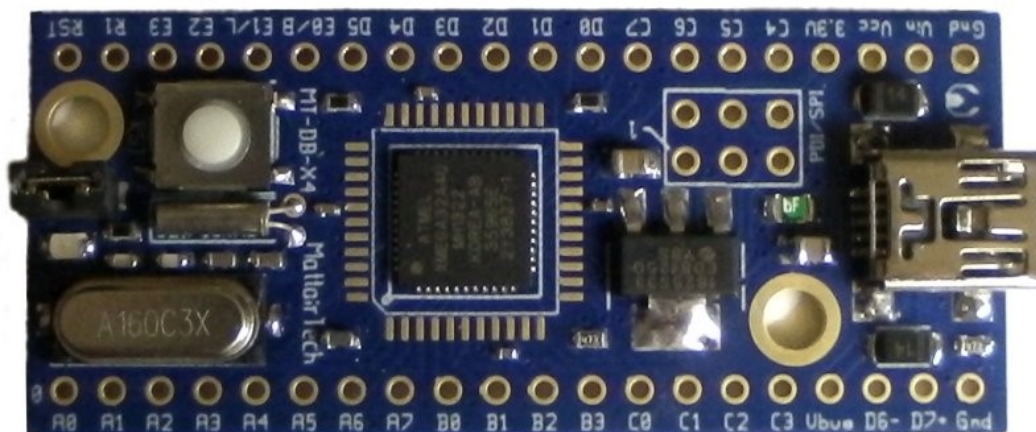


Table of Contents

Overview.....	3
Introduction.....	3
Board Features.....	4
ATxmegaXXXA4U Features.....	5
MT-DB-X4 Hardware.....	6
Top View / Pinout.....	6
Main Header Pins (Power).....	7
Main Header Pins (Signal).....	8
Inboard Header Pins.....	9
Solder Jumpers.....	10
USB DFU Bootloader (AU or C only).....	12
Schematic.....	14
Fuse and Lock Settings.....	15
Blink Demo.....	15
Troubleshooting / FAQ.....	15
Support Information.....	16
Legal.....	17
Appendix A: Precautions.....	18
Appendix B: Other MattairTech Products.....	19

Overview



Introduction

The MT-DB-X4 is a development board for the 44-pin Atmel AVR XMEGA AU, A, C, and D microcontrollers. The AU (ie: ATxmega128a4u) and C variants support USB device mode. All variants can be powered from USB or from the Vin pin. Two schottky diodes facilitate simple switching (and reverse-polarity protection) between the two power sources. This voltage is regulated to 3.3V by the onboard 500mA, low quiescent current LDO regulator that supports up to 6V DC input voltage. Overcurrent protection is provided by a 500mA PTC resettable fuse. Also mounted is a mini USB connector, green LED, 16MHz crystal, 32.768KHz RTC crystal, reset button, and a boot jumper. The board has 40 main dual inline header pins with 100 mil pin spacing and 700 mil row spacing which allows for mounting on a breadboard or perfboard. The PDI/SPI header can be used with an external programmer, or be reconfigured to be used as a SPI master or slave. A DFU bootloader comes preinstalled (except A and D variants) which allows programming of the chip over USB without an external programmer. The PCB measures approximately 2.1" x 0.9" x 0.062" (52mm x 23mm x 1.6mm).

Board Features

- **Atmel AVR XMEGA AU, A, C, or D 44-pin microcontroller**
 - ATxmega32A4U, ATxmega128A4U, ATxmega16C4, ATxmega32D4, ATxmega128D4
 - AU (A4U) and C variants support USB device mode
- **Onboard 3.3V, 500mA LDO regulator**
 - up to 6V DC input
 - low quiescent current (120uA typical, 220uA max.)
 - low dropout (210mV typical, 350mV max. @ 500mA)
 - 0.5% output tolerance typical
 - Over-current and over-temperature protection
- **Simple power source switching**
 - 2 schottky diodes (Vbus and Vin)
 - Low voltage drop (250mV @ 50mA, 375mV @ 500mA)
 - Reverse-polarity protection
 - Vbus can be disconnected (solder jumper)
 - Vin schottky can be shorted (solder jumper) to eliminate voltage drop (battery use)
 - Either Vin or Vbus can be routed through a resistor divider (/2) to pin B3 (ADC)
- **PTC resettable fuse (500mA)**
- **DFU bootloader preinstalled (FLIP or dfu-programmer) (NOT on A or D variants)**
- **PDI/SPI header (PDI by default)**
 - As a PDI header, it is used to program the AVR with an external programmer
 - Header can be converted to a SPI (master or slave) header
 - Can mount the MT-SD MicroSD card slot directly to this header
- **16MHz crystal for main clock**
- **32.768KHz crystal for real-time counter**
- **Green Status LED (can be disconnected)**
- Reset button
- Bootloader activation jumper
- Mini USB connector
- USB pins routed to header pins (for panel-mount USB connector)
- Powered by USB or external power source (up to 6V) on Vin
- Ferrite bead on analog supply
- 19 solder jumpers for configuration flexibility
- All PORT pins routed to headers (except R0)
- 2 main headers are on 0.1" spacing (breadboard/perfboard mounting)
- **Two 3mm mounting holes (~5mm pad)**
- High-quality PCB with gold-plated finish
- Measures approx. 2.1" x 0.9" (52mm x 23mm) and 0.062" (1.6mm) thick.

ATxmegaXXXA4U Features

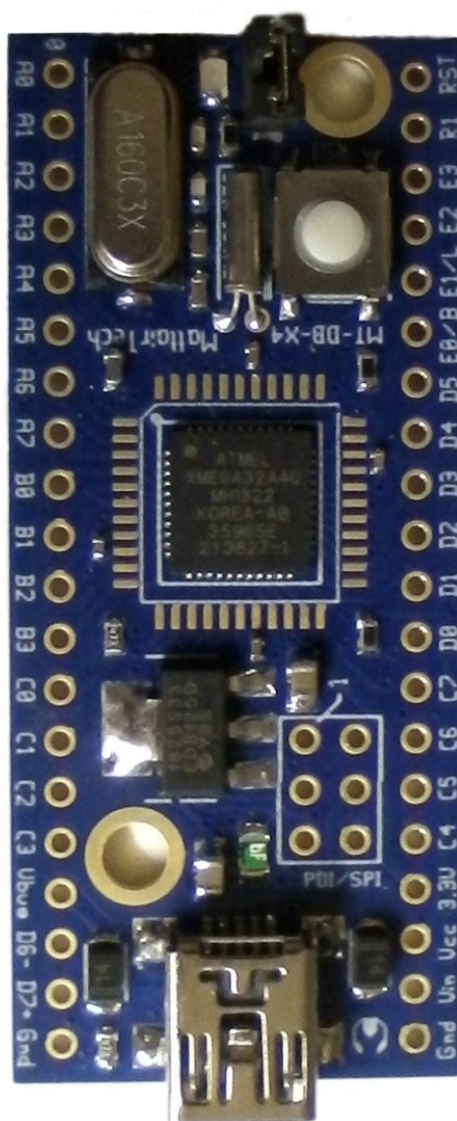
(See Atmel website for A, D, or C variant specs)

- **High-performance, low-power Atmel® AVR® XMEGA® 8/16-bit Microcontroller**
- **Nonvolatile program and data memories**
 - 16K - 128KB of in-system self-programmable flash
 - 4K - 8KB boot section
 - 1K - 2KB EEPROM
 - 2K - 8KB internal SRAM
- **Peripheral Features**
 - Four-channel DMA controller
 - Eight-channel event system
 - Five 16-bit timer/counters
 - Three timer/counters with 4 output compare or input capture channels
 - Two timer/counters with 2 output compare or input capture channels
 - High-resolution extensions on all timer/counters
 - Advanced waveform extension (AWeX) on one timer/counter
 - One USB device interface
 - USB 2.0 full speed (12Mbps) and low speed (1.5Mbps) device compliant
 - 32 Endpoints with full configuration flexibility
 - Five USARTs with IrDA support for one USART
 - Two Two-wire interfaces with dual address match (I2C and SMBus compatible)
 - Two serial peripheral interfaces (SPIs)
 - AES and DES crypto engine
 - CRC-16 (CRC-CCITT) and CRC-32 (IEEE® 802.3) generator
 - 16-bit real time counter (RTC) with separate oscillator
 - One twelve-channel, 12-bit, 2msps Analog to Digital Converter
 - One two-channel, 12-bit, 1msps Digital to Analog Converter
 - Two Analog Comparators with window compare function, and current sources
 - External interrupts on all general purpose I/O pins
 - Programmable watchdog timer with separate on-chip ultra low power oscillator
 - QTouch® library support
 - Capacitive touch buttons, sliders and wheels
- **Special microcontroller features**
 - Power-on reset and programmable brown-out detection
 - Internal and external clock options with PLL and prescaler
 - Programmable multilevel interrupt controller
 - Five sleep modes
 - Programming and debug interfaces
 - PDI (program and debug interface)
- **I/O and packages**
 - 34 Programmable I/O pins
 - 44 - lead TQFP
 - 44 - pad VQFN/QFN
 - 49 - ball VFBGA
- **Operating voltage**
 - 1.6 - 3.6V
- **Operating frequency**
 - 0 - 12MHz from 1.6V
 - 0 - 32MHz from 2.7V

MT-DB-X4 Hardware

Top View / Pinout

A0, ADC0, AC0, AREF
 A1, ADC1, AC1
 A2, ADC2, AC2
 A3, ADC3, AC3
 A4, ADC4, AC4
 A5, ADC5, AC5
 A6, ADC6, AC6, AC10OUT
 A7, ADC7, AC7, AC00OUT
 B0, ADC8, AREF
 B1, ADC9
 B2, ADC10, DAC0
 B3, ADC11, DAC1
 C0, OC0A, SDA
 C1, OC0B, XCK0, SCL
 C2, OC0C, RXD0
 C3, OC0D, TXD0
 Vbus
 D6-, USB D-, RXD1, MISO
 D7+, USB D+, TXD1, SCK
 Gnd



RST
 R1, XTAL1, TOSC1, Ext. Clock
 E3, OC0D, TXD0, TOSC1 Alt.
 E2, OC0C, RXD0, TOSC2 Alt.
 E1/L, LED, OC0B, XCK0, SCL
 E0/B, Boot, OC0A, SDA
 D5, OC1B, XCK1, MOSI
 D4, OC1A, SS
 D3, OC0D, TXD0
 D2, OC0C, RXD0
 D1, OC0B, XCK0
 D0, OC0A
 C7, TXD1, SCK
 C6, RXD1, MISO
 C5, OC1B, XCK1, MOSI
 C4, OC1A, SS
 3.3V
 Vcc
 Vin
 Gnd

Main Header Pins (Power)

Pin	Description
Gnd (2)	Ground
Vbus	Vbus is connected directly to the Vbus pin (5V) of the USB connector and to a 10uF capacitor. It is routed through a schottky diode and through J6 to the regulator input circuitry. Disconnect J6 if only using external power on the Vin pin but still need USB communications. Vbus voltage can be measured on pin B3 by connecting J13 and setting J3 toward the Vbus side of the board. J13 will connect B3 to a resistor divider consisting of two 47Kohm resistors (divide by 2), and J3 connects to Vbus. The resistor divider will pull B3 to near ground level when Vbus is disconnected. Because of a small leakage current from the schottky diode, a small voltage should be interpreted as USB disconnected. The leakage is small enough such that the pin can still be used as a digital input.
Vin	Vin is the external power input pin. Up to 6V can be connected. It is routed through a schottky diode to the regulator input circuitry. The schottky diode can be shorted with J1, eliminating the voltage drop across the diode, which can be useful for battery applications. Note that when the diode is shorted, reverse-polarity protection is disabled, and J6 should be disconnected to prevent Vbus current from flowing into Vin. Vin voltage can be measured on pin B3 by connecting J13 and setting J3 toward the Vin side of the board. J13 will connect B3 to a resistor divider consisting of two 47Kohm resistors (divide by 2), and J3 connects to Vin. The resistor divider will pull B3 to near ground level when Vin is disconnected. Because of a small leakage current from the schottky diode, a small voltage should be interpreted as Vin disconnected. The leakage is small enough such that the pin can still be used as a digital input.
Vcc	This pin is connected to the Vcc and AVcc (through a ferrite bead) pins on the XMEGA, the PDI/SPI header Vcc pin, and the reset pullup. Vcc is connected to 3.3V through J5, which in turn is connected to the output of the onboard regulator. The Vcc pin can also be used as an input. Disconnect J5 to supply power from an external source to the Vcc pin.
3.3V	3.3V is connected to the output of the onboard 3.3V regulator. There is a 10uF capacitor on this line. 3.3V is normally connected to Vcc through J5.

Main Header Pins (Signal)

Pin	Description
A0 - A7	Port A. These can be used for analog functions or GPIO.
B0 - B3	Port B. These can be used for analog functions or GPIO. B3 can be connected to an onboard voltage divider, which in turn can be connected to either Vbus or Vin. See Vbus or Vin pin description for more information.
C0 - C7	Port C. These can be used for digital functions. Pins C4 – C7 can be connected to the PDI/SPI header by using jumpers J8, J9, J11, and J12. To convert the header to SPI mode, connect J9 and J11, then change J8 and J12 to the alternate positions.
D0 - D5	Port D. These can be used for digital functions.
D6-, D7+	These pins are connected to pins D6 and D7 of the XMEGA. They are also routed to the D- and D+ pins of the USB connector through J4 and J7 (J4 and J7 are connected on boards with the XMEGA AU or C variants). These pins, along with the adjacent Vbus and Ground pins can be used for a panel-mount USB connector. Or, D6- and D7+ can be used for digital functions with J4 and J7 disconnected. NO USB for A or D variants.
E0 / B	This pin is connected to the boot jumper (though a 249 ohm resistor) and to pin E0 of the XMEGA. The jumper in turn connects to ground. If the jumper is installed, the bootloader is run, otherwise, the user application is run. This pin can function as a normal GPIO pin at all other times. The 249 ohm resistor provides short-circuit protection in case the pin is used as an output and the jumper is installed.
E1 / L	This pin is connected to the green LED and to pin E1 of the XMEGA. The LED is connected to E1 through a 249 ohm resistor and jumper J14. Drive pin E1 high to turn on the LED. Disconnect J14 to disable the LED.
E2, E3	Port E. These can be used for digital functions. On the AU variant, E2 and E3 can also be used as the TOSC pins. To use for digital, set J15 and J16 to the positions toward the Vin side of the board. To connect to the 32.768KHz crystal, set J15 and J16 toward the Vbus side of the board. Because R0 and R1 can also connect to this crystal, be sure that J18 and J19 are set toward the Vbus side of the board or are disconnected.
R1	This pin can be used as an external clock input or for GPIO when connected to pin R1 of the XMEGA. To use, disconnect J18 from both positions so that R1 does not connect to either the 16MHz or 32.768KHz crystals. Then connect J17.
RST	RST connects to the reset pin of the XMEGA, the reset button, and through J8 to the PDI/SPI header. A 10K pullup resistor is connected to this pin. J8 is used along with other jumpers to set the header to PDI or SPI mode. Note that the XMEGA reset pin is also used as the PDI clock.

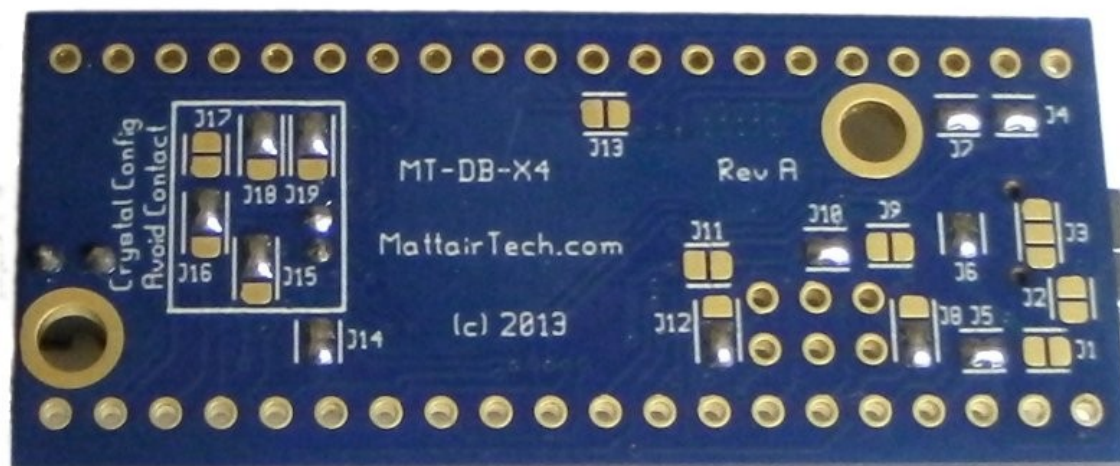
Inboard Header Pins

<i>Pin</i>	<i>Description</i>
PDI/SPI Header	This header can be connected to an external PDI programmer/debugger. Alternatively, this header can be used for SPI communications (master or slave). To convert the header to SPI mode, connect J9 and J11, then change J8 and J12 to the alternate positions. This will route 4 header pins to pins C4 – C7 of the XMEGA (there is a SPI peripheral here). Pin C4 can be used as a chip select (or as a slave select if in slave mode). See schematic for pinout information.
Jumper	This jumper is used to select between running the bootloader or running the application. The jumper shorts XMEGA pin E0 to ground through a 249 ohm resistor. By default, a bootloader is pre-installed, and the fuses are set to jump to the bootloader section of the FLASH after reset. The bootloader will read the state of the pin (after turning on the pullup and waiting for it to stabilize) and either jump to the application section of the FLASH or continue with the bootloader. If the jumper is installed, the bootloader is run, otherwise, the user application is run. The E0 pin can function as a normal GPIO pin at all other times. The 249 ohm resistor provides short-circuit protection in case the pin is used as an output and the jumper is installed. Note that there is no on-chip function that reads a pin for the purpose of bootloader activation (ie: HWB pin of other USB-enabled AVRs).

Solder Jumpers

<i>Pin</i>	<i>Description</i>
J1: Vin diode disable	Vin is the external power input pin. Up to 6V can be connected. It is routed through a schottky diode to the regulator input circuitry. The schottky diode can be shorted by closing J1, eliminating the voltage drop across the diode, which can be useful for battery applications. Note that when the diode is shorted, reverse-polarity protection is disabled, and J6 should be opened to prevent Vbus current from flowing into Vin.
J2: USB Shield Ground	Jumper J2 can be closed to connect the USB shield to ground. The USB specification calls for the USB shield to be grounded on the host side only. However, some prefer to have it grounded, although this could cause the shield to act as an antenna or a ground loop would be formed. An 0603 component (or maybe two 0402) may be soldered on the pads.
J3: Voltage divider input	Vin or Vbus voltage can be measured on pin B3 by closing J13 and setting J3 toward either the Vin (for Vin measurement) or the Vbus (for Vbus measurement) side of the board (refer to printing on top side of pcb). J13 will connect B3 to a resistor divider consisting of two 47Kohm resistors (divide by 2), and J3 connects to Vin or Vbus. The resistor divider will pull B3 to near ground level when Vin or Vbus is disconnected. Because of a small leakage current from the schottky diode, a small voltage should be interpreted as Vin or Vbus disconnected. The leakage is small enough such that the pin can still be used as a digital input.
J4: USB D+ / Pin D7	The D6- and D7+ pins are connected to pins D6 and D7 of the XMEGA. By default, they are also routed to the D- and D+ pins of the USB connector through J7 and J4 (J7 and J4 are not closed on boards with the XMEGA D variant). The D6- and D7+ pins, along with the adjacent Vbus and Ground pins can be used for a panel-mount USB connector. D6- and D7+ can be used for digital functions with J7 and J4 open.
J5: Vcc – 3.3V	This connects the 3.3V regulator output rail to Vcc. Disconnect if supplying a regulated voltage (3.6V or less) externally on the Vcc pin.
J6: Vbus Power	This routes Vbus to the regulator input circuitry. There are two schottky diodes, one for Vin and one for Vbus. They facilitate automatic power switching between these two sources. If only external power will be used (Vin), disconnect J6. This will prevent Vbus power from being used when a USB cable is plugged in for communications.
J7: USB D- / Pin D6	See J4.
J8: PDI/SPI Selection	The PDI/SPI header can be connected to an external PDI programmer/debugger. Alternatively, this header can be used for SPI communications (master or slave). The image below shows the PDI configuration, which is the default. To convert the header to SPI mode, close J9 and J11, then change J8 and J12 to the alternate positions. This will route 4 header pins to pins C4 – C7 of the XMEGA (there is a SPI peripheral here). Pin C4 can be used as a chip select (or as a slave select if in slave mode). See schematic for pinout information.
J9: PDI/SPI Selection	See J8. The image below shows the PDI configuration.
J10: Regulator Input	May be used for planned MattairTech IO board. See schematic.

J11: PDI/SPI Selection	See J8. The image below shows the PDI configuration.
J12: PDI/SPI Selection	See J8. The image below shows the PDI configuration.
J13: Pin B3 to voltage divider	See J3.
J14: LED	Pcb pin E1/L is connected to the green LED (through a 249 ohm resistor and jumper J14) and to pin E1 of the XMEGA. Drive pin E1 high to turn on the LED. Open J14 to disable the LED.
J15: E2/E3 function	XMEGA pins E2 and E3 can be used for digital functions. On the AU variant, E2 and E3 can also be used as the TOSC pins. To connect to the 32.768KHz crystal, set J15 and J16 as shown in the image below. In this case, because R0 and R1 can also connect to this crystal, be sure that J18 and J19 are set as shown in the image or have all three pads desoldered. To connect E2 and E3 to the header, set J15 and J16 to the alternate position.
J16: E2/E3 function	See J15. The image below shows the 32.768KHz crystal position.
J17: R1 header pin enable	Header pin R1 can be used as an external clock input or for GPIO when connected to pin R1 of the XMEGA. To use, remove solder from all three pads of J18 so that R1 does not connect to either the 16MHz or 32.768KHz crystals. Then close J17.
J18: R0/R1 function	R0 and R1 can be connected to either the 16MHz crystal (XTAL) or the 32.768KHz crystal (TOSC). Additionally, on XMEGA AU variants, pins E2 and E3 can connect to the 32.768KHz crystal. To use the 16MHz crystal (default), set J18 and J19 as shown in the image below. To use the 32.768KHz crystal, set toward the alternate position. In this case, because E2 and E3 can also connect to this crystal, be sure that J15 and J16 are set to the opposite as shown below so that E2 and E3 connect to the header instead of to the 32.768KHz crystal. Remove solder from all three pads of J18 if using header pin R1 (see J17). See also J15.
J19: R0/R1 function	See J18. The image below shows the 16MHz crystal position.



USB DFU Bootloader (AU or C only)

USB enabled XMEGAs (A4U or C4) come with a USB DFU bootloader pre-installed. The XMEGA D4 does not have onboard USB hardware, so no bootloader is installed. The bootloader is from Atmel. Documentation can be found in [AVR1916](#). Note that the MT-DB-X4 uses a different bootloader activation pin than the Atmel default. The hex files provided by Atmel (from [AVR1916.zip](#)) were patched directly to use the MT-DB-X4 boot jumper pin. The patched hex files can be found at http://www.mattairtech.com/software/MT-DB-X4/MT-DB-X4_DFU_Bootloaders_104.zip.

Installing FLIP / USB DFU Drivers

FLIP is a graphical utility for Windows used to load firmware into the XMEGA. FLIP supplies the USB DFU bootloader driver. Download FLIP 3.4.7 or higher from <http://www.atmel.com/tools/flip.aspx> and install. Older versions may not support the latest XMEGA variants. Once FLIP is installed, the USB DFU drivers can be loaded. Install the boot jumper and power the board (or press reset). This will start the DFU bootloader. Windows will then prompt you for the driver, which is located in the Program Files/Atmel/Flip 3.4.7/usb directory. Point the installer to that directory and install. Once the driver is loaded, the device will appear under Atmel USB Devices in the device manager. No driver is needed for Linux or OS X.

Using FLIP

Install the boot jumper and power the board (or press reset). This will start the DFU bootloader. Now launch the FLIP utility. When it has loaded, click on the chip icon and select your XMEGA variant. Next, click on the USB icon, select USB, then connect. The screen should now show information about your XMEGA variant. Click on the File menu, and open the appropriate hex file. More information will appear about the program. Be sure that erase is checked. The firmware cannot be loaded unless the flash is erased first. Program must be checked. Verify should also be checked. Now click on the Run button in the lower-left of the screen, and the firmware will be quickly loaded into the XMEGA FLASH.

You may also program the EEPROM. If so, click on Select EEPROM at the bottom. Then, click on the File menu and open the appropriate eep file. You will have to change the file filter to allow you to see the eep file. Note that eep files are just hex files but with the eep extension instead of hex. More information will appear about the file when selected. Both Program and Verify should be checked. Click run to program the EEPROM. You can run your application without removing the jumper or pressing reset by unchecking the reset box and pressing the "Start Application" button (lower right).

Using dfu-programmer

dfu-programmer is a Linux command line utility used to program the XMEGA memories. Driver installation is not required. Download version 0.6.2 or higher from <http://dfu->

programmer.sourceforge.net/ . The following commands can be used (atxmega128a4u shown):

```
dfu-programmer atxmega128a4u erase
dfu-programmer atxmega128a4u flash Blink_128a4u.hex
dfu-programmer atxmega128a4u flash-EEPROM YourEep.eep (if applicable)
dfu-programmer atxmega128a4u start (to jump to application section without reset)
```

- dfu-programmer (through version 0.6.2) for Linux fails when the lockbits are set. Normally, these bits are set on chips that Atmel supplies with a DFU bootloader (ie: AT90USB, USB Mega, though I'm not sure about AVR32). Atmel has created a DFU bootloader for the XMEGAs, however, they do not yet pre-install it on the ATxmegaXXXA4U (and possibly others) as of chip datecode 1334. Thus, I install the bootloader, change the fuses, and set the lockbits. When programming with the Atmel DFU bootloader, the user must perform a chip erase first. This chip erase clears a security flag, allowing full read and write access to the FLASH. However, with the XMEGA DFU bootloaders, the flag is set again between command invocations (the flag remains clear with Windows version). Until this problem is resolved, please download a patch from http://www.mattairtech.com/software/dfu-programmer-0.6.2_xmega_linux_patch.patch, which will optionally perform a chip erase just before programming the FLASH (within the same command invocation). Move the patch to your dfu-programmer-0.6.2 directory (where you run configure). From this directory, apply the patch:

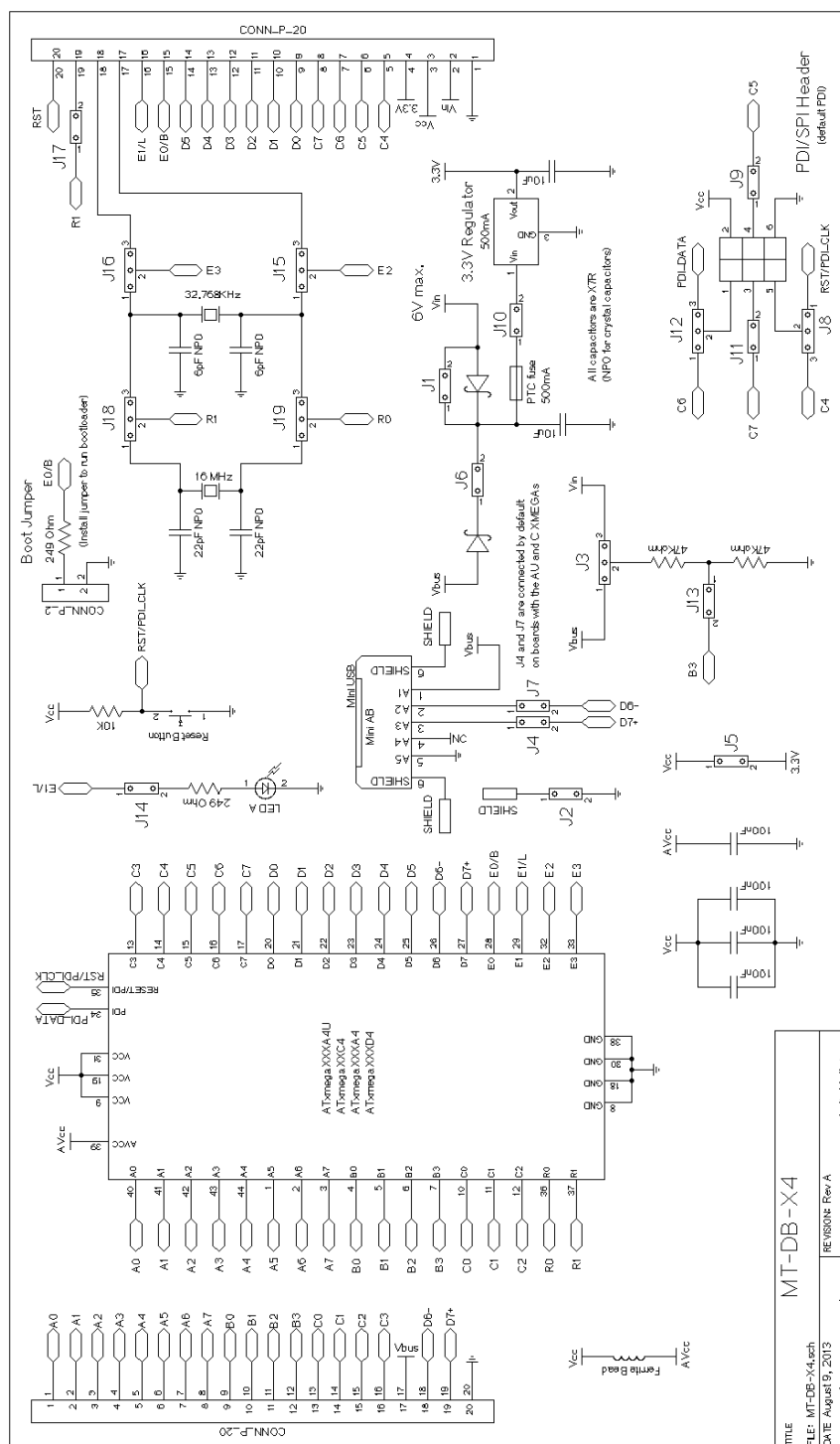
```
patch -p1 < dfu-programmer-0.6.2_xmega_linux_patch.patch
```

Recompile and run with `--erase-first`:

```
cygnus src # dfu-programmer atxmega128a4u flash --erase-first Blink.hex
```

This patch also addresses an issue with programming small-memory devices (ie: the ATxmega32a4u) where programming would fail, as well as an issue regarding total FLASH size (the bootloader section is in addition to the application section, ie: 128K + 8K).

Schematic



Fuse and Lock Settings

The bootloaders were pre-installed with the following commands (ATxmega128a4u shown):

```
avrdude -p x128a4u -c avrisp2 -P usb -e  
avrdude -p x128a4u -c avrisp2 -P usb -U fuse1:w:0x00:m -U fuse2:w:0x9F:m -U fuse4:w:0xFF:m -U fuse5:w:0xFF:m  
avrdude -p x128a4u -c avrisp2 -P usb -U flash:w:"MT-DB-X4-128a4u_104.hex"  
avrdude -p x128a4u -c avrisp2 -P usb -U lock:w:0x3c:m
```

The Blink program was pre-installed with the following commands (ATxmega128a4u shown):

```
dfu-programmer atxmega128a4u flash --erase-first Blink_128a4u.hex  
dfu-programmer atxmega128a4u reset
```

Note that the `--erase-first` argument requires the patch described in the troubleshooting section below.

UPDATE! As of September 7, 2014, the lockbits are no longer set due to incompatibilities with dfu-programmer.

The Blink program will then be pre-installed with the following commands (ATxmega128a4u shown):

```
dfu-programmer atxmega128a4u reset  
dfu-programmer atxmega128a4u flash Blink_128a4u.hex  
dfu-programmer atxmega128a4u reset
```

Blink Demo

A demo program comes pre-installed. It simply blinks the LED at 1Hz using the 16MHz crystal as the clock source. The hex files and source code can be found at http://www.mattairtech.com/software/MT-DB-X4/MT-DB-X4_Blink.zip.

Troubleshooting / FAQ

- dfu-programmer (through version 0.6.2) for Linux fails when the lockbits are set. Normally, these bits are set on chips that Atmel supplies with a DFU bootloader (ie: AT90USB, USB Mega, though I'm not sure about AVR32). Atmel has created a DFU bootloader for the XMEGAs, however, they do not yet pre-install it on the ATxmegaXXA4U (and possibly others) as of chip datecode 1334. Thus, I install the bootloader, change the fuses, and set the lockbits. When programming with the Atmel DFU bootloader, the user must perform a chip erase first. This chip erase clears a security flag, allowing full read and write access to the FLASH. However, with the XMEGA DFU bootloaders, the flag is set again between command invocations (the flag remains clear with Windows version). Until this problem is resolved, please download a patch from http://www.mattairtech.com/software/dfu-programmer-0.6.2_xmega_linux_patch.patch, which will optionally perform a chip erase just before programming the FLASH (within the same command invocation). Move the patch to your dfu-programmer-0.6.2 directory (where you run configure). From this directory, apply the patch:

```
patch -p1 < dfu-programmer-0.6.2_xmega_linux_patch.patch
```

Recompile and run with `--erase-first`:

```
cygnus src # dfu-programmer atxmega128a4u flash --erase-first Blink.hex
```

This patch also addresses an issue with programming small-memory devices (ie: the ATxmega32a4u) where programming would fail, as well as an issue regarding total FLASH size (the bootloader section is in addition to the application section, ie: 128K + 8K).

UPDATE! As of September 7, 2014, the lockbits are no longer set due to this incompatibility with dfu-programmer (so you can use the normal dfu-programming commands).

- **Are there any alternatives to Atmel Studio?**

The Atmel Standalone Toolchain (based on GCC) is available for Windows (<http://www.atmel.com/tools/atmelavrtoolchainforwindows.aspx>) and Linux (<http://www.atmel.com/tools/atmelavrtoolchainforlinux.aspx>). Atmel is the primary contributor to AVR support in GCC. As such, their own toolchain has the latest support/fixes. I would recommend using only this toolchain. It is more up to date than either the latest stock GCC or the distro builds. WinAVR is very old and does not support XMEGAs at all.

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 AVRs in general can be found at AVRfreaks (<http://www.avrfreaks.net/>). There, I monitor the forums section as the user physicist.

Justin Mattair
MattairTech LLC
PO Box 1079
Heppner, OR 97836 USA
541-626-1531
justin@mattair.net
<http://www.mattairtech.com/>

Acknowledgments

Thanks to Dean Camera (<http://www.fourwalledcubicle.com/>) for his excellent LUFA library, CDC bootloader, DFU bootloader, and AVRISP mkII clone programmer. Thanks to the members of AVRfreaks (<http://www.avrfreaks.net/>) for their support. Finally, thanks to Atmel for creating a great product, the AVR microcontroller.

Legal

Copyright Notices

Portions of this code are copyright (c) 2009-2013 Justin Mattair (www.mattairtech.com)

Portions of this code are copyright © 2003-2012, Atmel Corporation (<http://www.atmel.com/>)

ATxmegaXXXA4U Features (page 5) taken from Atmel datasheet.

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.

Trademarks

AVR® is a registered trademark of Atmel Corporation.

XMEGA is a trademark of Atmel Corporation.

All other trademarks are the property of their respective owners.

Appendix A: Precautions

CAUTION

Do not change power configuration, or solder any jumper while unit is powered. Do not short Vin, Vbus, 3.3V, or ground to each other (ie: solder jumpers on bottom shorting on clipped lead).

CAUTION

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. The regulator will enter thermal shutdown if it gets too hot.



CAUTION

Normally, power is supplied from Vin or Vbus. However, it is possible to disconnect the regulator and supply an externally regulated voltage on the 3.3V and/or Vcc pins. When doing this, care must be taken to limit inrush current on these pins due to the low ESR of the ceramic capacitors. Failure to do so may cause damaging inductive voltage spikes due to any wire inductance (ie: benchtop power supply leads). Inrush current is normally controlled by the PTC fuse, which has a small series resistance.

CAUTION

The MT-DB-X4 contains static sensitive components. Use the usual ESD procedures when handling.

Appendix B: Other MattairTech Products

 <p>AVR Programmer and Multitool</p> <p>ZeptoProg II</p>	<p>ZeptoProg II AVRISP mkII Programmer</p> <ul style="list-style-type: none"> • AVRISPMkII compatible AVR Programmer • Supports all AVR's with ISP, PDI, or TPI • Optional 5V output via headers to target board, with standard jumper and PTC fuse • 4-channel Logic Analyzer • Serial bridge / pattern generator / SPI interface • GPIO / PWM / frequency input & output • Atmel Studio / AVRDUDE support • Target board voltage of 2V to 5.5V via level-shifted pins on two main headers
	<p>MT-DB-U6 USB AVR development board</p> <ul style="list-style-type: none"> • AT90USB646 / AT90USB1286 USB AVR • 64KB/128KB FLASH, 4KB/8KB SRAM • 5V, 500mA LDO regulator (3V-30V input) • Auto power source selection IC (USB/External) • 16MHz and 32.768KHz crystals • Arduino compatible • CDC or DFU bootloader
	<p>MT-DB-X3 USB AVR XMEGA board</p> <ul style="list-style-type: none"> • 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)
	<p>MT-D21E USB ARM Cortex M0+ board</p> <ul style="list-style-type: none"> • ATSAMD21E17A or ATSAMD21E18A (32-pin) • 128KB/256KB FLASH, 16KB/32KB SRAM • Onboard 3.3V, 250mA LDO regulator (2uA quiescent) • 16MHz and 32.768KHz crystals • USB connector (power by USB or external up to 15V) • Blue LED, 10-pin Cortex header, 2 buttons, I2C pullups • USB Mass Storage Bootloader (no programmer required)