reset vector.

Blink Demo

This currently describes MT-D21E Rev A boards only. Xeno will have a test program soon.

A demo program comes pre-installed. It simply blinks the LED at 1Hz using an internal clock source. The hex files can be found on the Xeno product page at https://www.mattairtech.com/. The blink demo was compiled using the Atmel Standalone Toolchain for Linux. It makes use of Atmel Software Framework (ASF) so it is rather large for a blink program. I can send the source upon request. I will post source if I ever recompile a simpler version that does not depend on ASF.

SAM-BA (USB CDC) Bootloader

Both the SAM-BA USB CDC bootloader and the blink sketch were pre-installed by using the Arduino IDE. Neither the region lock bits nor the security bit is set. The fuses are left at default settings.

Without Bootloader

The Blink program can be installed using atprogram.exe (included with Atmel Studio) with the following commands (ATSAMD21E18A shown):

```
atprogram -t atmelice -i SWD -d atsamd21e18a -cl 500khz program -c --verify -f
c:\MT_D21E_Blink_256_no_offset_flash.hex
```

Troubleshooting / FAQ

- Linux users: disable modemmanager (Ubuntu), install udev rules (see driver installation)

Support Information

Please check the MattairTech website (http://www.MattairTech.com/) for firmware and software updates. Email me if you have any feature requests, suggestions, or if you have found a bug. If you need support, please contact me (email is best). You can also find support information at the MattairTech website. A support forum is planned. Support for Atmel ARM in general can be found at http://www.at91.com/.

Justin Mattair
MattairTech LLC
PO Box 1079
Heppner, OR 97836 USA
justin@mattair.net (preferred)
541-626-1531 (message only)
http://www.mattairtech.com/

Document Revision History

- October 3, 2017: Initial Release. Because the documentation is in flux at this time, not all updates will be shown.
- October 18, 2017: Modified 'Special Notes for D51'. SPI requires 3 jumpers wires (not 1).
- November 5, 2017: Added Custom variant, which can be customized and ordered online.

Legal

Copyright / Licenses

Arduino core files / SAM-BA Bootloader:

This core has been developed by Arduino LLC in collaboration with Atmel.
This fork developed by Justin Mattair of MattairTech LLC.

Petit FatFS:

Petit FatFs module is an open source software to implement FAT file system to
small embedded systems. This is a free software and is opened for education,
research and commercial developments under license policy of following trems.

MSD Bootloader:

```
* 1. Redistributions of source code must retain the above copyright notice,
*    this list of conditions and the following disclaimer.
*
* 2. Redistributions in binary form must reproduce the above copyright notice,
*    this list of conditions and the following disclaimer in the documentation
*    and/or other materials provided with the distribution.
*
* 3. The name of Atmel may not be used to endorse or promote products derived
*    from this software without specific prior written permission.
*
* 4. This software may only be redistributed and used in connection with an
*    Atmel microcontroller product.
*
* THIS SOFTWARE IS PROVIDED BY ATMEL "AS IS" AND ANY EXPRESS OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
* MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT ARE
* EXPRESSLY AND SPECIFICALLY DISCLAIMED. IN NO EVENT SHALL ATMEL BE LIABLE FOR
* ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
* DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
* OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
* HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
* STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN
* ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
* POSSIBILITY OF SUCH DAMAGE.
*
* \asf_license_stop
*/
```

Portions of this code are Copyright (C) 2009-2012 ARM Limited. All rights reserved.

```
* @note
* Copyright (C) 2009-2012 ARM Limited. All rights reserved.
*
* @par
* ARM Limited (ARM) is supplying this software for use with Cortex-M
* processor based microcontrollers.  This file can be freely distributed
* within development tools that are supporting such ARM based processors.
*
* @par
* THIS SOFTWARE IS PROVIDED "AS IS".  NO WARRANTIES, WHETHER EXPRESS, IMPLIED
* OR STATUTORY, INCLUDING, BUT NOT LIMITED TO, IMPLIED WARRANTIES OF
* MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE APPLY TO THIS SOFTWARE.
* ARM SHALL NOT, IN ANY CIRCUMSTANCES, BE LIABLE FOR SPECIAL, INCIDENTAL, OR
* CONSEQUENTIAL DAMAGES, FOR ANY REASON WHATSOEVER.
```

Portions of this code are copyright © 2003-2014, Dean Camera (www.fourwalledcubicle.com)
Specifically, the virtual FAT implementation from his MSD bootloader is used in the Xeno bootloader:

```
             LUFA Library
     Copyright (C) Dean Camera, 2014.

  dean [at] fourwalledcubicle [dot] com
           www.lufa-lib.org


  Copyright 2014  Dean Camera (dean [at] fourwalledcubicle [dot] com)

  Permission to use, copy, modify, distribute, and sell this
  software and its documentation for any purpose is hereby granted
  without fee, provided that the above copyright notice appear in
  all copies and that both that the copyright notice and this
  permission notice and warranty disclaimer appear in supporting
  documentation, and that the name of the author not be used in
```

advertising or publicity pertaining to distribution of the software without specific, written prior permission.

The author disclaims all warranties with regard to this software, including all implied warranties of merchantability and fitness. In no event shall the author be liable for any special, indirect or consequential damages or any damages whatsoever resulting from loss of use, data or profits, whether in an action of contract, negligence or other tortious action, arising out of or in connection with the use or performance of this software.

Software Warranty Disclaimer

The author disclaim all warranties with regard to this software, including all implied warranties of merchantability and fitness. In no event shall the author be liable for any special, indirect or consequential damages or any damages whatsoever resulting from loss of use, data or profits, whether in an action of contract, negligence or other tortious action, arising out of or in connection with the use or performance of this software.

Hardware Disclaimer

This development board/kit is intended for use for FURTHER ENGINEERING, DEVELOPMENT, DEMONSTRATION, OR EVALUATION PURPOSES ONLY. It is not a finished product, and may not (yet) comply with some or any technical or legal requirements that are applicable to finished products, including, without limitation, directives regarding electromagnetic compatibility, recycling (WEEE), FCC, CE, or UL (except as may be otherwise noted on the board/kit). MattairTech LLC supplied this board/kit AS IS, without any warranties, with all faults, at the buyer's and further users' sole risk. The user assumes all responsibility and liability for proper and safe handling of the goods. Further, the user indemnifies MattairTech LLC from all claims arising from the handling or use of the goods. Due to the open construction of the product, it is the user's responsibility to take any and all appropriate precautions with regard to electrostatic discharge and any other technical or legal concerns.

The product described in this document is subject to continuous development and improvements. All particulars of the product and its use contained in this document are given by MattairTech LLC in good faith. However all warranties implied or expressed including but not limited to implied warranties of merchantability or fitness for particular purpose are excluded.

This document is intended only to assist the reader in the use of the product. MattairTech LLC shall not be liable for any loss or damage arising from the use of any information in this document or any error or omission in such information or any incorrect use of the product.

Export Compliance

If the item(s) are exported from the United States, then the purchaser will be screened using the following sources (other sources not listed here may also be used):

Department of Commerce – Bureau of Industry and Security
Denied Persons List - Unverified List - Entity List

Department of State – Bureau of International Security and Non-proliferation
Nonproliferation Sanctions

Department of State – Directorate of Defense Trade Controls
AECA Debarred List

Department of the Treasury – Office of Foreign Assets Control
Specially Designated Nationals List
Foreign Sanctions Evaders List
Sectoral Sanctions Identifications (SSI) List
Palestinian Legislative Council (PLC) List
The List of Foreign Financial Institutions Subject to Part 561 (the Part 561 List)
Non-SDN Iranian Sanctions Act List (NS-ISA)

If encryption hardware is present (L21 or D51 installed), then the following statements apply:

MattairTech LLC submits proper documentation to Bureau of Industry and Security (BIS) for classification and review. After a waiting period, the item can then be shipped worldwide.

This product contains encryption technology and is classified as ECCN 5A002A.1.
Exported under license exception ENC.

These commodities, technology, or software were exported from the United States in accordance with the export administration regulations. Diversion contrary to U.S. law is prohibited.

In some cases, MattairTech LLC will need to obtain further information from the purchaser. If so, we will send you an email with further instructions.

RoHS, FCC, CE, UL, WEEE, etc.

This development board/kit is intended for use for FURTHER ENGINEERING, DEVELOPMENT, DEMONSTRATION, OR EVALUATION PURPOSES ONLY. It is not a finished product, and may not (yet) comply with some or any technical or legal requirements that are applicable to finished products, including, without limitation, directives regarding electromagnetic compatibility, recycling (WEEE), FCC, CE, or UL (except as may be otherwise noted on the board/kit). This development board is not RoHS compliant.

Trademarks

References

The microcontroller features / comparisons taken from respective Atmel datasheet:

D21: http://ww1.microchip.com/downloads/en/DeviceDoc/40001882A.pdf
L21: http://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-42385-SAM-L21-Datasheet.pdf
C21: http://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-42365-SAM-C21_Datasheet.pdf
D51: http://www.microchip.com/mymicrochip/filehandler.aspx?ddocname=en599585

Features list in Power Supply chapter from Microchip datasheet

Features list in Memory Device chapter from Microchip datasheet
Features list in Memory Device chapter from Adesto datasheet
Features list in Memory Device chapter from ON Semiconductor datasheet

Precautions

CAUTION

Do not operate the D21, L21, D51, SD card, optional sensors, or optional FLASH memory at >3.6V. This is only possible when using the VccL to VccH jumper (although it is possible to connect the buck converter directly to VccH and use an output voltage of 3.6V or less). **See the 'Power Supply' chapter** and 'IC Voltage Compatibility Table'.

CAUTION

Several power components on the Xeno require thermal derating, including the motor controller, buck converter, power multiplexer, Host / AUX switch, and both linear regulators. As with all linear regulators, higher regulator input voltages mean larger voltage drops and thus higher thermal dissipation for a given amount of current.

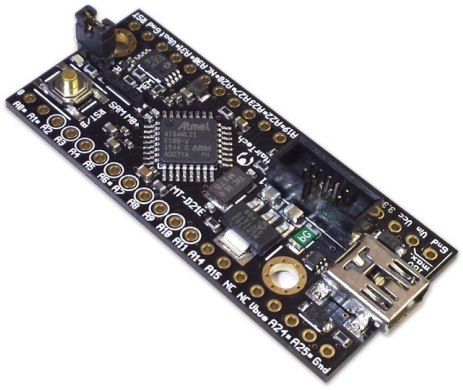
Be sure to limit current consumption to prevent excessive heat when using higher voltages and/or currents. **See 'Thermal Derating'** for information on current limits for a given ambient temperature.

All power supply ICs have thermal protection. They will power down if they get too hot, then power up after cooling down a small amount. Do not use this as a substitute for proper thermal control measures, as **excessive use of thermal protection degrades the device**.

CAUTION

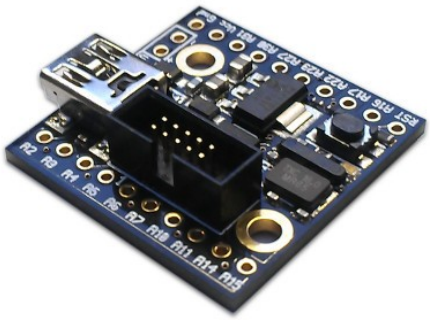
The Xeno contains static sensitive components.
Use the usual ESD procedures when handling.
All IC's have basic ESD protection (2KV HBM) while some have enhanced protection (4KV HBM). The reverse polarity protection P-FET has a zener diode and resistor at the gate.

Other MattairTech Products



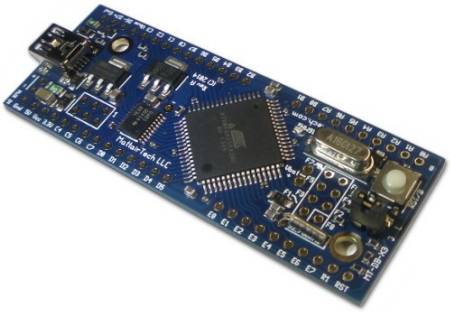
MT-D21E Rev B USB ARM Cortex M0+ board

- Choice of SAM D21 / L21 / C21 (32-pin)
- Up to 256KB FLASH, Up to 32KB SRAM
- L21 buck converter supported. 5V support for C21.
- Optional memory device (SRAM, EEPROM, or FLASH)
- Onboard 3.3V, 250mA LDO regulator (2uA quiescent)
- 16MHz and 32.768KHz crystals
- USB connector (power by USB or external up to 16V)
- Blue LED, 10-pin Cortex header, 2 buttons, I2C pullups
- USB CDC or MSD Bootloader (no programmer required)
- Arduino 1.6.6+ support (core and bootloader)



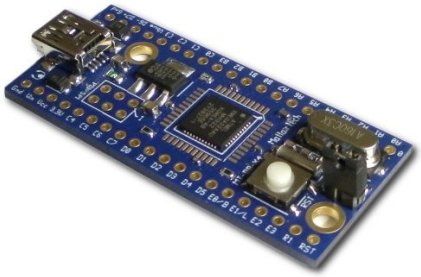
MT-D11 USB ARM Cortex M0+ board

- ATSAMD11D14AM (24-pin)
- 16KB FLASH, 4KB SRAM
- Onboard 3.3V, 250mA LDO regulator (2uA quiescent)
- 16MHz and 32.768KHz crystals
- USB connector (power by USB or external up to 16V)
- Blue LED, 10-pin Cortex header, button, I2C pullups
- USB CDC Bootloader (no programmer required)
- Arduino 1.6.6+ support (core and bootloader)



MT-DB-X3 USB AVR XMEGA board

- XMEGA A3U, A3BU, C3, and D3 (64-pin)
- 32KB - 384KB FLASH, 4KB – 32KB SRAM
- 3.3V 250mA regulator (2uA quiescent current)
- Optional 5V 500mA regulator (23uA quiescent current)
- Optional auto-direction sensing level shifter
- 16MHz and 32.768KHz crystals, optional coin cell holder
- LED, boot jumper, PDI header, button, TWI pullups
- USB DFU bootloader preinstalled (except D variant)



MT-DB-X4 USB AVR XMEGA board

- ATxmega128A4U USB XMEGA AVR
- 128KB FLASH, 8KB SRAM, 2KB EEPROM
- 3.3V LDO regulator (low quiescent current)
- 16MHz and 32.768KHz crystals
- LED, boot jumper, PDI header
- Reset button, mounting holes
- USB DFU bootloader preinstalled