

Hello Arduino Uno

Getting a “Hello World” with the GNU GCC toolchain

Revision	Author	Date	Description
0.01	D Snider	15/11/05	Initial Outline
0.02	D Snider	15/11/12	Cleanup screenshots, updated to 115.2kbaud, added putty ref

Table of Contents

1. SW DEV ENVIRONMENT.....	3
2. HW TOOLS.....	3
2.1. ATMEL JTAGICE3.....	3
3. HELLO UNO.....	4
3.1. BUILD.....	4
3.2. CONNECT ISP POD TO ATMEL STUDIO AND UNO.....	5
Connect to ISP pod.....	5
Verify Pod connection to Atmega328p.....	7
3.3. PROGRAMMING ATMEGA328P.....	8
Erase Chip.....	8
Fuses.....	8
Load hello_uno.hex.....	9
3.4. VERIFY 1SEC LED.....	9
3.5. ONE-TIME WINDOWS DRIVER INSTALL.....	10
3.6. VERIFY SERIAL I/O.....	11
4. DESIGN NOTES - FUSES.....	13
5. RESOURCES.....	13

NOTE: WHILE THIS DOCUMENT IS BEING REFINED, META INFORMATION WILL BE IN < BRACKETS > .

1. SW Dev Environment

Google keywords	Description & URLs	tested versions
Cygwin	GNU file utils and make for Windows https://www.cygwin.com/	> make --version GNU Make 4.1
TortoiseGit	Windows based GIT: https://tortoisegit.org/ https://git-for-windows.github.io/	> git --version git version 2.6.2.windows.1
WinAvr	Windows GCC Compiler, Linker and C libraries http://sourceforge.net/projects/winavr/	> avr-gcc --version avr-gcc.exe (winAVR 20100110) 4.3.3
Atmel Studio	Programmer Software which uses the JTAGICE3 debug pod http://www.atmel.com/tools/atmelstudio.aspx	Version: 7
putty	A serial terminal program with reliable cut & paste. http://www.chiark.greenend.org.uk/~sgtatham/putty/	Version: 0.66

2. HW tools

2.1. Atmel JTAGICE3



Google	"Atmel JTAGICE3"
Source	Digikey: http://www.digikey.com/product-detail/en/ATJTAGICE3/ATJTAGICE3-ND/
Support pages	http://www.atmel.com/tools/JTAGICE3.aspx http://www.atmel.com/webdoc/jtagice3/

3. Hello Uno

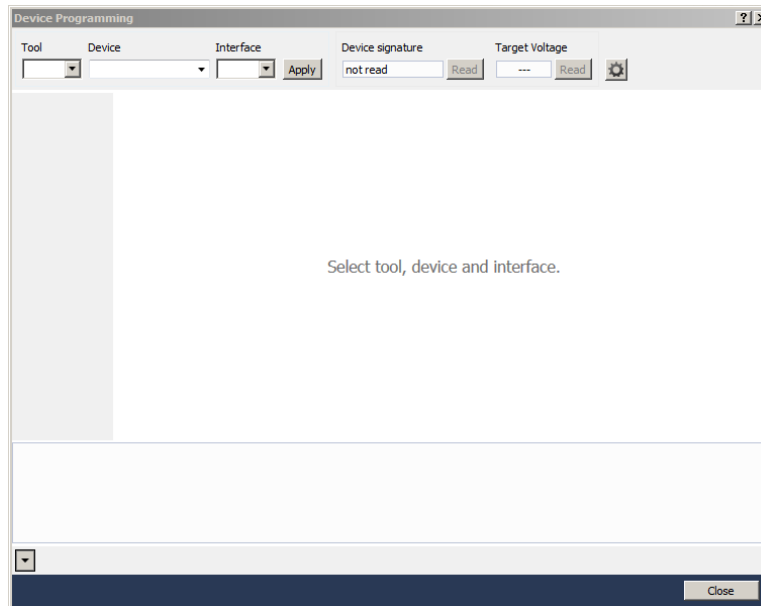
3.1. Build

```
> git clone https://github.com/dsnider0909/hello_arduino.git
> cd hello_arduino/uno
> make
...
Linking: hello_uno.elf
...
Creating load file for Flash: hello_uno.hex
...
> ls hello_uno.hex
hello_uno.hex
```

3.2. Connect ISP pod to Atmel Studio and Uno

Connect to ISP pod

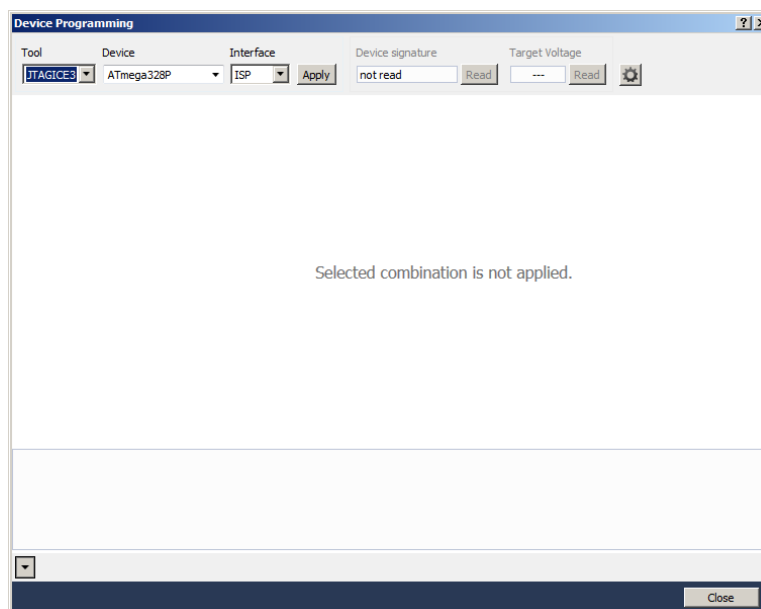
within Atmel Studio, Select: Tools→ Device Programming



Connect the JTAGICE3 pod to a windows PC USB port.

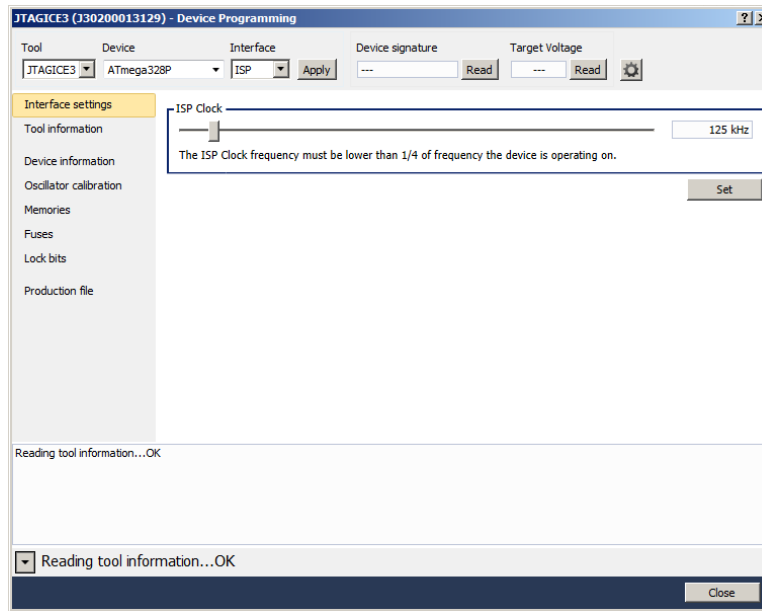
Configure:

Tool: JTAGICE3
Device: ATmega328P
Interface: ISP



select: Apply

If the pod is found, the following screen should show:

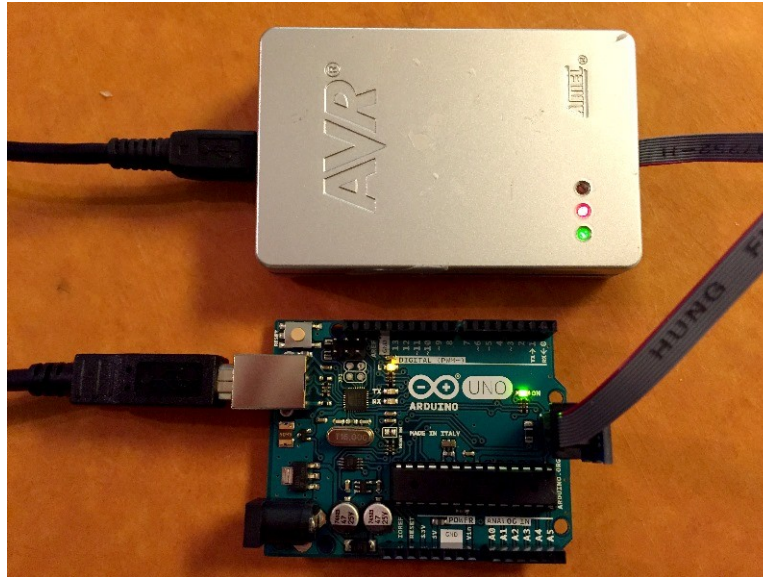


Verify Pod connection to Atmega328p

Connect the Arduino Uno board to a USB cable.

Connect this USB cable to another windows PC for both power and serial I/O.

Connect the JTAGICE3 6pin ISP header to the UNO header: ICSP.



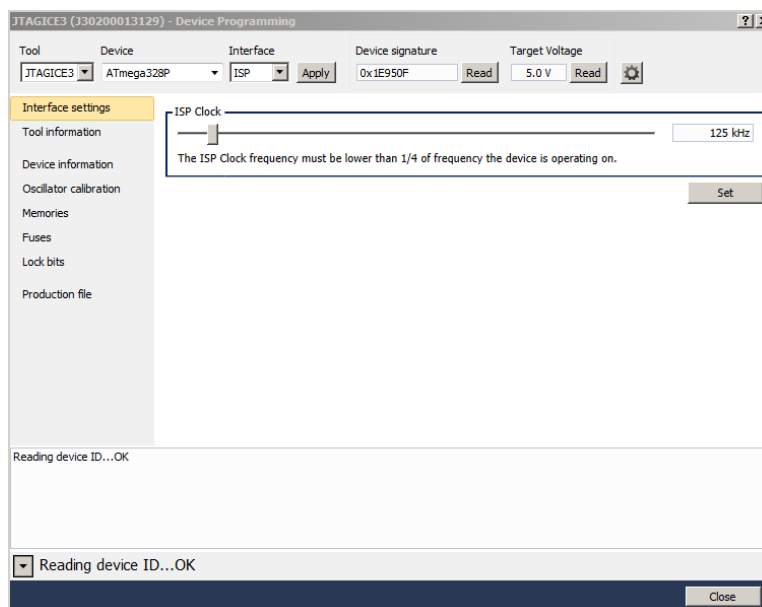
Select: Target Voltage → Read

It should return 3 to 5V.

Select: Device signature → Read

There should be no error messages of Device ID mismatch.

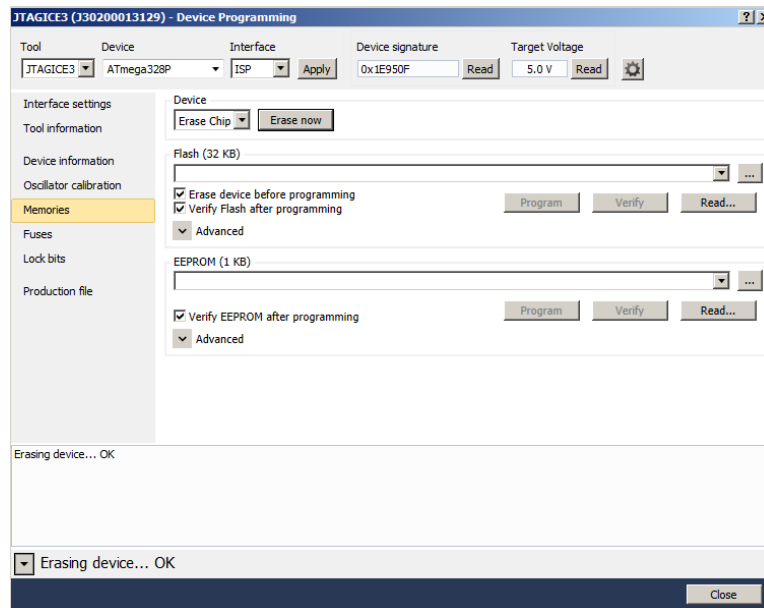
For the Atmega328P, the signature should be 0x1E950F.



3.3. Programming Atmega328p

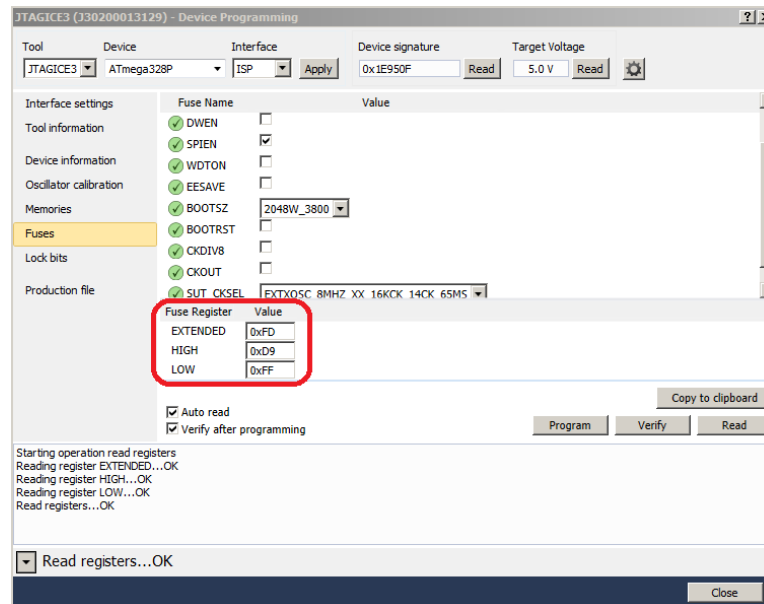
Erase Chip

Select: Memories → Device: Erase Chip → Erase Now



Fuses

Select: Fuses



For Arduino Uno - ATmega328p:

EXTENDED: 0xFD

HIGH: 0xD9

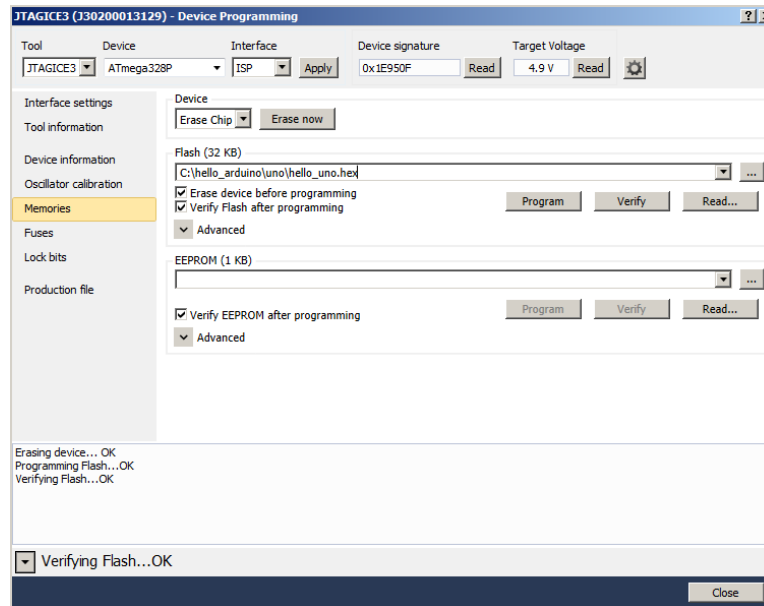
LOW: 0xFF

Load hello_uno.hex

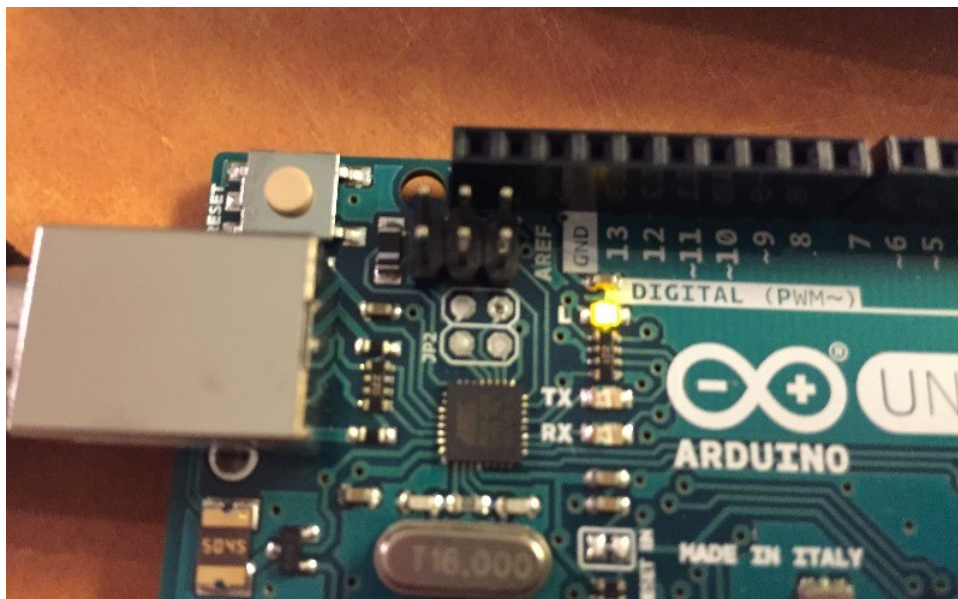
Select: Memories

Select: Flash: hello_arduino\uno\hello_uno.hex

Select: Program



3.4. Verify 1sec LED



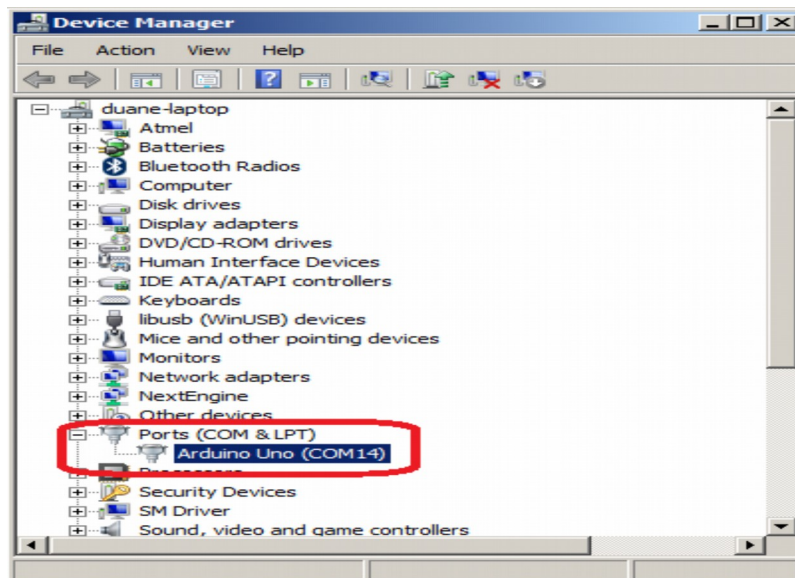
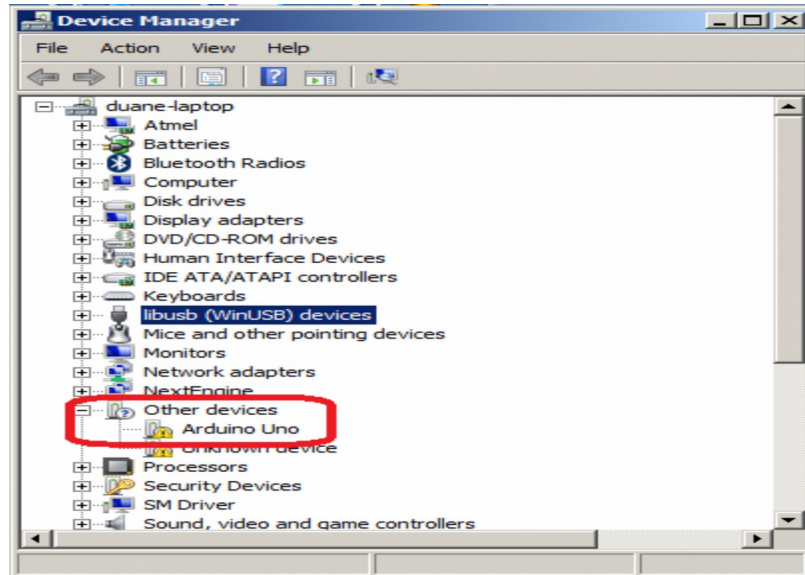
This should precisely transition once a second.

3.5. One-time Windows Driver Install

within windows, open the “Device Manager”

Select: Start, at the prompt enter: devmgmt.msc

Check whether there is a “Ports – COM port” defined,
or an “Other devices/ Arduino Uno” showing?



The “Other devices/ Arduino Uno” means a driver needs to be installed:

Right Click “Other Devices / Arduino Uno”

Select Driver Update and point the update to the directory:

'hello_arduino/Uno/doc' directory for the *.inf file

References:

<https://www.arduino.cc/en/Main/USBSerial>

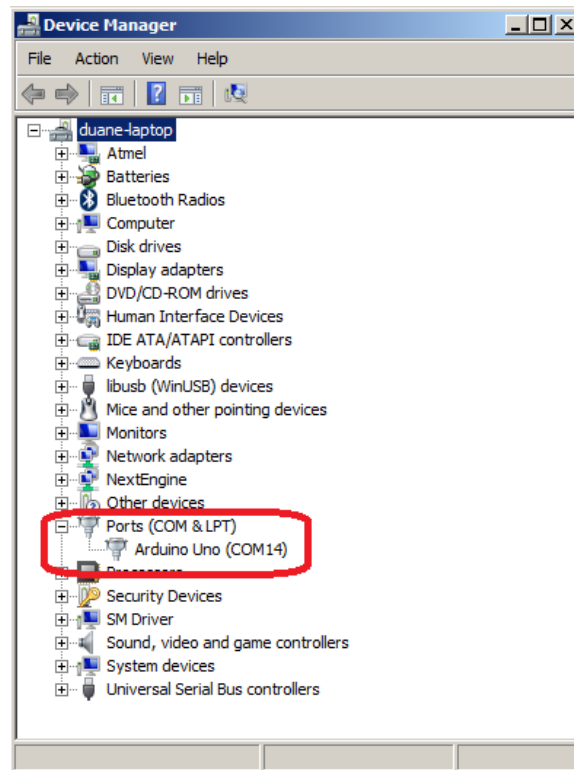
<https://www.arduino.cc/en/Guide/windows#toc4>

3.6. Verify Serial I/O

Open the windows “Device Manager”

within windows: select Start, at the prompt enter: devmgmt.msc

Search for what COM ports are connected.



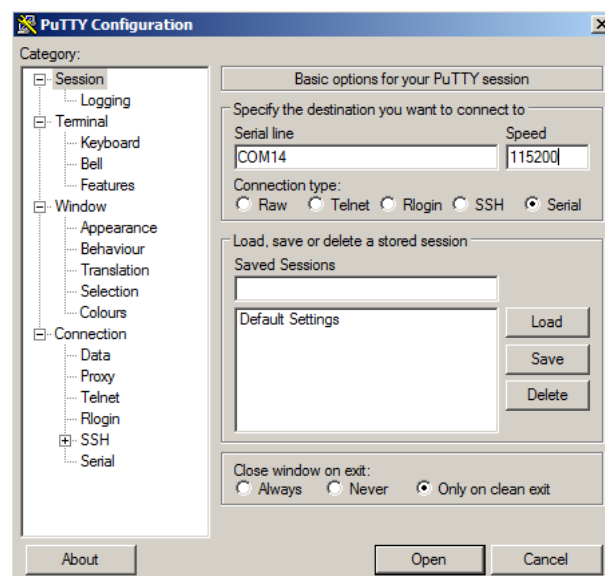
Start up a Putty window and configure:

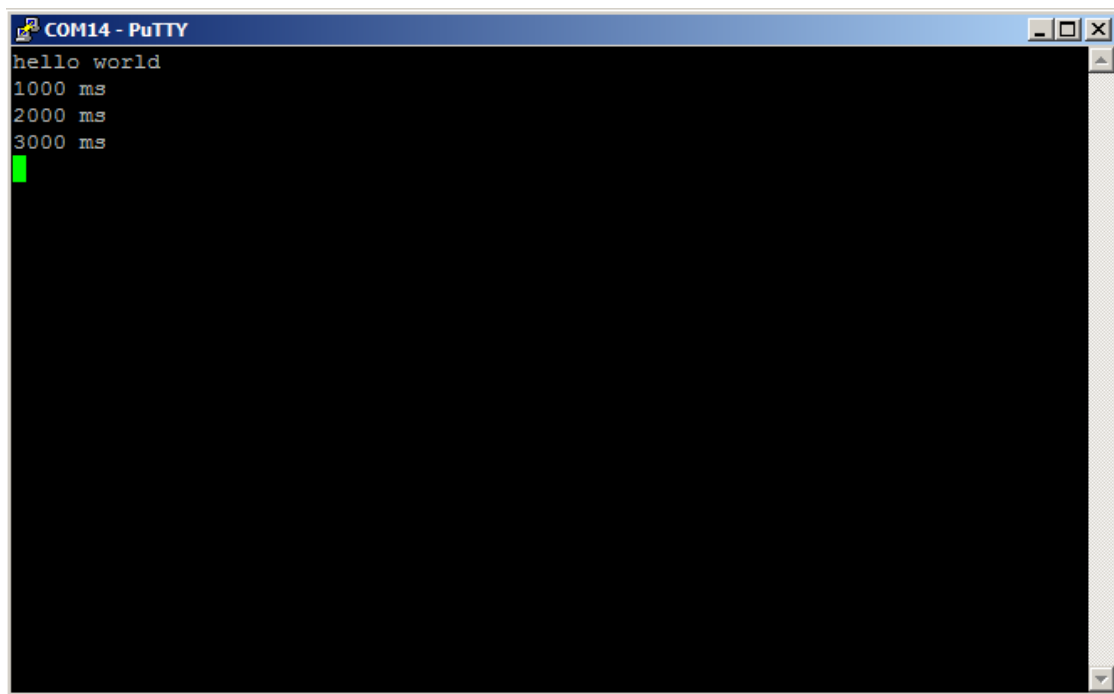
Connection type: serial

Serial line: COM14 (from this example)

Speed: 115200

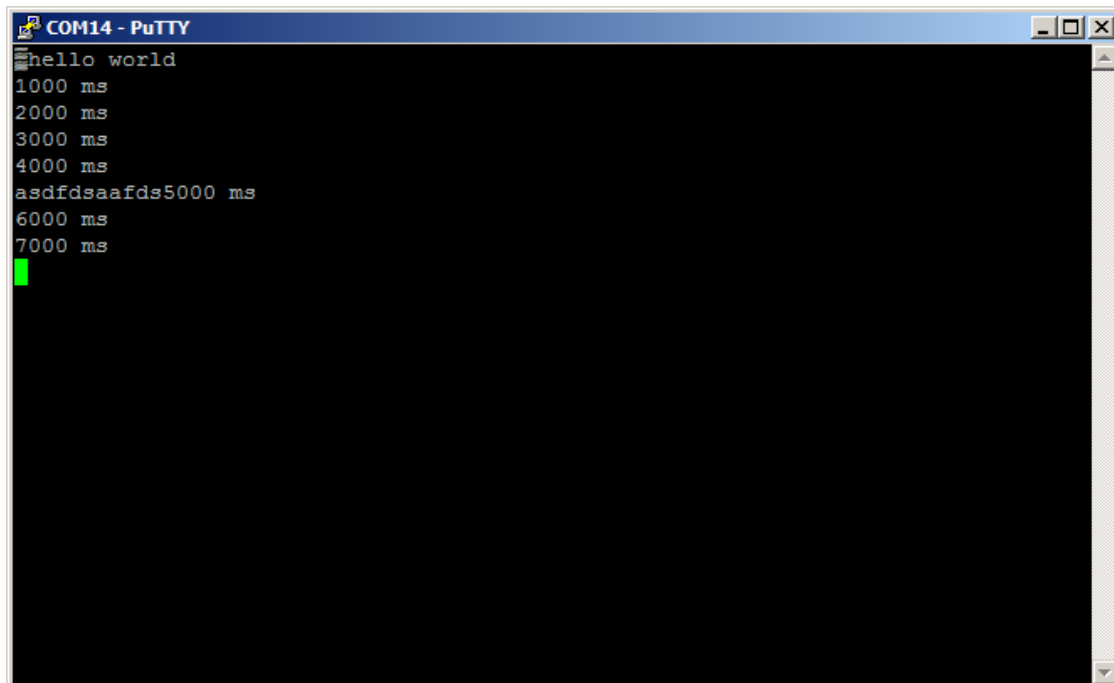
Select: Open





```
COM14 - PuTTY
hello world
1000 ms
2000 ms
3000 ms
█
```

Hit a few keys to verify the Arduino is receiving serial data:



```
COM14 - PuTTY
hello world
1000 ms
2000 ms
3000 ms
4000 ms
asdfsaaafds5000 ms
6000 ms
7000 ms
█
```

4. Design Notes - Fuses

Atmega328P Fuse Notes:

```
Extended:      // 7 6 5 4 3 2 1 0
                // 1 1 1 1 1 1 1 1 => Atmel factory default: FF
                FD // 1 1 1 1 1 1 0 1 => Arduino UNO setting
                   +---+---+ BODLVL[2.0]: 111 - Disabled
                                   100 - 4.3V
                                   101 - 2.7V

High:          // 7 6 5 4 3 2 1 0
                // 1 1 0 1 1 0 0 1 => Atmel factory default
                D9 // 1 1 0 1 0 1 1 0 => Arduino UNO original setting: D6
                   // 1 1 0 1 1 0 0 1
                   // | | | | | | | + BOOTRST: 1 - no Boot Loader
                   // | | | | | | +---+---+ BOOTSZ[1.0]: 00 - 2K words
                   // | | | | | | +-----+ EESAVE: 1 - EEPROM not reserved
                   // | | | | | +-----+ WDTON: 1 - Watchdog timer off
                   // | | | +-----+ SPIE: 0 - SPI programming enabled
                   // | | +-----+ DWEN: 1 - debugwire disabled
                   // | +-----+ RSTDISBL: 1 - External reset enabled
                   // +-----+

Low:           // 7 6 5 4 3 2 1 0
                // 0 1 1 0 0 0 1 0 => factory default: 62
                // 1 1 1 0 0 0 1 0 => internal 8Mhz RC: E2 → F_CPU = 8000000
                FF // 1 1 1 1 1 1 1 1 => 8-16Mhz ext crystal → F_CPU = xtal = 16000000
                   // | | | | | | | + CLKSEL[3.0]: 0010 - internal 8Mhz
                   // | | | | | | +---+---+ 111x - 8-16Mhz ext crystal
                   // | | | | | | +-----+ 110x - 3-8Mhz ext crystal
                   // | | | +-----+ SUT[1.0],CLKSEL[0]: 11,1 - slowest start up
                   // | | +-----+ CKOUT:1 - CLK not out on PB0
                   // | +-----+ CKDIV8: 1 - do not divide clk by 8
                   // +-----+

UNO : FD D9 FF
```

5. Resources

Google keywords	Description
Atmega 328p datasheet	<i>The final word on register settings.</i> http://www.atmel.com/devices/atmega328p.aspx http://www.atmel.com/images/Atmel-8271-8-bit-AVR-Microcontroller-ATmega48A-48PA-88A-88PA-168A-168PA-328-328P_datasheet_Complete.pdf
Arduino uno schematic r3	https://www.arduino.cc/en/uploads/Main/Arduino_Uno_Rev3-schematic.pdf
AVR libc	WinAVR library source and documentation http://www.nongnu.org/avr-libc/
AVR freaks	A good online community for most AVR questions http://www.avrfreaks.net/