Arduino Ethernet

Overview

The Arduino Ethernet is a microcontroller board based on the ATmega328 (<u>datasheet</u>). It has 14 digital input/output pins, 6 analog inputs, a 16 MHz crystal oscillator, a RJ45 connection, a power jack, an ICSP header, and a reset button.

NB: Pins 10, 11, 12 and 13 are reserved for interfacing with the Ethernet module and should not be used otherwise. This reduces the number of available pins to 9, with 4 available as PWM outputs.

An optional Power over Ethernet module can be added to the board as well.
The Ethernet differs from other boards in that it does not have an onboard USB-to-serial driver chip, but has a Wiznet Ethernet interface. This is the same interface found on the Ethernet shield.

An onboard microSD card reader, which can be used to store files for serving over the network, is accessible through the SD Library. Pin 10 is reserved for the Wiznet interface, SS for the SD card is on Pin 4.

The 6-pin serial programming header is compatible with the <u>USB Serial</u> adapter and also with the FTDI USB cables or with Sparkfun and Adafruit FTDI-style basic USB-to-serial breakout boards. It features support for automatic reset, allowing sketches to be uploaded without pressing the reset button on the board. When plugged into a USB to Serial adapter, the Arduino Ethernet is powered from the adapter.

Summary

Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage Plug (recommended)	7-12V
Input Voltage Plug (limits)	6-20V
Input Voltage PoE (limits)	36-57V
Digital I/O Pins	14 (of which 4 provide PWM output)

Arduino Pins reserved:

10 to 13 used for SPI

4 used for SD card

2 W5100 interrupt (when bridged)

Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Clock Speed	16 MHz
W5100 TCP/IP Embedded Ethernet Controller	
Power Over Ethernet ready Magnetic Jack	
Micro SD card, with active voltage translators	

Schematic & Reference Design
EAGLE files: arduino-ethernet-reference-design.zip
Schematic: arduino-ethernet-schematic.pdf

Power
The board can also be powered via an external power supply, an optional Power over Ethernet (PoE) module, or by using a FTDI cable/USB Serial connector.

External power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The power pins are as follows:

- VIN. The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- 5V. This pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 - 12V), the USB connector (5V), or the

2

VIN pin of the board (7-12V). Supplying voltage via the 5V or 3.3V pins bypasses the regulator, and can damage your board. We don't advise it.

- 3V3. A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- GND. Ground pins.

The optional PoE module is designed to extract power from a conventional twisted pair Category 5 Ethernet cable:

- IEEE802.3af compliant
- Low output ripple and noise (100mVpp)
- Input voltage range 36V to 57V
- Overload and short-circuit protection
- 9V Output
- High efficiency DC/DC converter: typ 75% @ 50% load
- 1500V isolation (input to output)

NB: the Power over Ethernet module is proprietary hardware not made by Arduino, it is a third party accessory. For more information, see the **datasheet**

When using the power adapter, power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

Memory
The ATmega328 has 32 KB (with 0.5 KB used for the bootloader). It also has 2 KB of SRAM and 1 KB of EEPROM (which can be read and written with the **EEPROM library**).

Input and Output
Each of the 14 digital pins on the Ethernet board can be used as an input or output, using **pinMode()**, **digitalWrite()**, and **digitalRead()** functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

- Serial: 0 (RX) and 1 (TX). Used to receive (RX) and transmit (TX) TTL serial data.
- External Interrupts: 2 and 3. These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the **attachInterrupt()** function for details.
- PWM: 3, 5, 6, 9, and 10. Provide 8-bit PWM output with the **analogWrite()** function.
- SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). These pins support SPI communication using the **SPI library**.
- LED: 9. There is a built-in LED connected to digital pin 9. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off. On most other arduino boards, this LED is found on

pin 13. It is on pin 9 on the Ethernet board because pin 13 is used as part of the SPI connection.

The Ethernet board has 6 analog inputs, labeled A0 through A5, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though is it possible to change the upper end of their range using the AREF pin and the analogReference() function. Additionally, some pins have specialized functionality:

* TWI: A4 (SDA) and A5 (SCL). Support TWI communication using the Wire library.

There are a couple of other pins on the board:

* AREF. Reference voltage for the analog inputs. Used with analogReference().
* Reset. Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

See also the mapping between Arduino pins and ATmega328 ports.

Communication

The Arduino Ethernet has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers.

A SoftwareSerial library allows for serial communication on any of the Uno's digital pins. The ATmega328 also supports TWI and SPI communication. The Arduino software includes a Wire library to simplify use of the TWI bus; see the documentation for details. For SPI communication, use the SPI library.

The board also can connect to a wired network via ethernet. When connecting to a network, you will need to provide an IP address and a MAC address. The Ethernet Library is fully supported.

The onboard microSD card reader is accessible through the SD Library. When working with this library, SS is on Pin 4.

Programming

It is possible to program the Arduino Ethernet board in two ways: through the 6 pin serial programming header, or with an external ISP programmer.

The 6-pin serial programming header is compatible with FTDI USB cables and the Sparkfun and Adafruit FTDI-style basic USB-to-serial breakout boards including the Arduino USB-Serial connector. It features support for automatic reset, allowing sketches to be uploaded without pressing the reset button on the board. When plugged into a FTDI-style USB adapter, the Arduino Ethernet is powered off the adapter.

You can also program the Ethernet board with an external programmer like an AVRISP mkII or USBTinyISP. To set up your environment for burning a sketch with a programmer, follow these instructions. This will delete the serial bootloader, however.

All the Ethernet example sketches work as they do with the Ethernet shield. Make sure to change the network settings for your network.

Physical Characteristics

The maximum length and width of the Ethernet PCB are 2.7 and 2.1 inches respectively, with the RJ45 connector and power jack extending beyond the former dimension. Four screw holes allow the board to be attached to a surface or case. Note that the distance between digital pins 7 and 8 is 160 mil (0.16"), not an even multiple of the 100 mil spacing of the other pins.

Setup

If you want to use a FTDI cable to download your sketches on the Arduino Ethernet, please refer to this guide: <u>Upgrade the Arduino Ethernet bootloader to the latest version</u>