Intel® Edison Kit for Arduino*

Hardware Guide

September 2014

Revision 001

Intel Confidential

Notice: This document contains information on products in the design phase of development. The information here is subject to change without notice. Do not finalize a design with this information.

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

A "Mission Critical Application" is any application in which failure of the Intel Product could result, directly or indirectly, in personal injury or death. SHOULD YOU PURCHASE OR USE INTEL'S PRODUCTS FOR ANY SUCH MISSION CRITICAL APPLICATION, YOU SHALL INDEMNIFY AND HOLD INTEL AND ITS SUBSIDIARIES, SUBCONTRACTORS AND AFFILIATES, AND THE DIRECTORS, OFFICERS, AND EMPLOYEES OF EACH, HARMLESS AGAINST ALL CLAIMS COSTS, DAMAGES, AND EXPENSES AND REASONABLE ATTORNEYS' FEES ARISING OUT OF, DIRECTLY OR INDIRECTLY, ANY CLAIM OF PRODUCT LIABILITY, PERSONAL INJURY, OR DEATH ARISING IN ANY WAY OUT OF SUCH MISSION CRITICAL APPLICATION, WHETHER OR NOT INTEL OR ITS SUBCONTRACTOR WAS NEGLIGENT IN THE DESIGN, MANUFACTURE, OR WARNING OF THE INTEL PRODUCT OR ANY OF ITS PARTS.

Intel may make changes to specifications and product descriptions at any time, without notice. Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The information here is subject to change without notice. Do not finalize a design with this information.

Intel software products are copyrighted by and shall remain the property of Intel Corporation. Use, duplication, or disclosure is subject to restrictions stated in Intel's Software License Agreement, or in the case of software delivered to the government, in accordance with the software license agreement as defined in FAR 52.227-7013.

The products described in this document may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

The code names presented in this document are only for use by Intel to identify products, technologies, or services in development that have not been made commercially available to the public, i.e., announced, launched, or shipped. They are not "commercial" names for products or services and are not intended to function as trademarks.

Contact your local Intel sales office or your distributor to obtain the latest specifications and before placing your product order.

Copies of documents which have an order number and are referenced in this document, or other Intel literature may be obtained by calling 1-800-548-4725 or by visiting Intel's website at *http://www.intel.com/design/literature.htm*.

Intel processor numbers are not a measure of performance. Processor numbers differentiate features within each processor family, not across different processor families. See *http://www.intel.com/products/processor_number* for details.

Intel, the Intel logo, and Intel Atom are trademarks of Intel Corporation in the United States and other countries.

* Other brands and names may be claimed as the property of others.

Contents

Figures

Tables

Revision History

Revision	Description	Date
ww32	Initial release	August 4, 2014
ww34	Minor edits.	August 20, 2014
ww36	Removed a column from Table 2.	September 5, 2014
001	First public release.	September 9, 2014

§

1 Introduction

This document describes the hardware interface of the Intel® Edison kit for Arduino*.

1.1 References

Table 1 Product-specific documents

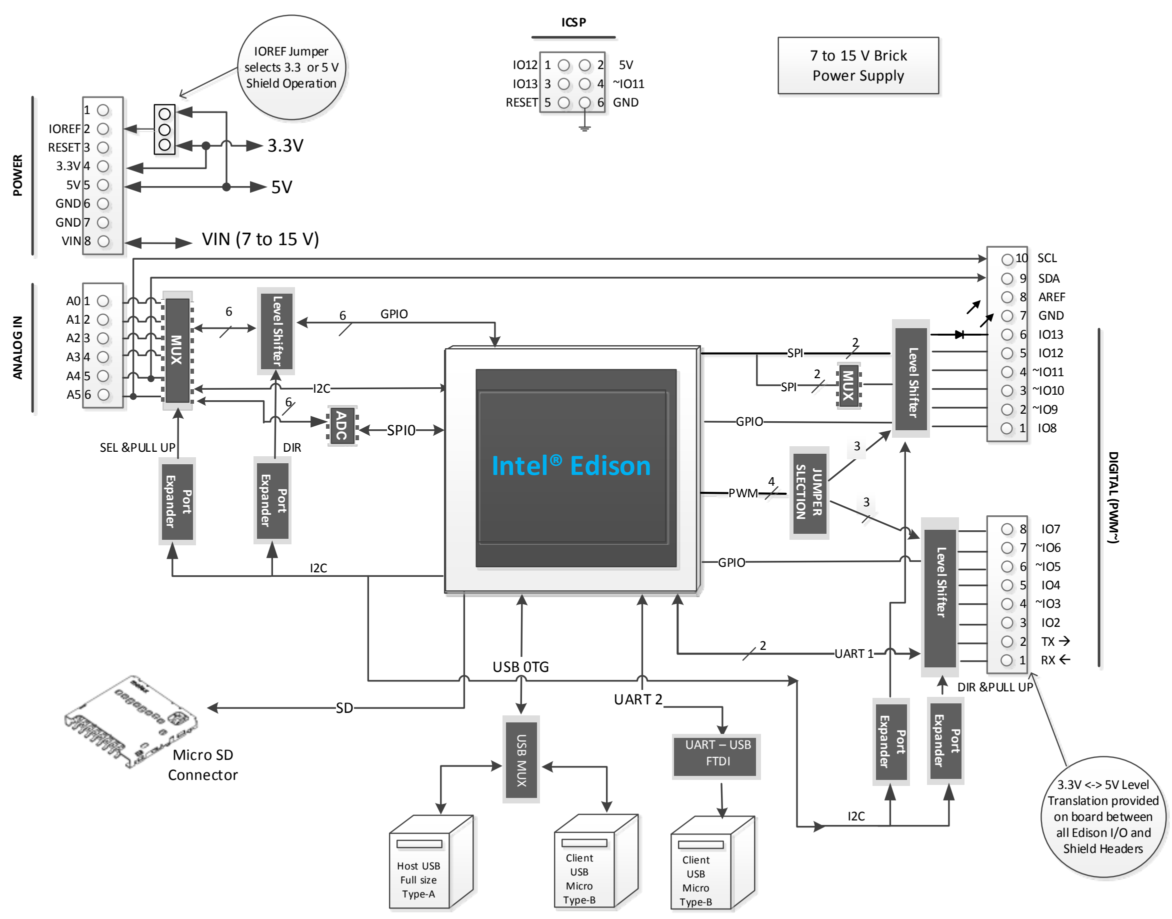
Reference	Name	Number/location
331188	Intel® Edison Board Support Package User Guide	
331189	Intel® Edison Module Hardware Guide	
331190	Intel® Edison Breakout Board Hardware Guide	
331191	Intel® Edison Kit for Arduino* Hardware Guide	(This document)
331192	Intel® Edison Native Application Guide	
331193	Intel® Edison Quick Start Guide	
[RN]	Intel® Edison Board Support Package Release Notes	
[GSG]	Intel® Edison Getting Started Guide	

§

2 High-Level Functional Description

The Intel® Edison kit for Arduino*expansion board is designed to be hardware and software pin-compatible with Arduino shields designed for the Uno R3. Digital pins 0 to 13 (and the adjacent AREF and GND pins), analog inputs 0 to 5, the power header, ICSP header, and the UART port pins (0 and 1) are all in the same locations as on the Arduino Uno R3. This is also known as the Arduino 1.0 pinout. Additionally, the Intel® Edison kit for Arduino* board includes a micro SD card connector, a micro USB device port connected to UART2, and a combination micro USB device connector and dedicated standard size USB 2.0 host Type-A connector (selectable via a mechanical microswitch).

Figure 1 Intel® Edison kit for Arduino* block diagram



2.1 Intel® Edison kit for Arduino* header signal list

The Intel® Edison kit for Arduino* digital signals can be configured as input or output. When programmed as an input, a GPIO can serve as an interrupt. The Edison board's 1.8 V I/O are translated to 3.3 or 5 V using SN74LVC1T45 dual supply bus transceivers with 3 state outputs. Both outputs go tristate if either supply rail is at ground. The port direction is referenced to VCCA. The drive level for the transceiver is: ±4 mA at 1.8 V, ±24 mA at 3.3 V, and ±32 mA at 5 V.

Note: Drive level at 1.8 V is for reference only – pertains to drive level towards Edison.

Table 2 Intel® Edison kit for Arduino* header signal list

Header	Arduino pin name	Signal function
Power	N/C	Not connected
Power	IOREF	Shield I/O reference voltage (select 3.3 or 5 V via jumper on board)
Power	RESET	Shield reset (programmable via software or manual push button)
Power	3.3 V	System 3.3 V output
Power	5 V	System 5 V output
Power	GND	Ground
Power	GND	Ground
Power	VIN	System input power (7 to 15 V)
Analog	A0	Analog input or digital I/O
Analog	A1	Analog input or digital I/O
Analog	A2	Analog input or digital I/O
Analog	A3	Analog input or digital I/O
Analog	A4 / SDA	Analog input, digital I/O, or I^2C data (also connected to digital header)
Analog	A5 / SCL	Analog input, digital I/O, or I^2C data (also connected to digital header)
Digital	SCL	I^2C clock
Digital	SDA	I^2C data
Digital	AREF	ADC reference voltage (select AREF or IOREF via jumper J8 on board)
Digital	GND	Ground
Digital	13 / SCK	Digital I/O, or SPI clock
Digital	12 / MISO	Digital I/O, or SPI receive data
Digital	~11 / MOSI	Digital I/O, SPI send data, or PWM (configured with PWM swizzler)
Digital	~10	Digital I/O, SPI signal select, or PWM (configured with PWM swizzler)
Digital	~9	Digital I/O, PWM (configured with PWM swizzler)
Digital	8	Digital I/O
Digital	7	Digital I/O
Digital	~6	Digital I/O, PWM (configured with PWM swizzler)
Digital	~5	Digital I/O, PWM (configured with PWM swizzler)
Digital	4	Digital I/O

Header	Arduino pin name	Signal function
Digital	~3	Digital I/O, PWM (configured with PWM swizzler)
Digital	2	Digital I/O
Digital	1 / TX →	Digital I/O
Digital	0 / RX ←	Digital I/O
ICSP	MISO	SPI receive data (connected to digital pin 12)
ICSP	5V	System 5 V output
ICSP	SCK	SPI clock (connected to digital pin 13)
ICSP	MOSI	SPI send data (connected to digital pin 11)
ICSP	RESET	Shield reset (programmable via software or manual push button)
ICSP	GND	Ground

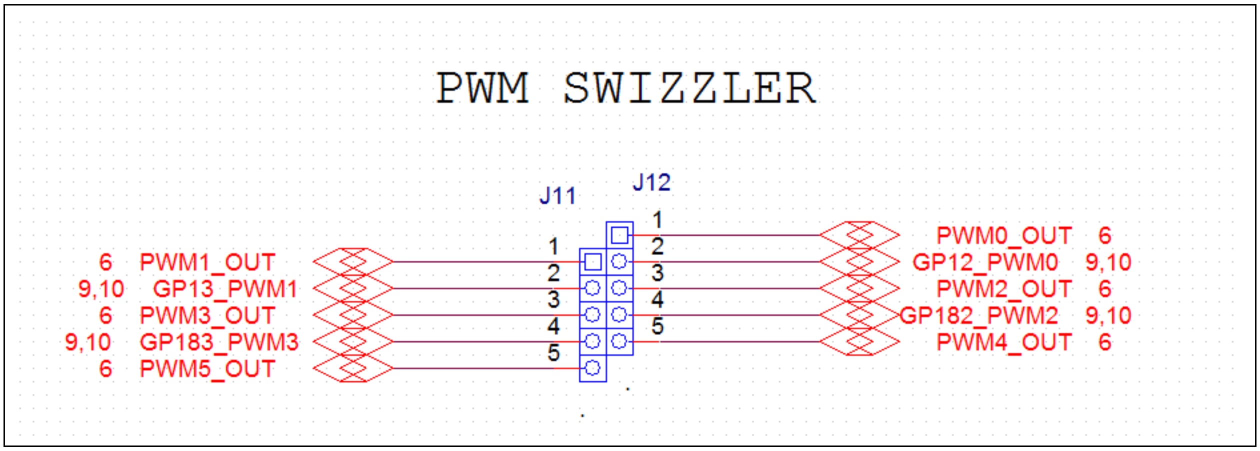
2.2 Intel® Edison kit for Arduino* PWM swizzler

There are four available GPIO that can be configured as PWM outputs. The PWM features are:

- The PWM Output Frequency and Duty Cycle can be estimated by the equations:
- Target frequency ~= 19.2 MHz * Base_unit value / 256
- Target PWM Duty Cycle ~= PWM_on_time_divisor / 256

The four PWM sources are wired to a PWM “swizzler”. This pin header arrangement allows the four PWM sources to be routed to any four of the six Arduino header pins. Figure 2 shows the PWM swizzler.

Figure 2 Intel® Edison kit for Arduino* PWM swizzler



The four PWM sources from Edison GP12_PWM0, GP13_PWM1, GP182_PWM2, and GP183_PWM3 can be configured to drive 4 of the six Arduino header PWMs. Each Edison PWM can be jumpered to one of three Arduino PWMs. For example, GP12_PWM0 can be jumpered to PWM0_OUT, PWM2_OUT, or PWM1_OUT.

Arduino multiplexing has secondary multiplexing options of SPI (or GPIO). No other PWM has these secondary multiplexing options. Therefore, if the four Edison PWMs are used and are not connected to the first four Arduino PWM pins, then those unused pins of the first four pins cannot be used as a GPIO. They will have any function; they cannot be inputs or outputs (Table 3).

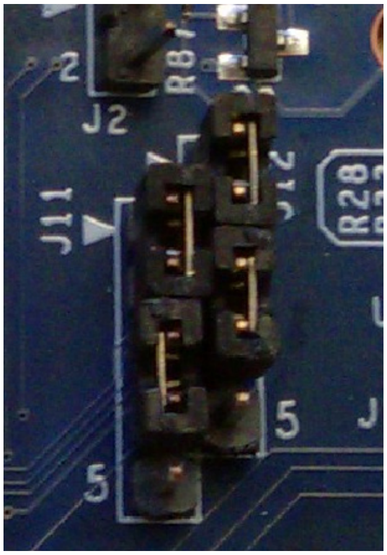
Table 3 Intel® Edison kit for Arduino* PWM swizzler signal assignments

Digital pin	Uno	Uno	Edison I/O	Edison PWM
11	IO	PWM(5)	GP43 (SSP2_TXD)	PWM3
10	IO	PWM(4)	GP41 (SSP2_FS0)	PWM3, PWM2
9	IO	PWM(3)	GP183_PWM3	PWM3, PWM2, PWM1
6	IO	PWM(2)	GP182_PWM2	PWM2, PWM1, PWM0
5	IO	PWM(1)	GP13_PWM1	PWM1, PWM0
3	IO	PWM(0)	GP12_PWM0	PWM0

Digital pins 3, 5, 6, and 9 are supported by GPx_PWMx in Edison. These pins can be configured to be either a GPIO or a PWM output. The swizzler allows the four Edison PWMs to be mapped to the six Arduino pins as shown in the last column of Table 3. For example, if PWM0 is mapped to digital pin 5, then there is no Edison pin available to connect to Digital pin 3. So this pin no longer has a function. If it is driven as an output, it will output high. If it is driven as an input, the signal is lost in the swizzler.

The default configuration is DIG3 = GP12_PWM0, DIG5 = GP13_PWM1, DIG6 = GP182_PWM2, and DIG9 = GP183_PWM3. This requires jumpers on J12 1-2, and J12 3-4, J11 1-2, and J11 3-4, as shown in Figure 3.

Figure 3 PWM swizzler on the Edison board



2.3 Intel® Edison kit for Arduino* analog inputs

The analog inputs are fed to an ADS7951 A/D converter. This device has the following features:

- 20 MHz clock rate
- 12-bit A/D conversion
- 1 MHz sample rate
- 70 dB signal to noise ratio
- 0 to 2.5 V or 0 to 5 V input range (select either AREF or IOREF via jumper J8 onboard)

The analog inputs are multiplexed with digital I/O using SN74LVC2G53 analog switches. These switches isolate the digital I/O from the analog input to prevent crosstalk. The SN74LVC2G53 also has an inhibit pin that places the I/O in a tristate condition. The switch also has low on state resistance of 15 ohm at 4.5 V VCC.

2.4 Intel® Edison kit for Arduino* signal pullup resistors

The analog and digital pins can be configured to have an external pull-up resistor connected. The pullup value is fixed at 47 kohm.

2.5 Intel® Edison kit for Arduino* USB interface

The Edison module has a single USB 2.0 interface. This interface is the primary method for downloading code. Edison is design to support OTG, using the ID signal. Circuitry on the Intel® Edison kit for Arduino* board uses a USB multiplexer, and an external switch to configure the USB interface as a host port or device port. SW1 is a slider switch which selects between host mode and device mode. When the slider is switched towards the USB standard size Type A connector, the Edison will go to host mode. When the switch is towards the micro USB Type B connector, the Edison will go to device mode. USB host mode always requires use of an external power adapter.

2.6 Intel® Edison kit for Arduino* power supply

Edison is a low power device. In general it will not draw more than 200 mA (approximately 430 mA (final value TBD) when transmitting over Wi-Fi) from the main power source. Therefore, an Intel® Edison device may run on USB power (when configured as a device), or off an external power adapter from 7 to 15 V.

Power from the external power adapter goes to a DC-DC converter and down converted to 5V. The 5 V rail is diode-ORed with the USB micro B VBUS rail. This power goes to a DC-DC converter which down converts the power to 4.4 V. This voltage is in the safe range for the Edison module VSYS. The VSYS power range is 3.3 V min to 4.5 V max. This allows VSYS to run off a standard lithium ion battery. The charger IC is configured to detect the input power source, and to limit the input power to either 500 mA (if connected to USB micro B port) or up to 1 A if connected to the DC power jack. The charger is programmed to charge at 100 mA. This charger is designed to charge standard lithium ion batteries with 4.2 V maximum charging voltage. End-users are responsible for choosing a suitable battery and following all safety precautions, to assure overcharging or charging when the battery temperature is too high is avoided.

For low power applications (those shields running off 3.3 V) a lithium ion battery (3.0 to 4.3 Vmax) can be attached to J2, which will power Edison and provide 100 mA of 3.3 V to the shield.

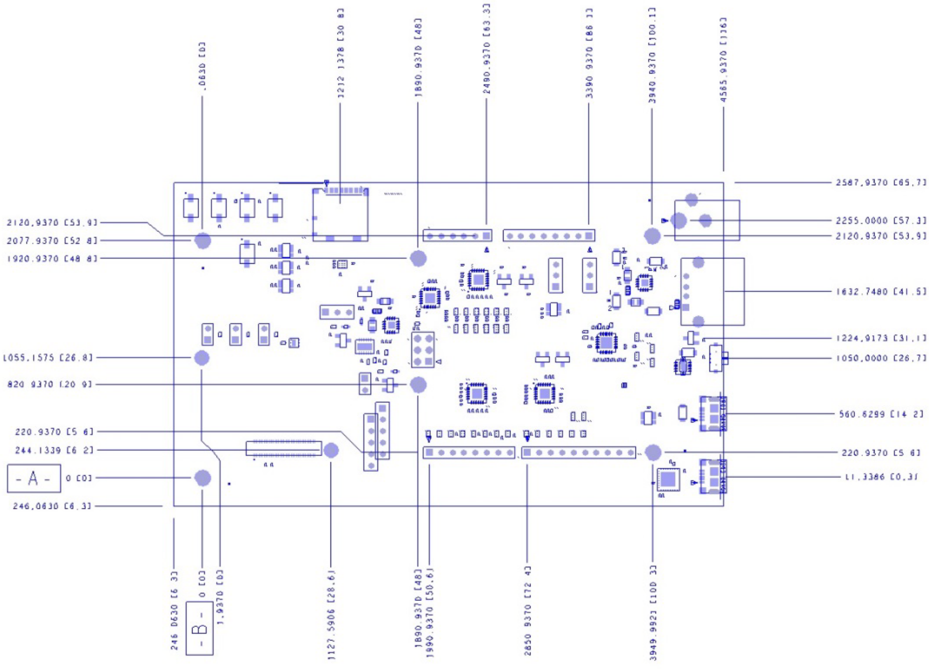
Some considerations of the power distribution in the Intel® Edison kit for Arduino*:

- Due to the diode ORing of the 5 V DC/DC and the VBUS input, means the 5 V power to the shield header will be nominally below 5 V. In the case of VBUS the voltage may be as low as 4.4 V (4.75 V VBUS min – 0.3 V diode drop. In the case of external power adapter 4.7 V.
- USB host mode always requires use of an external power adapter.

2.7 Intel® Edison kit for Arduino* expansion mechanicals

Figure 4 lists the dimensions (in thousands of inches and [mm]) of the Intel® Edison kit for Arduino* board.

Figure 4 Intel® Edison kit for Arduino* mechanical dimensions

H40482-004_edsion_ arduino_HVM_dim.dxf	

§

3 Powering Edison

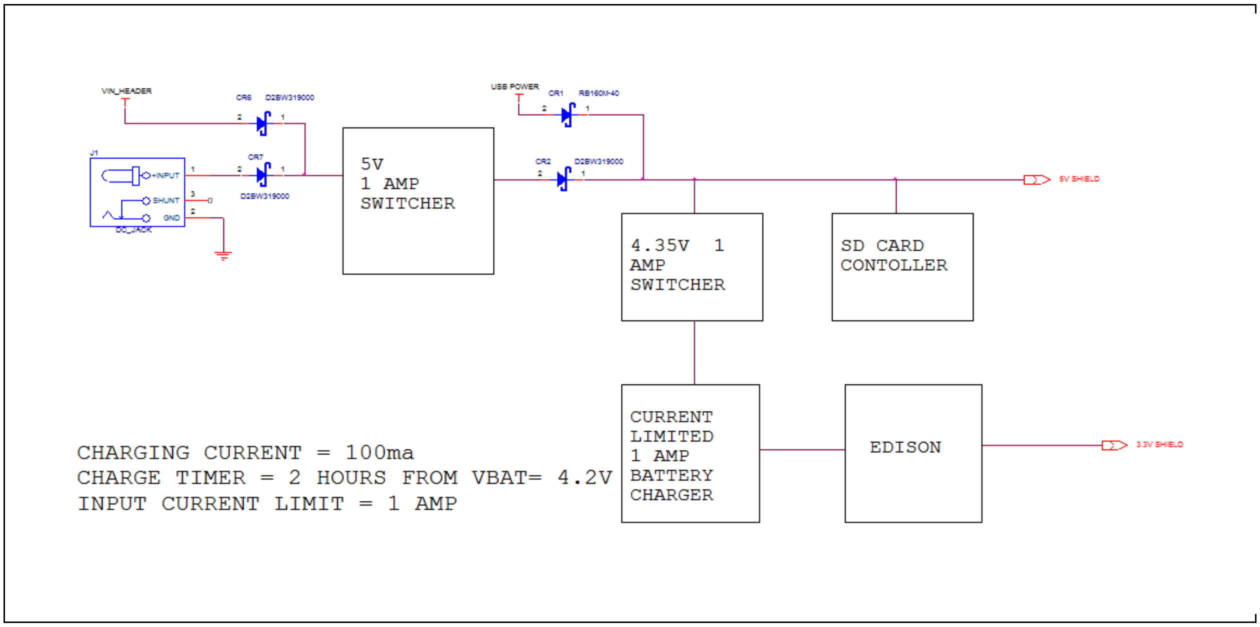
You can power the Intel® Edison board using any of the following:

- an external power supply on J1;
- DCIN via shield header pin VIN;
- a USB cable via micro USB connector J16; or
- a lithium-ion battery connected to J2.

When power is applied to J1 or VIN, the external power must be in the range of 7 to 17 V. The power is converted to 5 V via a switching power supply, which powers the rest of the system. This supply was designed for a 1 A continuous supply. Higher currents will generate more power losses and may thermally damage the switcher. The switcher does have internal short circuit protection, and thermal shutdown protection. The end-user should not rely on thermal not short circuit protection.

Figure 5 shows the power distribution network of the Intel® Edison kit for Arduino*.

Figure 5 Intel® Edison kit for Arduino* power distribution network



Power from the 5 V switcher is diode-ORed with power from the USB connector. This arrangement allows Edison Arduino to run off external power or USB power. This rail is used to power the shields, the SD card slot, and a 4.35 V switcher. The total current on this rail should be limited to 1 A maximum continuous.

The 4.35 V rail powers a battery charger and the Edison module. The 4.3 V supply is also designed to generate 1 A, and has the same protections (thermal and short circuit) as the 5 V supply.

The charger is designed to only accept 1 A maximum from the 4.35 V rail, and will charge a battery at 100 mA. The charger will supply power from the 4.35 V input or from the battery (if attached). The charger will charge the battery (from the 4.35 V supply) autonomously using whatever power is left over from powering Edison.

For low voltage systems, Edison can provide 3.3 V at 250 mA to the shields. The user should limit the current from the Edison 3.3 V rail. Higher currents will cause the 3.3 V output to droop (due to IR losses), and may cause excessive heating of the Edison module.

Edison is a low power device. It normally operates at 200 mA. During Wi-Fi transmit bursts, the current could reach 600 mA for milliseconds. The sum of the Edison current, recharging, SD card, and shield power could exceed the 500 mA specification. This could cause triggering of the USB power switch within a PC, causing loss of USB functionality until the PC is restarted.

Some considerations of the power distribution in the Edison Arduino:

- There is a diode ORing of the 5 V DC/DC and the VBUS input. In the case of powering Edison Arduino from VBUS, the shield voltage may be as low as 4.4 V (4.75 V VBUS min – 0.3 V diode drop). In the case of external power adapter, voltage to the shield will be 5 V ±2%.
- Using Edison as a USB HOST requires use of an external adapter.
- End-users are responsible for choosing a suitable battery and following all safety precautions, to prevent overcharging or charging when the battery temperature is too high. The battery should be at least 200 mAH capacity due to the 100 mA charging current. We recommend battery packs with internal protection circuits.

3.1 Boot voltage selection – DCIN signal

DCIN is a signal that indicates whether Edison is being powered from a battery or from an external power source. DCIN also sets the voltage level required on VSYS in order to boot. When DCIN is floating or tied to ground, the voltage on VSYS *must* rise from 2.5 to 3.5 V in 10 ms; otherwise the boot is aborted. When the boot is aborted, power must be cycled below 2.5 V. If DCIN is connected to VSYS, Edison will start to boot when VSYS is above 2.5 V for 100 ms.

Note: When DCIN is connected to VSYS, boot will occur whenever the voltage is above 2.8 V for 100 ms. The DCIN signal is attached to VSYS on the PCB.

Note: The absolute minimum voltage to assure Wi-Fi and Bluetooth functionality is 3.15 V.

§

4 Batteries

The rechargers chosen on the Intel® Edison kit for Arduino* and the Intel® Edison breakout board were designed for lithium-ion or lithium-polymer batteries. Follow the manufacturer's guidelines when charging batteries. Generally, charging current should not exceed 50 to 70% of the rated capacity. For example, a 200 mAH battery should be charged with 70% • 200 mA = (140 mA).

The Intel® Edison kit for Arduino has a 100 mA charging current; the Intel® Edison breakout board has a 190 mA charging current.

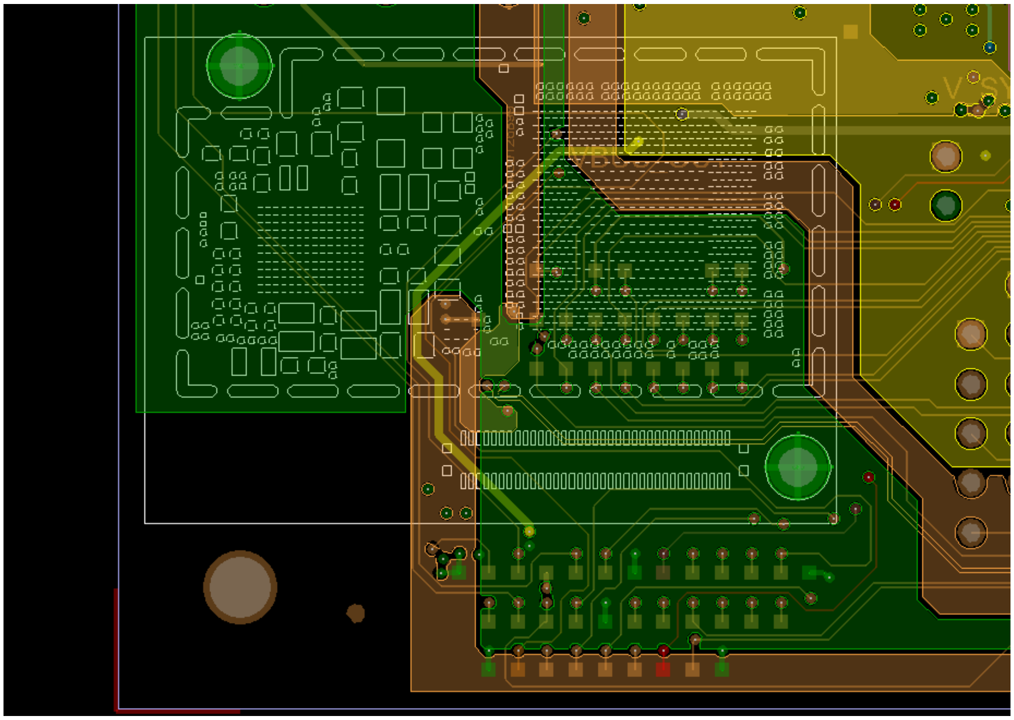
§

5 Layout

5.1 Antenna keepout

Placeholder per Brian G's request. See Figure 6.

Figure 6 Area around antenna



5.2 Layout SD card, I2S, SPI, I2C

Table 4 Layout SD card

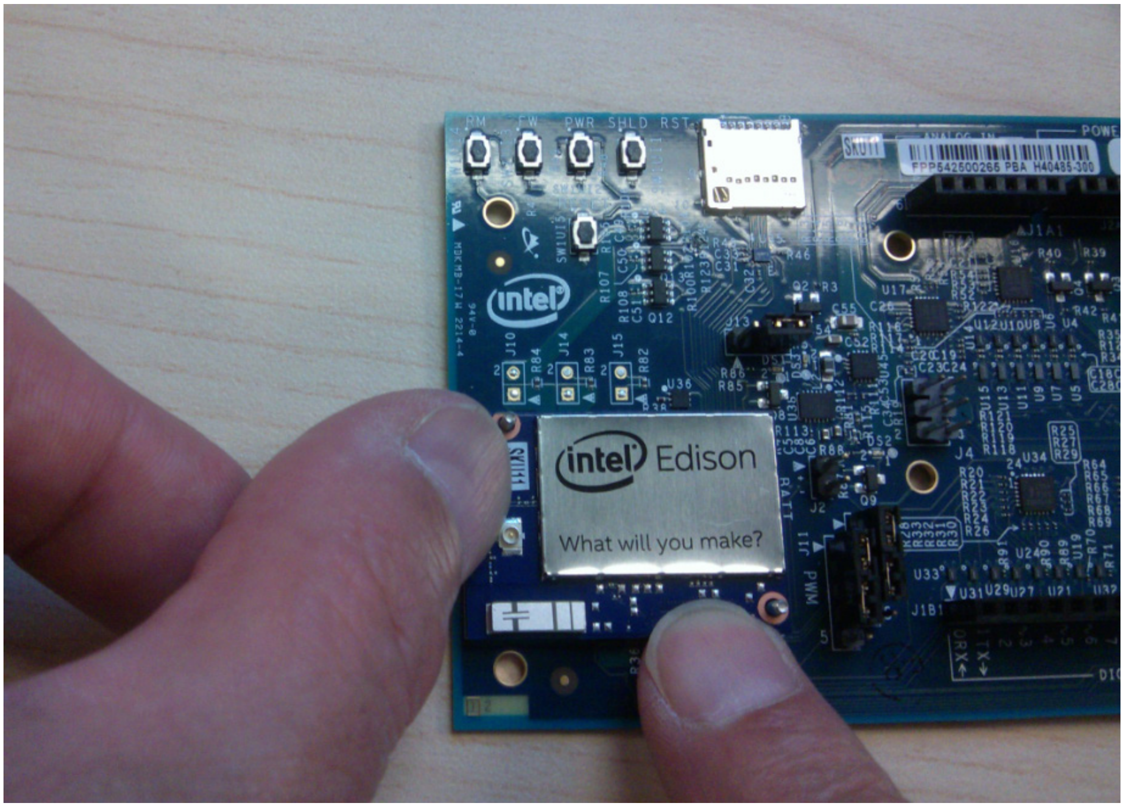
Signal parameter	Metric (mm)	Standard (mils)
Total length L1		
DATA/CMD/CTRL to CLK maximum pin-to-pin length mismatch		
Minimum main route spacing ratio		
CLK to DATA/CMD/CTRL matching	(also a placeholder)	
Characteristic single ended impedance		
Load capacitance		

Note: 1) For SPI, total length is xxxxx mils.
2) For I^{2}C, total length is xxxxx mils.

6 Handling

When assembling an Intel® Edison module to an Arduino board, handle the Intel® Edison module by the PCB edges. Avoid holding or exerting pressure to the shields. To mate the Intel® Edison board to the Arduino* board, apply pressure directly above the connector and to the left corner, as shown in Figure 7.

Figure 7 Inserting an Edison module



§

7 Debug UART Errata

The Intel® Edison board has a known error on all UARTs. When Edison goes into low power sleep, the UART internal FIFO and interface is powered down. Therefore, a two-wire UART (Rx/Tx) will lose the first received character whenever Edison is in sleep mode. In order to avoid this condition, when sleep mode is enabled, a four-wire UART (Rx, Tx, CTS, and RTS) is required.

§

8 Buttons

This section explains the software functionality of the Intel® Edison board's buttons.

The Intel® Edison module has the following buttons:

- **System reset.** Pressing the system reset button (SW1UI5) will reset the Intel® Edison board, and reset the I/O expanders, setting all the shield pins to high impedance state with no pullups.
- **Shield reset.** Pressing the shield reset button (SW1UI1) will pull the shield signal reset to the active low state. It does not affect the state of the Edison module nor its I/O.
- **Power button.** The power button (SW1UI2) is configured by software. In general, pressing and holding this button will cause the Intel® Edison module to power down. (It will leave the I/O configuration in the port expanders in its current state.) Pressing this button momentarily when Edison is powered down (but power is still applied) will cause the Intel® Edison module to reboot. If Edison is running, then a momentary press will cause Edison to go into low power sleep mode. Pressing the button momentarily when Edison is asleep, will bring Edison into full power mode. You must press and hold SW1U15 for 8 seconds to reset Edison. Pressing the reset button for 4 seconds will restart Edison.

8.1 FWR_RCVR and RCVR_MODE

SW1UI3 and SW1UI4 are used to recover an Edison that has a corrupted software image. Powering off Edison and then pressing FWR_RCVR and then applying power will cause Edison to go to firmware recovery mode. Edison will be ready to receive a new image over USB. Two recovery modes are available depending on the state of the SW1UI4 when power is applied.

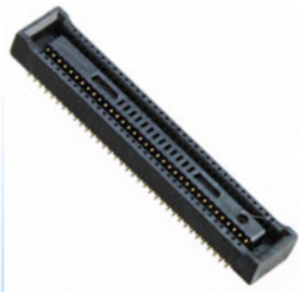
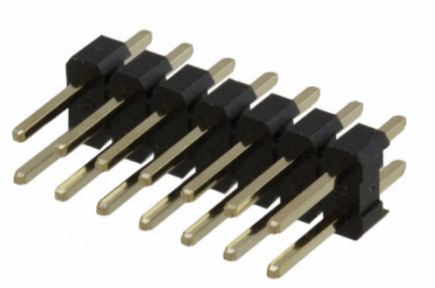
During boot, If FWR_RCVR is high (button FW pressed), the processor attempts to load the firmware from flash from the location specified in the UMIP header. During boot, if FWR_RCVR is high (button FW pressed) and RCVR_MODE is low (button RM pressed), the processor attempts to download from USB-B port, regardless of the UMIP header.

§

9 Digikey sources

Figure 8 shows some third-party accessories you can use.

Figure 8 Digikey sources

 **Edison Mating connector 2.0 mm** **DF40C(2.0)-70DS-0.4V(51) - H11908CT-ND Cut tape** **DF40C(2.0)-70DS-0.4V(51) H11908TR-ND Tape and Reelt**	 **Mini-breakout power jack PJ-002BH-SMT-TR** **CP-002BHPJCT-ND Cut tape** **PJ-002BH-SMT-TR CP-002BHPJTR-ND Tape and reel**
 Mini-breakout USB adapter cable **USB A female to Micro A male 10-00649 839-1105-ND**	 **Mini-breakout male header** **2x14 M20-9980745 952-1932-ND**

§